

АНАЛИЗ АКТИВНОСТИ ЧЕЛОВЕКА В ЗАДАЧЕ АВТОМАТИЗИРОВАННОГО УПРАВЛЕНИЯ «УМНЫМ ДОМОМ»

Кириенко А.С.¹, Соловьев И.П.¹

¹СПбГУ, Санкт-Петербург, Россия

Аннотация

В статье рассматриваются основные подходы к решению задачи автоматизированной поддержки среды обитания человека, которую в современной литературе часто называют «умным домом». Предлагается подход, основанный на использовании информации о силе сигнала радио-маячков и показаний датчика сердцебиения.

В статье описывается модельная среда обитания, обладающая указанными характеристиками: квартира с размещенными Bluetooth-маячками, обитатель которой снабжен датчиком сердцебиения и устройством, измеряющим силу сигнала от маячков. Была реализована программа, которая в процессе эксплуатации модельной среды собирает данные о силе сигнала маячков, показания датчика сердцебиения и предоставляет возможность показывать текущее состояние приборов, являющихся объектами автоматизации. На полученных данных с помощью известных методов машинного обучения удалось выделить некоторые зависимости между собираемыми входными данными и состояниями устройств.

Таким образом, для описанной задачи реализована базовая модель системы автоматизированной поддержки среды обитания «умный дом» с оценкой точности предсказания предпочтительного состояния приборов.

Ключевые слова: *умный дом, машинное обучение, распознавание шаблонов поведения, позиционирование в помещении.*

Цитирование: Кириенко А.С., Соловьев И.П. Анализ активности человека в задаче автоматизированного управления «умным домом» // Компьютерные инструменты в образовании. 2017. № 1. С. 15–29.

1. ВВЕДЕНИЕ

Задача автоматизированного управления оборудованием в жилом помещении является важной составляющей всей обширной области деятельности, связанной с разработкой средств автоматизированной поддержки среды обитания современного человека. В этой статье мы обратимся к вопросу сбора и анализа данных, описывающих активность человека при решении конкретной задачи управления относительно небольшим помещением, скажем, частной квартирой, которое мы здесь будем называть «умным домом».

Производители бытового оборудования предлагают различные варианты объединения своих устройств в единую сеть. Хотя в этом направлении предстоит еще много рабо-

ты (нет стандартов для обмена информацией между такими устройствами), уже сейчас возможно частично автоматизировать работу окружающих приборов.

Относительно несложный и довольно распространенный в силу простоты реализации подход к автоматизированному управлению жилища в настоящее время — создание набора правил автоматизации для «ручного» управления самим жителем помещения. В то же время идея автоматизации управления предполагает совершенно иные подходы, исключая ручную конфигурацию набора правил. При реализации соответствующих решений появляется необходимость решать широкий круг задач: выделение шаблонов поведения, предсказание событий, распознавание активности, обнаружение аномалий, управление устройствами, принятие решений (о подаче различных управляющих сигналов). Во всех этих задачах хорошо зарекомендовали себя методы машинного обучения.

Реализация конкретной программы автоматизации управления обычно сталкивается с необходимостью определения источников данных для алгоритмов автоматизированного управления. В настоящее время чаще всего [1] устанавливают специальные стационарные датчики, соединенные в общую сеть. Это могут быть датчики движения, температуры, открытия дверей и т. д. Однако, в связи с широким распространением носимых устройств, появились и более доступные источники данных, которые потенциально также можно использовать. Это, например, данные, получаемые от окружающих устройств беспроводной (радио) связи (Wi-Fi и Bluetooth) и данные с датчика сердечбиения (часто бывает встроен в носимые устройства). Цель данной статьи — рассмотреть возможность использования подобных дополнительных видов данных и указать, в каких именно подсистемах это оказывается наиболее целесообразным.

2. ОБЗОР СУЩЕСТВУЮЩИХ ПОДХОДОВ

Опишем модельную задачу создания программной системы автоматизированной поддержки «умного дома». Термин «модельная» в данном случае означает то, что мы абстрагируемся от небольших и несущественных здесь для нас деталей, которые можно обнаружить в огромном разнообразии вариантов «жилищ» человека и его меньших братьев.

Введем основные понятия, которые нам потребуются в дальнейшем.

Окружение («умный дом») — небольшое жилое помещение (квартира или небольшой частный дом).

Пользователь — единственный житель помещения (задача, подразумевающая нескольких пользователей одного помещения, представляет интерес для отдельного исследования, базирующегося отчасти на данном). Мы стремимся сделать жизнь этого обитателя комфортнее за счет дополнительной автоматизированной поддержки, основанной на наблюдении за его поведением обитателя и функционированием используемого оборудования.

Рассмотрим модельную ситуацию с точки зрения подразумеваемого пользователя системы. Мы исходим из того, что его жизнь описывается множеством типичных шаблонов действий, которые потенциально поддаются анализу и могут стать объектом автоматизированной поддержки. Разумеется, набор этих шаблонов является достаточно условным. Уточнение данного набора приведет к определенной специализации системы. К примеру, для похода в душ необходимо включить свет в ванной и пустить воду определенной температуры. В момент пробуждения (а лучше немного заранее) необхо-

димом приоткрыть занавески и вскипятить чайник. Когда обитатель лежит в кровати и уже готов ко сну, необходимо выключить свет и закрыть занавески (а лучше регулировать яркость и тон света в соответствии с типом активности: ярче и белее для работы, слабее и в более теплых тонах — для отдыха). Во время приготовления ужина необходимо держать включенным локальное освещение над плитой, освещение на кухне и в гостиной, куда человек периодически ходит в ожидании следующего этапа готовки. Во время отсутствия обитателя необходимо закрыть окна, выключить подачу воды и желательно запустить не требующие вмешательства человека процессы (стирку, мойку посуды). При возвращении домой вечером после рабочего дня, усевшись на диван, обитатель с большой вероятностью захочет посмотреть любимый сериал.

Очевидно, этот список шаблонов не является полным и актуален только для конкретного человека. У других обитателей схожего окружения могут быть другие шаблоны. Более того, со временем могут формироваться новые шаблоны: к примеру, человек стал предпочитать пить кофе вместо чая по утрам. В таком случае необходимо вместо чайника включать кофеварку.

На самом верхнем уровне абстракции ключевые компоненты рассматриваемой системы показаны на рисунке 1.

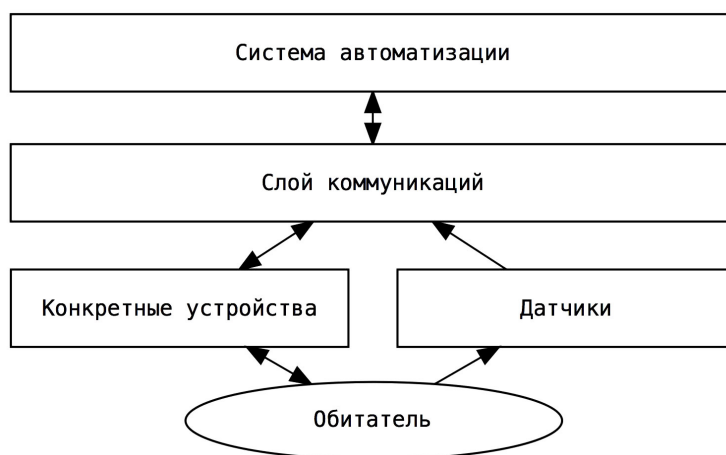


Рис. 1. Основные компоненты системы

Система автоматизации занимается сбором и анализом данных об окружении, получаемых с датчиков и устройств (данные о работе устройства, в свою очередь, могут собираться с датчиков, расположенных внутри устройства, например, датчик температуры в чайнике). На основе этих данных система должна принимать решения об управлении устройствами. Все взаимодействие осуществляется через слой коммуникаций.

Слой коммуникаций представляет собой набор программно-аппаратных средств, обеспечивающих обмен информацией между различными узлами системы «умный дом». В этой области имеется достаточно большое количество разработок, а в настоящее время свой вклад вносят крупные корпорации, такие как, например, Apple и Microsoft. Основная проблема же на текущий момент заключается в отсутствии стандартов. Из-за этого невозможно без дополнительных усилий объединить компоненты различных производителей в единую систему. Один из возможных подходов к решению этой проблемы — набирающая популярность концепция интернета вещей (Internet-of-Things). Впрочем, конкретная реализация слоя коммуникаций является отдельной задачей,

подразумевающей также необходимость разработки соответствующего оборудования и использования значительных ресурсов.

Устройства (напр. управляемые лампочки, краны, замки, чайник) и датчики (напр. датчик движения, открытия окна) непосредственно взаимодействуют с обитателями умного окружения.

В существующих исследованиях в данном направлении в качестве источников данных используются в основном специфические датчики с двумя состояниями: открытия двери, движения, включения электроприбора. Качественный набор таких данных собран в работе [6]. В исследовании [2] также использовался датчик температуры в помещении.

С таким набором датчиков можно получать достаточную для практического применения точность в задаче выявления определенных шаблонов поведения (например, приготовление завтрака). Но эти данные никак не учитывают состояние обитателя «умного» дома. Поэтому особый интерес представляет сбор и анализ таких данных (связанных непосредственно с человеком).

Подобные датчики стали широко доступны относительно недавно, с распространением носимой электроники. Это, к примеру, датчик сердцебиения и акселерометр (встроенные в браслет или часы), Bluetooth-маячки (можно использовать как носимый Bluetooth-маячок, так и Bluetooth-модуль носимого устройства в качестве приемника сигналов с окружающих Bluetooth-маячков), шагомер (основанный на том же акселерометре).

Датчик сердцебиения позволяет отличать некоторые похожие шаблоны активности (при неизменном хорошем состоянии здоровья обитателя). К примеру, прилегал ли пользователь ненадолго отдохнуть после пробежки, или же уже готов спать. Сидит он на диване и смотрит телевизор, или же прилегал вздремнуть. Определение аномалий в показаниях датчика сердцебиения с целью выявления различных заболеваний является отдельной масштабной задачей и требует гораздо более точных медицинских датчиков. В нашей же модели достаточно точности распространенных и недорогих оптических датчиков (учитывая принятое ограничение о состоянии здоровья обитателя).

Bluetooth-маячки позволяют определить местоположение обитателя в доме с некоторой погрешностью. Точность позиционирования могут улучшить данные с акселерометра. На основе данных о местоположении и перемещении можно выделить множество шаблонов поведения. К примеру, узнать, что человек идет по направлению к душе. Или что человек находится на диване. На самом деле, в качестве маячков можно использовать любые окружающие устройства беспроводной (радио) связи, что делает этот тип данных относительно легким для получения и особенно привлекательным для анализа. Для эксплуатации алгоритмов, основанных на данных такого типа, в перспективе можно вообще не докупать дополнительного оборудования, а ограничиться использованием имеющегося.

Необходимо заметить, что мы имеем дело с выделением заранее не очевидных взаимосвязей, которые, к тому же, могут изменяться с течением времени (пример про кофе и чай из начального описания). Таким образом, мы приходим к необходимости адаптации системы к изменяющимся условиям функционирования и задаче самоорганизации. Эти задачи можно решать, например, с помощью методов машинного обучения, что и можно наблюдать в опубликованных ранее работах из данной области.

Используемые в исследованиях на данную тему алгоритмы машинного обучения можно разделить на следующие категории (выделены в [7] на основе [4] и [5]):

- предсказание событий;
- выделение шаблонов поведения;
- распознавание активности;
- обнаружение аномалий;
- управление устройствами;
- принятие решений.

Рассмотрим каждую категорию более подробно.

2.1. Предсказание событий

Вся информация о деятельности человека и изменениях окружающей среды в рамках системы представлена набором состояний датчиков. Датчики могут предоставлять совершенно разную информацию, от температуры в помещении до частоты сердцебиения человека. Если система будет в состоянии предсказать показания датчиков (хотя бы частично) в определенный момент в будущем, появится возможность начать действовать заранее. К примеру, сварить кофе к пробуждению человека. Эту задачу решают с переменным успехом, используя различные алгоритмы: нейронные сети, статистически обучаемая модель, генетические алгоритмы, марковские цепи. Несмотря на кажущуюся сложность решения этой задачи, можно получить достаточную для практического применения в системе управления «умным» домом точность предсказаний на проверочной выборке.

2.2. Выделение шаблонов поведения

Как правило, действия человека в доме совсем не хаотичны. Между многими из них существует связь. Предположим, человек только что проснулся утром. Наверняка он включит свет, пойдет умыться, сядет завтракать. Важно понимать, что такие шаблоны всегда есть, но они индивидуальны для каждого человека. С помощью статистического анализа данных не составляет труда выделять такие шаблоны поведения и привязывать их к определенной реакции системы. Такой подход позволяет быстро и эффективно реагировать на типичное поведение человека. Шаблоны поведения выделяются с помощью анализа собранных данных, в частности, кластеризации.

2.3. Распознавание активности

Как было упомянуто ранее, все происходящее внутри системы описано лишь показаниями датчиков. Далеко не всегда преобразование данных датчиков в конкретный вид деятельности человека является тривиальным процессом. Для решения этой задачи используются различные виды классификаторов.

2.4. Обнаружение аномалий

Имея доступ к значительным объемам данных о происходящем в доме, можно воспользоваться статистическим анализом не только для выделения типичных шаблонов поведения, но и, напротив, для нетипичных элементов поведения. Своевременное получение такой информации поможет вовремя обнаружить, к примеру, что человек забыл выключить плиту, или же поломку оборудования в доме.

2.5. Управление устройствами

Казалось бы, для управления оборудованием, составляющим «умный» дом не нужно каких-то особенных алгоритмов. Но на деле оказывается, что это не всегда так. К примеру, системе нужно поддерживать комфортную температуру в помещении, имея только доступ к окну, обогревателю и вентилятору. Или же нужно поддерживать в комнате уровень освещенности, комфортный для чтения, учитывая естественный свет. В этих случаях совсем не очевидно, какие действия нужно производить с каждым конкретным устройством, ведь результаты их действий взаимосвязаны. Для решения этой задачи могут использоваться нейронные сети и обучение с подкреплением.

2.6. Принятие решений

Эта задача тесно связана с описанной выше. В каждый момент времени системе необходимо решать, как отвечать на изменения состояния и нужно ли отвечать вообще. Адекватные ответы системы на внешние воздействия и являются целью создания автоматизированных систем управления жилым помещением. Данная задача является одной из самых важных в рамках описываемой системы, поэтому в различных исследованиях было опробовано множество подходов, в том числе нейронные сети, обучение с подкреплением, НММ (скрытые марковские модели) и генетические алгоритмы.

3. ОПИСАНИЕ МОДЕЛИ

Поставим исследовательскую задачу в рамках содержательно описанной модели и с учетом ограничений, связанных с доступностью необходимого оборудования. Имеется фиксированный набор источников данных, описывающих состояние пользователя: распределенные по комнатам жилого помещения Bluetooth-маячки (рис. 2), с определенным интервалом сообщающие служебную информацию, на основе которой можно определить местоположение обитателя, и датчик сердцебиения, измеряющий частоту сердечных сокращений, на основе которой можно точнее распознать периоды различной активности обитателя (при отсутствии отклонений в показаниях, связанных с состоянием здоровья).



Рис. 2. Маячок, работающий по протоколу iBeacon

Данные собираются в следующем формате:

- временная метка,
- уникальный идентификатор источника данных (маячка, датчика),
- сила сигнала маячка или показание датчика (число с плавающей точкой),
- комментарий (для удобства анализа).

Пример (маячок):

```
1466771448.85298,  
EVEFD083-70A2-47C8-9837-E7B5634DF524,  
-79.0,  
WorkBeacon
```

Состояние группы устройств, являющихся объектами автоматизации (таких, как светильники, краны, бытовые приборы), складывается из состояний каждого конкретного устройства. Для простоты в нашей модели устройства имеют два состояния: включено/выключено. То есть, состояние этой группы устройств описывается вектором из 0 и 1. При необходимости, в дальнейшем модель можно усложнить.

Пользователь изменяет состояние окружения, включая и выключая устройства в соответствии со своими потребностями.

Наша задача — собрать и проанализировать данные в описанном формате, сделать вывод о возможности и целесообразности использования таких данных в различных подсистемах системы автоматизированной поддержки «умного дома», а также реализовать алгоритм работы одной из подсистем и оценить точность на собранных данных.

4. СБОР ДАННЫХ И ЭКСПЕРИМЕНТЫ

Для сбора необходимого набора данных была реализована программа для платформы iOS. Каждому датчику и устройству присвоен уникальный идентификатор. Программа сохраняет (в формате CSV) историю показаний датчиков и состояний устройств, которые изменяются жителем дома вручную (рис. 3).

Сбор данных проводился отдельно в светлое и темное время суток. Для использования полученных наработок при проектировании «умного» дома необходимо использовать датчики освещенности для подобного разбиения, так как поведение пользователя немного различается в зависимости от времени суток (как минимум в контексте управления освещением).

В однокомнатной квартире были расположены семь Bluetooth-маячков (рис. 4), причем использовались как специализированные маячки, работающие по протоколу iBeacon, так и обычные устройства с постоянно активным Bluetooth-модулем (телеприставка, весы). Местоположение маячков выбрано с учетом наполнения различных областей квартиры устройствами, которые необходимо контролировать.

5. АНАЛИЗ ДАННЫХ

С целью упрощения анализа собранных данных была реализована программа предварительной обработки данных на языке Python. Изначально показания всех датчиков и изменения состояния устройств сохраняются в виде отдельных записей. На этапе предварительной обработки эти данные группируются в пары из вектора показаний датчиков и соответствующего вектора состояний устройств вида:

```
((0, 0, -85, 0, 0, -62, -61, 86.0), (0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0))
```

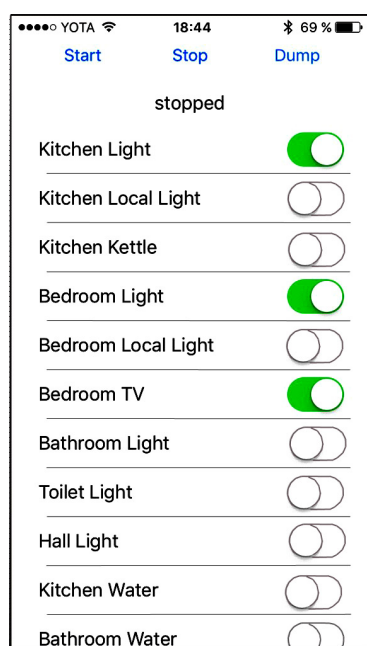


Рис. 3. Интерфейс программы для сбора данных

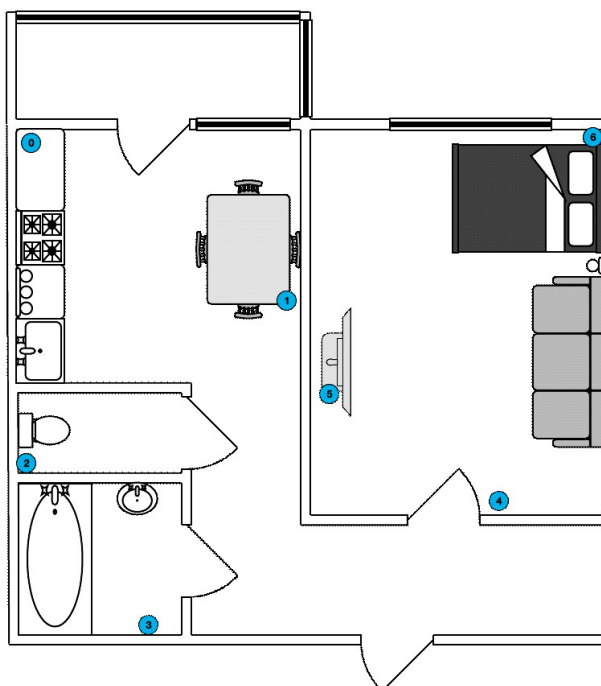


Рис. 4. Расположение Bluetooth-маячков в квартире

Вектор слева содержит показания датчиков. Первые 7 позиций содержат значения RSSI маячков (в том же порядке, что указан на схеме; RSSI — отрицательная величина, характеризующая мощность принимаемого сигнала относительно мощности источника), последняя позиция — частота сердечных сокращений с датчика сердцебиения. Вектор справа содержит значения состояния устройств (включено/выключено). Записываются состояния следующих устройств (в том же порядке):

- свет на кухне;
- локальное освещение на кухне (над плитой и столешницами);
- чайник на кухне;
- свет в спальне;
- локальное освещение в спальне (прикроватная лампа);
- телевизор в спальне;
- свет в ванной;
- свет в туалете;
- свет в прихожей;
- вода на кухне (кран в раковине);
- вода в ванной;
- душ в ванной.

В полученных данных можно проследить некоторые шаблоны поведения. К примеру, когда пользователь приходит домой, показания постепенно изменяются следующим образом (от момента приближения к двери до момента входа в помещение):

(0, 0, -86, -78, -72, -65, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -86, -78, -72, -65, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -86, -78, -60, -71, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -86, -78, -61, -71, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)

(0, 0, -86, -78, -61, -68, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -86, -78, -61, -68, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -82, -78, -61, -61, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -61, -61, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -61, -61, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -61, -59, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -58, -59, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -58, -59, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -58, -59, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -82, -77, -58, -59, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0)

Видно, что при приближении к ближайшим ко входу маячкам (3), (4) и (5) пользователь включает свет в прихожей.

Следующее действие — переход из прихожей в спальню. Маячки из ванной и туалета пропадают из видимости, сигнал от маячков спальни усиливается. Пользователь включает свет в спальне:

(0, 0, 0, 0, -56, -64, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -56, -64, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -56, -64, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -56, -64, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -56, -64, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -53, -57, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -57, 0, 86.0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -57, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -64, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -64, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -64, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -64, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -54, -64, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)

Подобное поведение в контексте освещения актуально и для других помещений. К примеру, посещение ванной комнаты:

(0, 0, -82, -78, -72, -70, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -82, -78, -72, -70, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -78, -72, -70, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -72, -70, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -72, -70, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -66, -68, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -67, -68, 0, 86.0), (0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -67, -73, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -67, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -67, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -79, -67, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -76, -67, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -76, -67, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)
 (0, 0, -83, -76, -61, -71, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)

В этом случае шаблон несколько более сложный. После включения света как правило открывается и кран с водой (для мытья рук):

(0, 0, -84, -71, -84, -60, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0)

(0, 0, -84, -71, -84, -60, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0)
 (0, 0, -84, -71, -72, -61, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0)
 (0, 0, -84, -71, -72, -61, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0)
 (0, 0, -84, -71, -72, -58, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0)
 (0, 0, -84, -71, -72, -66, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0)
 (0, 0, 0, -71, -72, -66, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0)
 (0, 0, 0, 0, -72, -66, 0, 86.0), (0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0)

По показаниям датчиков видно, что пользователь прошел дальше в помещение перед тем, как включить воду (сигнал маячка (5) из спальни слабеет, маячок (2) из туалета пропал из видимости). Стоит отметить, что в случае с краном в ванной при доступных данных имеется способ определить только момент, когда кран нужно открыть, но не симметричное событие.

Данные о сигналах маячков достаточно точны даже для определения момента, когда пользователь садится на диван и включает телевизор:

(0, 0, 0, 0, -78, -71, 0, 86.0), (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -78, -71, 0, 86.0), (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -69, 0, 86.0), (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -69, 0, 86.0), (1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -62, 0, 86.0), (1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -62, 0, 86.0), (1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -62, 0, 86.0), (1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0)
 (0, 0, 0, 0, -75, -62, 0, 86.0), (1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0)

Если проанализировать ситуацию в целом, можно заметить, что в темное время суток большую часть времени вектор состояний выглядит следующим образом:

(1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0)

либо:

(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0)

Включен свет на кухне, в спальне и, в некоторые моменты, в прихожей. По плану помещения видно, что такое состояние делает комфортным перемещение по всей квартире. Эту информацию можно использовать для скорейшего перехода в наиболее комфортное состояние сразу по приходу пользователя домой.

В целом, данные такого типа могут быть полезны во всех описанных ранее подсистемах «умного дома», особенно в подсистемах выделения шаблонов поведения, распознавания активности и принятия управляющих решений. В этих подсистемах собранные данные можно использовать и без каких-либо дополнений. Срез данных за определенный промежуток времени может быть полезен в подсистеме предсказания событий. Самый простой пример: использовать последние два значения RSSI для вычисления направления движения. Проанализированные данные могут быть полезны и в подсистеме выделения аномалий в контексте здоровья обитателя, но в этой задаче не обойтись без дополнительных данных с различных медицинских датчиков.

6. РЕАЛИЗАЦИЯ

Рассмотрим задачу принятия управляющих решений. По сравнению с другими подсистемами она накладывает меньше требований к входным данным (не нужно вручную размечать шаблоны поведения) и требует меньше усилий для реализации. Также подсистема принятия управляющих решений в данном случае сама по себе может рассмат-

риваться как система автоматизированной поддержки. Остальные подсистемы необходимы для улучшения точности и реализации дополнительных возможностей системы поддержки.

Для выполнения указанной задачи необходимо, используя набор непрерывных данных с датчиков и дискретных данных о предыдущих состояниях устройств, генерировать управляющие сигналы на основе обучающего множества имеющихся правильно сопоставленных данных (собранных в ходе обычной деятельности человека в «умном доме»). Для этой цели теоретически подходят, например, SVM (машины опорных векторов), ANN (нейронные сети) и kNN (метод k ближайших соседей). Но для kNN и основанных на нем методов необходимо хранить всю обучающую выборку и каждый раз обходить ее для поиска ближайших соседей. В нашем случае новые данные поступают несколько раз в секунду, поэтому перспективы применения подобных методов сомнительны. Представляется, что наилучшими методами в данном случае являются SVM и ANN (рис. 5).

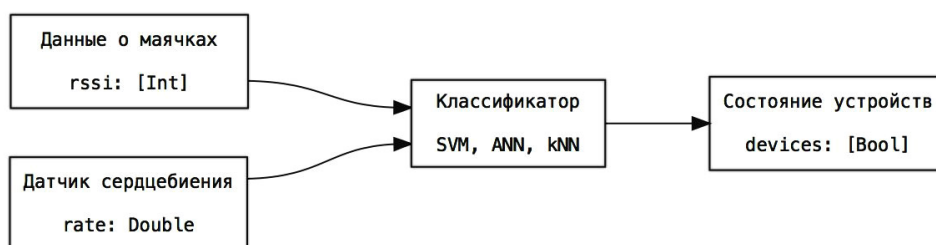


Рис. 5. Схема работы системы

В качестве конкретного алгоритма классификации был выбран SVM с радиально-базисной функцией ядра, так как он давал наилучшие результаты в решении схожих задач в более ранних исследованиях. В работе [7] автор, проанализировав существующие работы, приходит к выводу, что наилучшим образом для решения подобных задач подходят SVM и нейронные сети. В исследовании [3] авторы показали, что именно SVM с радиально-базисной функцией ядра дает наилучшие результаты при неизменном наборе датчиков (что и является нашим случаем).

Так как в выбранном подходе приходится иметь дело с большим количеством классов (более 1000, что соответствует всем вариантам состояний приборов), был выбран метод построения классификаторов One-vs-Rest (один против остальных). Этот метод предполагает обучение отдельного бинарного классификатора для каждого из множества классов, выдающего положительное значение, если образец принадлежит соответствующему классу, и отрицательное иначе.

Для реализации выбранной схемы необходимо было провести преобразование данных в удобный для классификации формат. Изначально выбранный формат хранения состояний приборов не подходит для рассматриваемой модели, поскольку он представляет собой вектор. Учитывая, что этот вектор содержит фиксированное число нулей и единиц, напрашивается естественное и однозначное преобразование в число: можно воспринимать вектор как число в двоичной записи. Этот подход и был принят. За основу реализации взята ранее упомянутая программа на языке Python. В итоге получаем векторы следующего формата:

```

0, -82, 0, -80, -81, 127, 0, 86.0, 264
0, -82, 0, -80, -81, 127, 0, 86.0, 264
  
```

0, -82, 0, -80, -81, 127, 0, 86.0, 265

0, -82, 0, -80, -81, 127, 0, 86.0, 265

Последняя компонента представляет собой номер класса, однозначно определяющий состояние контролируемых пользователем устройств. Остальные компоненты содержат показания датчиков.

С целью обучения и проверки модели были собраны данные об активности пользователя за неделю. Для работы были отобраны данные, собранные в темное время суток и представляющие активность пользователя по приходу домой в будние дни. Эксперименты проводились на наборе данных, состоящем из 10978 записей о соответствии показаний датчиков и состояний приборов в доме (в описанном выше формате).

Выборка разбивалась на обучающую и проверочную в различных соотношениях: 1:3, 1:2, 1:1, 2:1, 3:1. Для каждого разбиения обучен классификатор и измерена точность (путем сравнения ожидаемого класса, представляющего состояние устройств, и выданного классификатором).

7. ОЦЕНКА РЕЗУЛЬТАТОВ

Проверка точности реакции системы осуществляется посредством эмуляции реального окружения с помощью проверочной выборки. В систему передаются записи выборки в порядке, определяемом временными метками. Необходимо максимизировать количество правильных решений об изменении состояния окружения на проверочной выборке. Ниже (табл. 1) приведена таблица точности реализованной модели при разных размерах обучающей и проверочной выборок.

Таблица 1

Размер обучающей выборки	Размер проверочной выборки	Точность, %
2744	8234	21,2898
3659	7319	23,5278
5489	5489	22,4449
7318	3660	16,8033
8233	2745	23,9344

Мы видим, что при рациональном выборе размера обучающей выборки точность реакций достигает 21–24% (рис. 6). Это неплохой результат, учитывая значительное количество классов в выбранной модели. Он говорит о том, что первоначальная идея (использование Bluetooth-маячков и датчика сердцебиения для реализации системы автоматизированной поддержки) имеет право на жизнь.

Стоит отметить, что значения RSSI считываются довольно часто, что может негативно сказаться на энергопотреблении используемого носимого устройства. Поэтому интересно также проверить, что произойдет, если часть данных не рассматривать (смоделировать ситуацию со считыванием в 2, 3 и более раз реже). Результаты такого эксперимента приведены в табл. 2.

Можно сделать вывод, что точность модели не ухудшается на разреженных данных (рис. 7). Однако увеличится время отклика конечной системы, а также могут быть упущены из вида кратковременные события, длительность которых не превышает интервал между считываниями. Для количественного измерения подобных эффектов необходима дальнейшая адаптация модели.

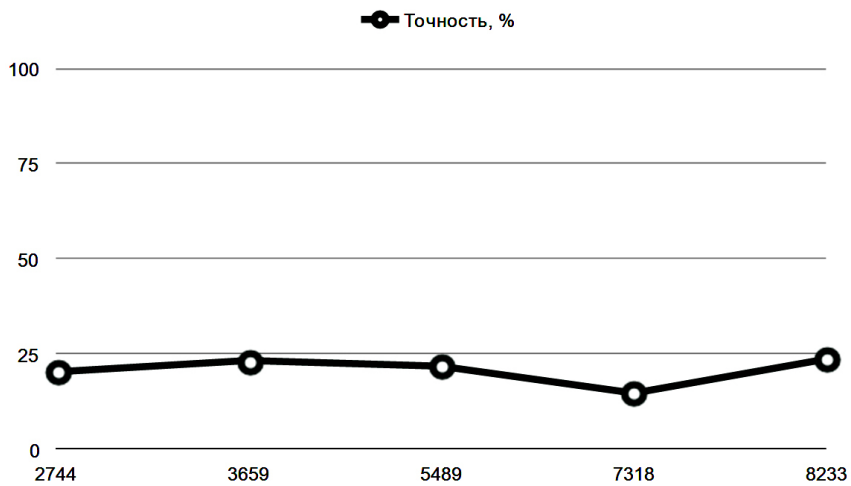


Рис. 6. Оценка точности принимаемых решений

Таблица 2

Доля использованных данных	Точность, %
1	23,5278
1/2	23,7431
1/3	23,5655
1/5	24,2486
1/10	24,0437

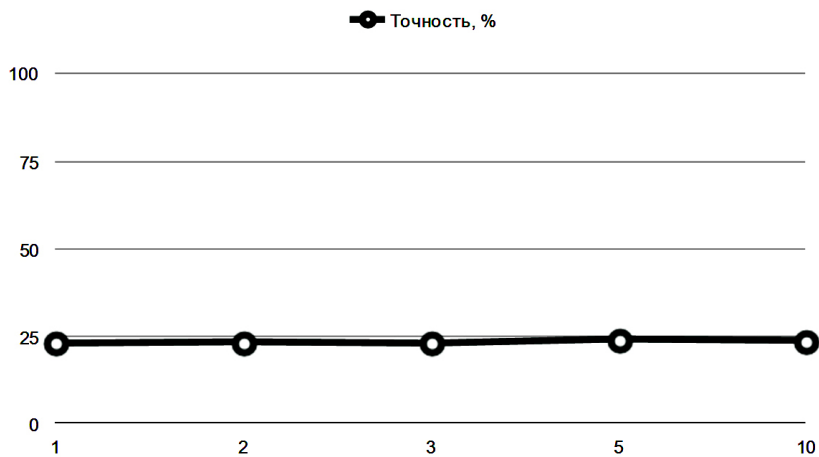


Рис. 7. Оценка точности на разреженных данных

В ходе проведенных экспериментов были обнаружены и слабые места исследованной модели

- состояние всех бытовых приборов в совокупности рассматривается как определенный класс (на самом деле, они зачастую меняются независимо друг от друга, но в зависимости от показаний датчиков);
- не предполагается какая-либо фильтрация данных (данные об RSSI содержат рез-

- кие выбросы, связанные с отсутствием сигнала в определенный небольшой промежуток времени);
- модель предполагает однозначное соответствие показаний датчиков и состояний приборов (на практике выяснилось, что однозначное соответствие есть скорее между показаниями датчиков и событиями включения/выключения приборов, а не между показаниями и состоянием всех приборов в данный момент, что можно заметить на примере света в прихожей, который с одной стороны переключается достаточно независимо от других приборов, а с другой стороны входит в шаблон «пользователь пришел домой»).

8. ЗАКЛЮЧЕНИЕ

В результате работы был собран и проанализирован набор данных об индикаторах уровня принимаемого сигнала (RSSI), показаниях датчика сердцебиения и пользовательской активности в «умном» доме (в контексте переключения приборов). Реализована базовая модель системы автоматизированной поддержки, работающая на основе этих данных. Получена точность предсказания предпочтительного состояния приборов приблизительно до 24%.

Количество собранных на данный момент данных с датчика сердцебиения пока не позволяет сделать однозначные выводы о целесообразности использования в системе поддержки данных о частоте сердечных сокращений. В то же время, показана принципиальная возможность построения системы автоматизированной поддержки (в частности, подсистемы принятия управляющих решений), работающей на основе данных о силе сигнала маячков (в качестве маячков могут выступать любые приборы, излучающие радиосигналы, в том числе имеющие Bluetooth и Wi-Fi передатчики; в данной работе используются Bluetooth-маячки и постоянно включенные приборы с Bluetooth-модулем).

Выделены основные обнаруженные на практике слабые места простейшей модели, понимание которых можно использовать в процессе построения и реализации других, более сложных моделей.

Список литературы

1. Кириенко А.С., Соловьев И.П. Подходы к использованию машинного обучения для решения задачи управления «умным» домом // Современные технологии в теории и практике программирования. 2016.
2. Babakura A. et al. HMM-Based Decision Model for Smart Home Environment // International Journal of Smart Home. 2014. Т. 8. №1.
3. Choi S., Kim E., Oh S. Human behavior prediction for smart homes using deep learning // RO-MAN, 2013 IEEE. 2013. С. 173–179.
4. Kulkarni S.R., Lugosi G., Venkatesh S.S. Learning pattern classification-a survey // IEEE Transactions on Information Theory. 1998. Т. 44. № 6. С. 2178–2206.
5. Langley P., Simon H.A. Applications of machine learning and rule induction // Communications of the ACM. 1995. Т. 38. № 11. С. 54–64.
6. Munguia Tapia E. Activity recognition in the home setting using simple and ubiquitous sensors : дис. Massachusetts Institute of Technology, 2003.
7. Stenudd S. Using machine learning in the adaptive control of a smart environment. 2010.

Поступила в редакцию 20.01.2017, окончательный вариант — 15.02.2017.

Computer tools in education, 2017

№ 1: 15–29

<http://ipo.spb.ru/journal>

HUMAN BEHAVIOUR ANALYSIS IN CONTEXT OF SMART ENVIRONMENT AUTOMATION

Kirienko A.S.¹, Soloviev I.P.¹

¹SPbSU, Saint-Petersburg, Russia

Abstract

Basic approaches to the automated control of smart environment problem are considered. A new method of solution based on the information about the signal strength of wireless broadcasting devices (considered as beacons) and the heart rate sensor readings is introduced. The model environment is an apartment with Bluetooth beacons; the inhabitant has a heart rate sensor and a RSSI measuring device. An application collecting beacons RSSI and heart rate measurements during the inhabitant activity in the model environment has been implemented. The application provides the user interface to display current state of devices which are objects of automated control. By using machine learning methods the relation between the collected data and device states is obtained. Thus, an automated smart environment control model has been implemented. The estimation of the accuracy of prediction of the devices preferable state has been obtained.

Keywords: *smart environment, machine learning, activity recognition, indoor geofencing.*

Citation: Kirienko, A. & Soloviev, I., 2017. "Analiz aktivnosti cheloveka v zadache avtomatizirovannogo upravleniya «umnym domom»" ["Human Behaviour Analysis in Context of Smart Environment Automation"]. *Computer tools in education*, no. 1, pp. 15–29.

Received 20.01.2017, the final version — 15.02.2017.

Artem S. Kirienko, Postgraduate at the computer science department of St. Petersburg State University; 198504 Saint-Petersburg, Botanicheskaya st., 70, b. 4, r. 717, artem.kirienko@gmail.com

Igor P. Soloviev, PhD, Associate Professor at Comp. Sci. Dept. of St. Petersburg State University, i.soloviev@spbu.ru

**Кириенко Артем Сергеевич,
аспирант кафедры информатики
математико-механического факультета
СПбГУ; 198504 Санкт-Петербург,
ул. Ботаническая, 70/4, ком. 717,
artem.kirienko@gmail.com**

**Соловьев Игорь Павлович,
кандидат физико-математических наук,
доцент кафедры информатики
математико-механического факультета
СПбГУ,
i.soloviev@spbu.ru**

©

Наши авторы, 2017.

Our authors, 2017.