

sistema-obnaruzheniya-anomalnogo-povedeniya-vychislitelnyh-protssesov-mikroyadernyh-operatsionnyh-sistem (accessed: 20.12.2020). (In Russ.)

24. A system for detecting anomalous behavior of telephone subscribers. Anomaly detection systems. URL: <http://www.academia.edu/8525607> (accessed: 20.12.2020). (In Russ.)

*Received 20.12.2020*

## ТЕХНОЛОГИИ ЦИФРОВОЙ ЭКОНОМИКИ

УДК 004.852

### МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ЧЕЛОВЕЧЕСКОЙ АКТИВНОСТИ С ИСПОЛЬЗОВАНИЕМ ДАТЧИКОВ ОКРУЖАЮЩЕЙ СРЕДЫ

*Трошин А.В.*

*Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ  
E-mail: a.v.troshin77@yandex.ru*

Распознавание человеческой активности представляет собой быстро развивающуюся область исследований, целью которой является определение типа поведения людей на основе собираемых данных. Системы распознавания человеческой активности находят широкое применение в медицине, спорте, производстве и многих других сферах. Данные для систем распознавания могут быть получены при помощи видеонаблюдения, а также с использованием различного рода датчиков. Датчики окружающей среды обладают рядом преимуществ: простота, дешевизна и широкое применение в системах «умного дома». Для обработки данных в системах распознавания часто используются методы машинного обучения. Данная статья посвящена приложению различных методов машинного обучения к распознаванию человеческой активности на основе данных, полученных с датчиков окружающей среды «умного дома».

**Ключевые слова:** *методы машинного обучения, распознавание человеческой активности, датчики окружающей среды*

#### Введение

Распознавание человеческой активности (Human Activity Recognition – HAR) является быстро развивающейся областью исследований, целью которой является идентификация действий людей на основе собираемых данных [1–3]. Распознавание человеческой активности находит широкое применение в медицине: диагностика ряда заболеваний, уход за пожилыми людьми и инвалидами; в производстве: отслеживание действий и реакции персонала в различных ситуациях; в спорте: оценка уровня подготовки спортсменов.

В настоящее время существует два основных направления развития систем HAR, определяемых типом используемых данных [1–3]: системы на основе видеонаблюдения и системы на основе данных, собираемых с различного вида датчиков. В системах первого типа данные представляют собой видеоизображения человеческой деятельности, которые необходимо отнести к заранее определенным классам. В системах второго типа данными являются показания датчиков, закрепляемых на теле человека или устанавливаемых

в окружающей человека среде, на основе этих данных требуется классифицировать текущую человеческую активность.

Несмотря на то что в области распознавания изображений были получены впечатляющие результаты, внедрение систем HAR на основе видеонаблюдения на практике может привести к ряду затруднений. Во-первых, для надежной работы таких систем необходимо получение изображений высокого разрешения и четкости, что может потребовать использования достаточно дорогостоящих видеокамер. Во-вторых, мониторинг активности в нескольких помещениях, а также наличие различных закрывающих обзор объектов делает необходимым использование нескольких видеокамер, что значительно удорожает внедрение систем данного типа. И в-третьих, распознавание активности с помощью видеонаблюдения в личных помещениях, что актуально, например, при наблюдении за пожилыми людьми, часто может быть неприемлемо из-за нарушения частной жизни и раскрытия персональных данных [1–3].

Большое распространение умных устройств, таких как смартфоны и фитнес-браслеты, которые уже снабжены встроенными датчиками,

а также достаточно низкая стоимость большинства датчиков в сравнении с видеокамерами высокого разрешения, делают использование систем HAR на основе показаний датчиков экономически весьма привлекательными. Помимо этого использование таких систем может вызвать меньше вопросов, связанных с раскрытием информации о частной жизни, так как собираемые данные могут быть ограничены только определенным набором, по которому идентификация личности будет сильно затруднена. Вместе с тем следует также отметить, что системы HAR на основе датчиков в настоящее время имеют более низкую надежность распознавания в сравнении с системами видеонаблюдения [1–3].

### **Датчики распознавания активности**

Датчики, используемые для распознавания активности, можно разделить на четыре группы [1–3]:

- датчики носимых устройств;
- датчики параметров окружающей среды;
- датчики идентификации объектов;
- неспециализированные датчики.

К датчикам носимых устройств относят датчики, которые часто встроены в персональные устройства, переносимые на теле человека, такие как смартфоны, смарт-часы, фитнес-браслеты, предметы одежды. Устройства с такими датчиками обычно закрепляют на руке, ноге или поясе. Наиболее широко распространенными датчиками данного типа являются акселерометры и гироскопы. Акселерометр – это датчик, измеряющий ускорение тела по трем осям несколько десятков раз в секунду. Гироскоп позволяет фиксировать изменения угловой скорости по трем осям с частотой несколько десятков герц. Носимые датчики позволяют получить непосредственную информацию о движениях тела человека, что делает их незаменимыми при распознавании индивидуальной динамической активности человека, например, ходьбы, бега, поднятия тяжестей [1].

Датчики, измеряющие параметры окружающей среды, обычно встраивают в различного рода устройства Internet вещей: лампы освещения, системы контроля микроклимата помещений, умную бытовую технику, системы безопасности. К датчикам данного типа относят: термометры, датчики освещения, датчики приближения и присутствия, счетчики воды и энергопотребления. В отличие от носимых датчиков, они подходят под распознавание сложной бытовой активности сразу нескольких индивидов [1].

Датчики идентификации объектов служат для определения взаимодействия человека с каким-

либо предметом обихода. В качестве подобных датчиков наиболее широко применяют считыватели радио-идентификационных меток (Radio Frequency Identification – RFID). Уникальные RFID-метки закрепляются на бытовых предметах, таких как кружка, тарелка, зубная щетка и т. п., после чего взаимодействие с предметом фиксируется считывателем меток, который должен быть закреплен на теле человека. Датчики идентификации объектов хорошо подходят для распознавания комплексной активности, которая включает взаимодействие с несколькими бытовыми предметами, снабженными метками, например, приготовление пищи и личную гигиену. Однако использование таких датчиков сопряжено с рядом неудобств, поскольку требует от человека постоянного ношения считывателя меток и крепления меток на множестве бытовых предметов [1–3].

К неспециализированным датчикам можно отнести устройства, которые помимо своего прямого назначения имеют дополнительные возможности, позволяющие их использовать для распознавания активности. Примером таких датчиков являются встроенные микрофоны смартфонов и бытовых аудиосистем. Встроенные микрофоны способны улавливать очень широкий диапазон звуковых частот, обработка которого позволяет распознавать разнообразную человеческую активность: ходьбу, относительное перемещение в пространстве, открытие и закрытие дверей, работу на компьютере. Также к подобным датчикам можно отнести беспроводные устройства, способные фиксировать изменения картины электромагнитного поля в помещении [1].

### **Методы распознавания активности**

Получаемые с датчиков данные, как правило, представляют собой хронологические последовательности нескольких параметров. Например, для носимых систем это могут быть показания гироскопа для трех пространственных осей на определенном временном этапе. Основной задачей обработки таких данных является определение типа активности на заданных временных интервалах. Таким образом, задача обработки данных HAR относится к задачам классификации многопараметрических временных рядов. Для решения данной задачи широко применяются методы статистики и машинного обучения [1–3].

При использовании статистических методов вычисляются математические ожидания и дисперсии сигналов датчиков на временных интервалах, после чего их соотносят с заранее определенными

эталонными значениями для заданных типов активностей. Такой подход, однако, пригоден только для весьма ограниченного набора простых динамических активностей [1]. В более сложных методах данного типа используется преобразование временных последовательностей в частотную область, далее составляются двумерные спектральные картины на заданных интервалах, обработка которых производится методами классификации изображений. Такие методы хорошо зарекомендовали себя при анализе данных с носимых и встраиваемых датчиков, однако не подходят для обработки больших объемов данных с множества датчиков, что актуально для систем, использующих данные датчиков окружающей среды [1–3].

Методы машинного обучения нашли широкое применение для задач классификации активностей. Наиболее часто для распознавания применяется обучение с учителем (supervised learning). При данном подходе с помощью заранее аннотированного набора данных производится обучение выбранного алгоритма, который затем используется для классификации вновь собранных данных. Преимуществом методов машинного обучения являются хорошая адаптируемость под разнообразный характер данных и большое число активностей, а также отсутствие необходимости в сложной предварительной обработке сигналов. Основным недостатком методов машинного обучения заключается в необходимости большого аннотированного набора данных для обучения алгоритмов [1–3].

Традиционно выделяют классические и ансамблевые методы машинного обучения, а также методы, основанные на использовании искусственных нейронных сетей [4; 5].

### Классические методы машинного обучения

К классическим методам машинного обучения часто относят [4; 5]:

- алгоритм  $k$ -ближайших соседей  $kNN$ ;
- дерево решений DesTree;
- метод опорных векторов SVM;
- наивный байесовский классификатор NB.

Алгоритм  $k$ -ближайших соседей ( $k$ -Nearest Neighbors –  $kNN$ ) является одним из простейших непараметрических алгоритмов машинного обучения, в котором классификация входного вектора данных производится на основе простого большинства классов  $k$ -ближайших аннотированных векторов из обучающего набора данных. Например, при  $k = 1$  входному вектору просто присваивается класс ближайшего соседнего век-

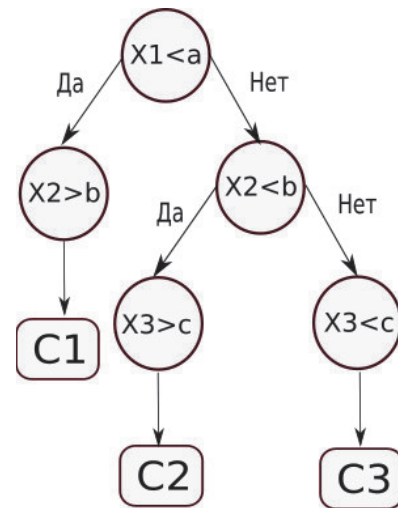


Рисунок 1. Пример дерева решений

тора. В качестве расстояния до соседних векторов обычно берется расстояние в евклидовом пространстве.

Также в большинстве практических реализаций  $kNN$  используется взвешенная оценка вклада соседей в итоговый класс: чем ближе сосед, тем выше его вес при определении класса. Преимуществом алгоритма  $kNN$  является простота реализации, а главный недостаток заключается в плохой устойчивости при большом наборе входных параметров, поскольку линейный рост параметров приводит к экспоненциальному росту расстояния между соседними векторами, что значительно затрудняет классификацию. Данная проблема, известная как «проклятие размерности», характерна для многих других классических методов машинного обучения.

Дерево решений (Decision Tree) представляет собой алгоритм классификации, в основе которого лежит последовательное бинарное разделение всего пространства выбора. Дерево решений представляет собой ациклический граф – см. рисунок 1, в узлах которого производится принятие решений путем сравнения входных данных, а ребра являются результатами сравнения (да/нет). Замыкающие вершины (листья) соответствуют итоговым классам  $C_1, C_2, \dots$  [4].

Получение дерева решений производится рекурсивным бинарным делением пространства обучающих данных. Критерием выбора наилучшего дерева является минимизация целевой функции, определяющей совокупную точность классификации обучающих данных. Часто для задач классификации в качестве целевой функции используется критерий Джини [4].

В методе опорных векторов (Support Vector Machine – SVM) для классификации используется

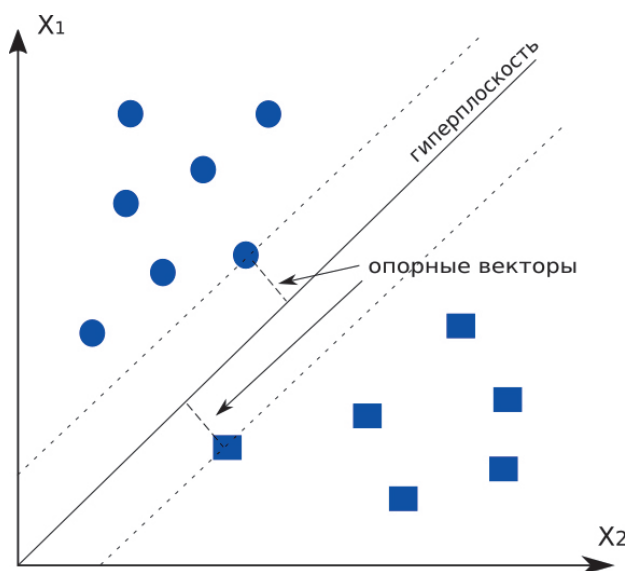


Рисунок 2. Принцип SVM

разделение пространства векторов при помощи гиперплоскостей. Класс входного вектора определяется исходя из области, которая ограничена гиперплоскостями. С целью уменьшения ошибки классификации подбираются гиперплоскости с максимальным расстоянием до ближайших векторов. Для этого от ближайшего вектора (точки) проводится перпендикуляр до гиперплоскости. Точки, лежащие на перпендикуляре, получили название опорных векторов (см. рисунок 2).

Классификация с наименьшей ошибкой обеспечивается при максимальной длине опорных векторов. В случае, когда точное разделение классов с помощью гиперплоскостей невозможно, их положение выбирается так, чтобы минимизировать ошибку классификации (так называемая мягкая классификация). Обучение алгоритма SVM производится путем подбора разделяющих гиперплоскостей для обучающего набора данных, при этом применяются различные методы линейной алгебры [4; 5].

Наивный байесовский классификатор (Naive Bayes – NB) основан на выборе наиболее вероятного класса ( $C_i$ ) для вектора входных данных  $X = (x_1, x_2, \dots, x_n)$ , используя предварительные «наивные» представления о данных, основанные на известной теореме Байеса об условной вероятности [4; 5]:

$$P(C_i | X) = \frac{P(C_i)P(X | C_i)}{P(X)}. \quad (1)$$

Как следует из выражения (1) для определения класса входных данных, необходимо знать предварительную вероятность класса  $P(C_i)$ , которая вычисляется по частоте класса в обучающих данных, а также условную вероятность входных

данных при заданном классе  $P(X | C_i)$ , при «наивном» допущении ее можно вычислить как произведение независимых вероятностей для каждой переменной  $x_j$  входного вектора по обучающим данным для заданного класса.

Вероятность входных данных  $P(X)$  является постоянной для всех классов и поэтому в расчетах обычно не используется. Таким образом, определение наиболее вероятного класса  $C'$  при наивном байесовском подходе сводится к вычислению:

$$C' = \arg \max_i \{P(C_i) \prod_i P(x_i | C_i)\}. \quad (2)$$

Простейший байесовский наивный классификатор основан на предположении о равновероятном распределении данных для всех классов, что на практике встречается достаточно редко. В том случае, если вероятность классов зависит от нормального распределения входных данных, используется гауссовский наивный байесовский классификатор (Gaussian NB). Другими распространенными вариантами являются наивные байесовские классификаторы для мультиномиального распределения (Multinomial NB) и распределения Бернулли (Bernoulli NB) [4; 5].

### Ансамбльные методы машинного обучения

В основе ансамбльных методов машинного обучения лежит одновременное использование множества простых алгоритмов классификации. Итоговое решение о принадлежности к определенному классу входного вектора признаков выносится на основе большинства вынесенных решений отдельными алгоритмами [4; 6].

К ансамбльным методам относят:

- бэггинг-классификация Bagging;
- лес случайных деревьев RForest;
- экстремальные деревья ExTree;
- градиентный бустинг GBoost.

В бэггинг-классификации (Bagging Classifier) производится одновременное обучение нескольких деревьев решений. Обучающие данные разбиваются на несколько наборов, сохраняющих репрезентативность исходных данных. Проводится построение дерева решений для каждого набора. Классификация входных данных производится всеми деревьями, после чего в качестве итогового класса выбирается наиболее частое решение.

Данный подход позволяет повысить стабильность и точность классификации, но требует большей вычислительной мощности и памяти. Основным параметром данного метода является



число деревьев решений, чаще всего оно подбирается итерационно, пока точность классификации не перестает расти. Бэггинг-классификация может также использоваться и для других методов машинного обучения, например, искусственных нейронных сетей [4; 6].

Лес случайных деревьев (Random Forest) представляет собой модификацию бэггинга для деревьев решений. При классическом бэггинге часто получаются структурно сходны деревья, что приводит к сильной корреляции их решений, это негативно сказывается на обобщающей способности алгоритма классификации. В лесе случайных деревьев для построения деревьев решений используются не все признаки, а только случайно выбранная их часть.

Таким образом, снижается точность классификации отдельного дерева, но при этом повышаются обобщающая способность и устойчивость общей классификации. Недостатком данного метода является значительно большее требуемое количество деревьев для достижения заданной точности в сравнении с простым бэггингом [4].

Метод экстремальных деревьев (Extra Trees) является дальнейшей модификацией леса случайных деревьев. Основное его отличие состоит в том, что обучение случайных деревьев решений производится на всех обучающих данных, кроме этого все точки принятия решения в дереве выбираются случайно, а не рекурсивно. Это позволяет, во-первых, повысить скорость обучения, во-вторых, снизить влияние на классификацию различных случайных факторов [4].

Суть метода градиентного бустинга (Gradient Boosting) состоит в том, что вклад в общее решение отдельных классификаторов учитывается с весом, обратно пропорциональным ошибке классификации. В конечном счете влияние на итоговую классификацию «хороших» классификаторов повышается, а «плохих» снижается.

Ошибка классификации и соответствующий вес рассчитываются после добавления каждого нового дерева, таким образом, происходит градиентный спуск или бустинг в направлении снижения общей ошибки классификации [4].

### Искусственные нейронные сети

Для задач классификации хронологических последовательностей используются те же типы искусственных нейронных сетей, что и для задач регрессии [4; 5]:

- многослойный перцептрон (MLP);
- сверточная нейронная сеть (CNN);
- рекуррентные нейронные сети (RNN).

Подробно реализация данных нейросетей рассмотрена в работе [7]. Основное отличие нейронных сетей для классификации заключается в особой функции активации выходного слоя нейронов. В задачах классификации двух классов обычно используется функция, называемая логистическим сигмоидом (sigmoid) [4; 5]:

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \quad (3)$$

Значения сигмоид-функции находятся в пределах от 0 до 1, что позволяет их интерпретировать как вероятности одного из классов. При числе классов больше двух для активации применяется softmax-функция [4; 5]:

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}}. \quad (4)$$

Данная функция позволяет нормализовать значения выходных нейронов и интерпретировать их как распределения вероятностей между классами.

Поскольку обучение искусственных нейронных сетей производится методом градиентного спуска и его модификациями, для этого необходима дифференцируемая функция оценки ошибки классификации. Для задач классификации наиболее часто в качестве функции оценки ошибки используется перекрестная энтропия (Cross Entropy –  $CE$ ). В задачах для двух классов перекрестная энтропия имеет вид [5]:

$$CE = -(y \log(p) + (1 - y) \log(1 - p)), \quad (5)$$

где  $p$  – вероятность одного из классов, предсказанная алгоритмом;  $y$  – показатель правильности классификации (0 или 1).

Для классификации с числом классов более двух перекрестная энтропия имеет вид [5]:

$$CE = -\sum_{i=1}^K y_i \log(p_i), \quad (6)$$

где  $p_i$  – вероятность  $i$ -го класса;  $y_i$  – индикатор правильности классификации данного класса.

### Оценка качества классификации

Наиболее часто для оценки качества классификации кого-либо метода на практике применяется точность (accuracy) классификации, определяемая как отношение числа корректно классифицированных экземпляров данных к их общему числу. Также в качестве меры оценки можно использовать частоту ошибок (error rate): долю ошибочно классифицированных данных в общем числе данных [4].

Таблица 1. Пример матрицы спутывания

Истинные классы	Предсказанные классы		
	$C_1$	$C_2$	$C_3$
$C_1$	5	1	0
$C_2$	1	4	1
$C_3$	2	0	4

Однако приведенные оценки качества являются интегральными и не позволяют получить представление, насколько точно данный метод классифицирует данные разных классов, что особенно актуально при неравномерном распределении данных из разных классов. В этом случае для оценки точности классификации разных классов широко применяется так называемая матрица спутывания (confusion matrix). Матрица спутывания представляет собой квадратную матрицу, названия строк и столбцов в которой содержат обозначения всех возможных классов данных. Пример такой матрицы спутывания для трех классов представлен в таблице 1 [4].

Значение на пересечении  $i$ -й строки и  $j$ -й колонки таблицы обозначает, сколько данных  $i$ -го класса было отнесено алгоритмом к  $j$ -му классу. Таким образом, значения на главной диагонали матрицы спутывания показывают число корректно классифицированных случаев. В представленном примере четыре экземпляра класса  $C_3$  были определены корректно, однако два экземпляра данного класса были отнесены к классу  $C_1$  [4]. Точность классификации для отдельного класса можно получить, разделив соответствующее значение на главной диагонали матрицы спутывания на общее число экземпляров этого класса в данных.

Точность классификации отдельного класса складывается из четырех факторов [4]:

- правильного отнесения экземпляров данного класса к этому классу ( $TP$  – True Positive);
- неверного отнесения экземпляров других классов к данному классу ( $FP$  – False Positive);
- неверного отнесения экземпляров этого класса к другим классам ( $FN$  – False Negative);
- правильного отнесения экземпляров других классов к другим классам ( $TN$  – True Negative).

Используя значения данных факторов, вычисленные на основе матрицы спутывания, определим точность классификации отдельного класса, согласно [4], в следующем виде:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}. \quad (7)$$

Как уже было сказано, точность является интегральной характеристикой качества работы

алгоритма классификации. Для получения более детальной информации о работе алгоритма по отношению к отдельным классам часто используются дополнительные параметры оценки: меткость и чувствительность.

Меткость (precision) показывает отношение правильных классификаций класса к общему числу случаев классификаций, отнесенных к этому классу [4]:

$$Pr = \frac{TP}{TP + FP}. \quad (8)$$

Чем выше меткость алгоритма по отношению к определенному классу, тем большее число данных будет отнесено к этому классу.

Чувствительность (recall) показывает отношение правильных классификаций класса к общему числу экземпляров этого класса [4]:

$$R = \frac{TP}{TP + FN}. \quad (9)$$

Чем выше чувствительность алгоритма по отношению к определенному классу, тем большее число экземпляров этого класса будет правильно классифицировано как экземпляры этого класса.

Меткость и чувствительность алгоритма часто находятся в обратной зависимости: высокая меткость алгоритма, как правило, ведет к меньшей чувствительности и наоборот. В связи с этим применяют комбинированный параметр:  $F_1$ -метрику, объединяющую оценки меткости и чувствительности [4; 5]:

$$F_1 = 2 \frac{Pr \times R}{Pr + R}. \quad (10)$$

Чем ближе значение  $F_1$  к единице, тем более сбалансированным является соотношение меткости и чувствительности алгоритма по отношению к данному классу.

Анализ отдельных параметров классов может быть затруднен при их большом числе. Интегральную оценку сбалансированности классификации по всем классам можно получить, используя корреляционный коэффициент Мэтью (Matthew Correlation Coefficient – MCC), который определяется по следующей формуле [4]:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \quad (11)$$

Здесь MCC может принимать значения от +1 (наилучшая классификация) до -1 (наихудшая классификация). Как правило, алгоритм классификации требуется для работы на постоянно поступающих новых данных, но для оценки качества его работы нельзя использовать обучающие

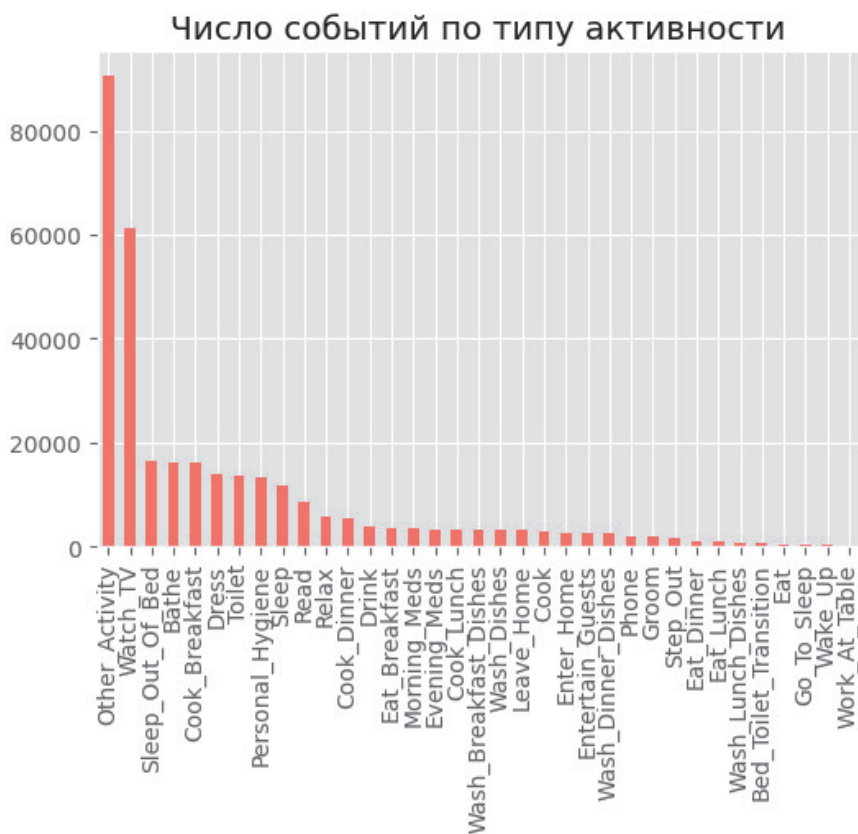


Рисунок 3. Пример распределения активностей резидентов

данные. Поэтому их всех доступных данных для итоговой оценки точности классификации выделяют определенную часть (20–30 %), остальные 70–80 % могут быть использованы для обучения алгоритма [4–6].

### Описание данных и постановка задачи

Для исследования возможностей различных методов машинного обучения по распознаванию человеческой активности на основе данных с датчиков окружающей среды был выбран открытый набор данных, собранный исследователями Вашингтонского университета [8]. Данные представляют собой непрерывные показания с датчиков «умного дома», установленных в домах резидентов-добровольцев. Показания собирались непрерывно в течение приблизительно двух месяцев с помощью протокола ZigBee с датчиков движения, открытия/закрытия дверей, управления освещением и других.

На основе необработанных данных формировался аннотированный вектор атрибутов событий в каждый момент времени, который включает в себя такие параметры, как:

- час дня в локальном времени (last Sensor Event Hour);
- отсчет в сек от полуночи (last Sensor Event Seconds);

- порядковый день недели (last Sensor Event Day Of Week);

- время в сек от момента последнего срабатывания датчика кухни (Sensor ElTime-Kitchen);

- время в сек от момента последнего срабатывания датчика прихожей (Sensor ElTime-Hall) и др.

Всего в векторе записано 36 различных атрибутов событий. Каждый вектор аннотирован соответствующей активностью резидента в данный момент времени, например:

- приготовление обеда (Cook\_Dinner);
- просмотр телевизора (Watch\_TV);
- сон (Sleep);
- утренний прием медицинских препаратов (Morning\_Med);
- нераспознанная активность (Other\_Activity).

Всего в наборе данных каждого резидента может насчитываться от 31 до 36 различных активностей. Следует отметить весьма значительную неравномерность событий для разных активностей. Пример активностей резидента с обозначением csh101 представлен на рисунке 3.

Для всех резидентов от 25 до 40 % событий составляет нераспознанная активность, в то время как активности вида прием лекарств или мытье посуды могут занимать менее 0,01 %, что значительно затрудняет их распознавание.

Цель данного исследования была сформулирована следующим образом: провести оценку

Таблица 2. Оценки методов классификации

Метод	csh101		csh103		csh107	
	Acc	MCC	Acc	MCC	Acc	MCC
kNN(5)	0,71	0,64	0,86	0,82	0,75	0,65
DesTree	0,56	0,44	0,81	0,75	0,76	0,65
SVM	0,75	0,68	0,89	0,86	0,83	0,75
NB	0,67	0,56	0,72	0,63	0,57	0,27
Bagging	0,63	0,54	0,79	0,73	0,81	0,72
RForest	0,73	0,66	0,89	0,85	0,85	0,79
ExTree	0,77	0,7	0,91	0,88	0,86	0,79
GBoost	0,8	0,75	0,89	0,86	0,85	0,79
MLP	0,76	0,7	0,9	0,86	0,83	0,77
CNN	0,76	0,7	0,89	0,86	0,84	0,77

различных методов машинного обучения для предсказания текущей активности резидента «умного дома», используя показания датчиков окружающей среды.

### Анализ полученных результатов

Пример полученных оценок точности классификации и коэффициент Метью, рассмотренных методов машинного обучения для трех резидентов, обозначенных как csh101, csh103 и csh107, представлены в таблице 2.

Расчеты проводились для активностей, доля событий которых превышает 2 % от общего числа событий, чтобы обеспечить статистическую достоверность результатов. Моделирование методов классического и ансамбльного машинного обучения проводилось с использованием пакета `scikit-learn` [9]. Моделирование искусственных нейронных сетей проводилось в пакете `TensorFlow` [10]. Использовалась MLP с двумя скрытыми слоями из 64 и 32 нейронов, а также CNN с одним сверточным слоем, состоящим из 32 фильтров, и размером ядра в два нейрона без слоя пуллинга, с дополнительным скрытым слоем из 32 нейронов. В обеих нейросетях для активации всех слоев, кроме выходного, использовался линейный выпрямитель (ReLU). Число нейронов в выходном слое соответствовало количеству классифицируемых активностей, для активации нейронов выходного слоя использовалась `softmax` функция. Исходные коды моделей всех рассмотренных методов машинного обучения представлены в репозитории [11].

Как видно из приведенных оценок, наилучшие результаты по точности и сбалансированности классификации показали алгоритмы экс-

тремальных деревьев (ExTree) и градиентного бустинга (GBoost). Точность классификации довольно значительно различается по резидентам, что может быть обусловлено индивидуальными привычками. Также необходимо отметить, что точность классификации имеет большой разброс и по активностям: для продолжительных во времени активностей (сон, просмотр телевидения, работа вне дома) точность классификации может достигать 80–90 %, в то время как для кратковременных активностей (прием лекарств, мытье посуды, одевание) точность классификации составляет 20–30 %. Данные особенности следует учитывать при внедрении систем распознавания активностей на основе датчиков окружающей среды.

### Заключение

В работе рассмотрены методы машинного обучения для классификации человеческой активности на основе данных датчиков окружающей среды. Показано, что системы, основанные на использовании датчиков окружающей среды, имеют ряд преимуществ, в том числе:

- использование готовой инфраструктуры «умного дома»;
- защита персональных данных и др.

В ходе проведенного моделирования на наборе данных реального «умного дома» было продемонстрировано, что наилучшую точность и сбалансированность классификации обеспечивают методы экстремальных деревьев и градиентного бустинга. Также показано, что системы распознавания на основе датчиков окружающей среды целесообразно применять для распознавания человеческих активностей, занимающих длительные интервалы времени в пределах «умного дома».

### Литература

1. Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities / K. Chen [et al.]. URL: <https://arxiv.org/abs/2001.07416> (дата обращения: 01.10.2020).
2. Deep learning for sensor-based activity recognition: A survey / J. Wang [et al.] // *Pattern Recognition Letters*. 2019. Vol. 119. P. 3–11. DOI: <https://doi.org/10.1016/j.patrec.2018.02.010>
3. Hussain Z., Sheng M., Zhang W.E. Different approaches for human activity recognition: A survey. URL: <https://arxiv.org/abs/1906.05074> (дата обращения: 01.10.2020).
4. Marsland S. *Machine Learning: An Algorithmic Perspective*; 2nd ed. Boca Raton: CRC Press, 2015. 430 p.



5. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. Cambridge: MIT Press, 2016. 800 p.
6. Kuhn M., Johnson K. *Applied Predictive Modeling*. Berlin: Springer, 2013. 600 p.
7. Трошин А.В. Машинное обучение для прогнозирования трафика в сети LTE // *Инфокоммуникационные технологии*. 2019. Т. 14, № 4. С. 400–406.
8. Human activity recognition from continuous ambient sensor data set. URL: [https://archive.ics.uci.edu/ml/datasets/Human+Activity+Reco](https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+from+Continuous+Ambient+Sensor+Data)
9. Scikit-learn. URL: <https://www.scikit-learn.org> (дата обращения: 01.10.2020).
10. TensorFlow. URL: <https://www.tensorflow.org> (дата обращения: 01.10.2020).
11. Troshin A. Human activity recognition with ambient sensors. URL: [https://github.com/avtroshin77/har\\_ambient\\_sensors](https://github.com/avtroshin77/har_ambient_sensors) (дата обращения: 04.10.2020).

*Получено 20.10.2020*

**Трошин Александр Викторович**, к.т.н., доцент кафедры сетей и систем связи Поволжского государственного университета телекоммуникаций и информатики. 443010, Российская Федерация, г. Самара, ул. Л. Толстого, 23. Тел. +7 846 339-11-26. E-mail: [a.v.troshin77@yandex.ru](mailto:a.v.troshin77@yandex.ru)

## MACHINE LEARNING METHODS FOR HUMAN ACTIVITY RECOGNITION USING AMBIENT SENSORS

*Troshin A.V.*

*Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation  
E-mail: [a.v.troshin77@yandex.ru](mailto:a.v.troshin77@yandex.ru)*

Human activity recognition is a fast growing research field with the goal to identify human behavior based on collected data. Human activity recognition systems have many applications in healthcare, sport, manufacturing and other spheres. Data for activity recognition may be collected from video-cameras or various sensors. Ambient sensors have a lot of advantages such as simplicity, inexpensiveness and wide application in smart-houses. Machine learning methods are often used for pattern activity recognition in sensor data. This work presents an application and comparison of some classical, ensemble and neural network machine learning methods for human activity recognition using smart-house ambient sensor data.

**Keywords:** *machine learning methods, human activity recognition, ambient sensors*

**DOI:** 10.18469/ikt.2021.19.1.12

**Troshin Alexander Victotovich**, Povolzhskiy State University of Telecommunications and Informatics, 23, L. Tolstoy Street, Samara, 443010, Russian Federation; Assistant Professor of Networks and Systems of Telecommunication Department, PhD in Technical Sciences. Tel. +7 846 339-11-26. E-mail: [a.v.troshin77@yandex.ru](mailto:a.v.troshin77@yandex.ru)

### References

1. Chen K. et al. Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities. URL: <https://arxiv.org/abs/2001.07416> (accessed: 01.10.2020).
2. Wang J. et al. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 2019, vol. 119, pp. 3–11. DOI: <https://doi.org/10.1016/j.patrec.2018.02.010>
3. Hussain Z., Sheng M., Zhang W.E. Different approaches for human activity recognition: A survey. URL: <https://arxiv.org/abs/1906.05074> (accessed: 01.10.2020).
4. Marsland S. *Machine Learning: An Algorithmic Perspective*; 2nd ed. Boca Raton: CRC Press, 2015, 430 p.
5. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. Cambridge: MIT Press, 2016, 800 p.
6. Kuhn M., Johnson K. *Applied Predictive Modeling*. Berlin: Springer, 2013, 600 p.
7. Troshin A.V. Machine Learning for LTE Traffic Prediction. *Infokommunikacionnye tehnologii*, 2019, vol. 14, no. 4, pp. 400–406. (In Russ.)

8. Human activity recognition from continuous ambient sensor data set. URL: <https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+from+Continuous+Ambient+Sensor+Data> (accessed: 01.10.2020).
9. Scikit-learn. URL: <https://www.scikit-learn.org> (accessed: 01.10.2020).
10. TensorFlow. URL: <https://www.tensorflow.org> (accessed: 01.10.2020).
11. Troshin A. Human activity recognition with ambient sensors. URL: [https://github.com/avtroshin77/har\\_ambient\\_sensors](https://github.com/avtroshin77/har_ambient_sensors) (accessed: 04.10.2020).

*Received 20.10.2020*

УДК 004.42

## АВТОМАТИЗАЦИЯ СОЗДАНИЯ ОТЧЕТОВ ПО ЭФФЕКТИВНОСТИ РАБОТЫ РЕКЛАМНЫХ КАМПАНИЙ

*Москалев М.Г., Газизов Т.Т.*

*Томский государственный педагогический университет, Томск, РФ*

*E-mail: moskalev@tspu.edu.ru*

В статье рассматривается необходимость разработки веб-приложения для автоматизации создания отчетов об эффективности работы рекламных кампаний из нескольких рекламных сервисов. Проводится сравнение имеющихся веб-сервисов, позволяющих решить проблему объединения нескольких отчетов с описанием недостатков данных сервисов. Описывается созданное веб-приложение, его функциональность и пример использования. Разработанное веб-приложение позволяет объединять отчеты и проводить сравнение как в онлайн-формате, так и с возможностью загрузки сгенерированного отчета. Результатом работы созданного веб-приложения является сгенерированный отчет в формате Excel с возможностью построения графиков на основе веб-приложения. Предоставляемый пользователю отчет содержит в себе информацию о числе показов кампании, числе переходов по объявлению, рассчитывает отношение числа переходов к числу показов, рассчитывает среднюю цену перехода, количество переходов, достигших цели, стоимость достижения цели, а также общее число потраченных средств на рекламную кампанию, что является удобным для проведения дальнейшего анализа и расчета эффективности проведенной рекламной кампании.

**Ключевые слова:** *эффективность рекламных кампаний, разработка веб-приложения, автоматизация, генерация отчетов, веб-разработка, РНР*

Актуальность работы связана с активным развитием рынка рекламы в сети Интернет. Так, согласно отчету, опубликованному Ассоциацией коммуникационных агентств России [1], в первом полугодии 2019 г. объем денежных средств, потраченных на рекламу, составил 227 млрд рублей. Из них более 100 миллиардов было потрачено на рекламу в сети Интернет. Одним из важных этапов работы по размещению контекстной рекламы является написание отчетов об эффективности ее работы.

Цель работы заключается в том, чтобы автоматизировать создание отчетов, это позволит сократить время, которое тратится на их написание, а также исключит возможные ошибки при написании отчетов путем использования механизма автоматизации, что позволит оптимизировать расходы рекламной кампании в сети Интернет. Разрабатываемое WEB-приложение на основе имеющихся данных о работе рекламной кампании должно генерировать отчет о показателях ее эффективности.

Созданный отчет должен включать в себя данные о числе показов кампании, количестве переходов по объявлению, рассчитывать отношение числа переходов к числу показов, рассчитывать среднюю цену перехода, количество переходов, достигших цели, стоимость достижения цели, а также общее количество потраченных средств на рекламную кампанию.

Разработанное WEB-приложение должно генерировать сводную таблицу в формате «.XLSX», удобном для просмотра, а также HTML-таблицу с возможностью сортировки по значению.

Существуют готовые сервисы, позволяющие решить поставленную задачу. Среди них можно выделить онлайн-сервисы «ReportKey», «Marilyn», «Яндекс.Директ». Сервис «ReportKey» является онлайн-сервисом генерации отчетов [2]. Сервис предоставляет пользователю возможность создавать отчеты из нескольких каналов, в том числе из систем контекстной рекламы «Яндекс.Директ» и «Google AdWords». Позволяет генерировать текстовые отчеты в форматах