

МАГІСТЕРСЬКА РОБОТА

МР.КІ-15.00.00.000 ПЗ

Група КІм-24-1

Регнер Володимир

2025

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу
Факультет інформаційних технологій
Кафедра комп'ютерних систем і мереж

Регнер Володимир Володимирович
(прізвище, ім'я, по батькові)

УДК 004.415
(індекс)

МАГІСТЕРСЬКА РОБОТА

Тема: *Використання байєсового класифікатора для інтелектуального модуля ідентифікації дій людини в контексті технологій Smart House*
(назва роботи)

Комп'ютерна інженерія
(назва освітньої програми)

123 – комп'ютерна інженерія
(шифр і назва спеціальності)

В. В. Регнер
(підпис, ініціали та прізвище здобувача освітнього ступеня)

Науковий керівник *Заячук Ярослав Іванович, к.т.н., доцент*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту
Завідувач кафедри

проф. С. І. Мельничук
(посада) (підпис) (дата) (ініціали та прізвище)

Рецензент

(посада) (підпис) (дата) (ініціали та прізвище)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

6. Консультанти по дипломній роботі, із зазначенням розділів роботи, що стосуються їх

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
<i>нормоконтроль</i>	<i>О. В. Мойсеєнко</i>		

7. Дата видачі завдання 12.03.2025

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1	<i>Аналіз технологій та рішень, які забезпечують інтелектуальність розумних будинків</i>	<i>12.03.2025-31.05.2025</i>	<i>Виконано</i>
2	<i>Класифікація видів діяльності людини у розумному будинку</i>	<i>01.06.2025-31.07.2025</i>	<i>Виконано</i>
3	<i>Інтелектуальний модуль системи розумного будинку на основі класифікації видів діяльності людини</i>	<i>01.08.2025-30.09.2025</i>	<i>Виконано</i>
4	<i>Класифікація активності та оцінка ефективності системи</i>	<i>01.10.2025-15.11.2025</i>	<i>Виконано</i>
5	<i>Оформлення пояснювальної записки</i>	<i>16.11.2025-10.12.2025</i>	<i>Виконано</i>

Студент-магістр

_____ (підпис)

Регнер В. В.

Керівник роботи

_____ (підпис)

Заячук Я. І.

АНОТАЦІЯ

Запропоновано інтелектуальну систему підтримки прийняття рішень, яка керує роботою компонентів розумного будинку, спираючись на класифікацію різновидів людської активності.

Така система здатна підвищити рівень автоматизованого управління та здатність оперативно реагувати на індивідуальні потреби мешканців. Це досягається завдяки створенню інтелектуального механізму класифікації видів діяльності, що фіксуються у середовищі розумного будинку.

РОЗУМНИЙ БУДИНОК, МАШИННЕ НАВЧАННЯ, ЕКСПЕРТНА СИСТЕМА

SUMMARY

An intelligent decision-support system has been proposed to manage the operation of smart-home components based on the classification of various types of human activity.

The practical value of the obtained results lies in the system's ability to enhance the level of automated control and improve responsiveness to the residents' needs. This is achieved through the development of an intelligent mechanism for classifying activities recorded within the smart-home environment.

SMART HOME, MACHINE LEARNING, EXPERT SYSTEM

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	4
ВСТУП.....	5
1 АНАЛІЗ ТЕХНОЛОГІЙ ТА РІШЕНЬ, ЯКІ ЗАБЕЗПЕЧУЮТЬ ІНТЕЛЕКТУАЛЬНІСТЬ РОЗУМНИХ БУДИНКІВ	7
1.1 Архітектура системи IoT	7
1.2 Інтелектуальні технології в розумних будинках	9
1.3 Інтелектуальні рішення ідентифікації діяльності людини в IoT.....	13
1.4 Постановка завдання.....	17
2 КЛАСИФІКАЦІЯ ВИДІВ ДІЯЛЬНОСТІ ЛЮДИНИ У РОЗУМНОМУ БУДИНКУ	19
2.1 Модель розумного будинку для збору даних.....	19
2.2 Опрацювання набору даних	23
2.3 Класифікація видів діяльності людини.....	25
2.4 Висновок до розділу.....	40
3 ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ СИСТЕМИ РОЗУМНОГО БУДИНКУ НА ОСНОВІ КЛАСИФІКАЦІЇ ВИДІВ ДІЯЛЬНОСТІ ЛЮДИНИ	42
3.1 Структура інтелектуальної системи.....	42
3.2 Збір та опрацювання даних від сенсорів.....	46
3.3 Класифікація активності та оцінка ефективності системи	49
3.4 Висновок до розділу.....	52
ВИСНОВКИ.....	53
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	54

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

AI – Artificial Intelligence

API – Application Programming Interface

AR – Augmented Reality

BLE – Bluetooth Low Energy

BSON – Binary JavaScript Object Notation

CART – Classification and regression tree

CHAP – Challenge Handshake Authentication Protocol

CNN – Convolutional neural network

CRISP-DM – Cross-Industry Standard Process for Data Mining

GPS – Global Positioning System

GPU – Graphics Processing Unit

HMM – Hidden Markov Model

IoT – Internet of Things

JSON – JavaScript Object Notation

KNN – K-Nearest Neighbor

LSTM – Long short-term memory

ML – Machine Learning

MSE – Mean Squared Error

MQTT – Message Queue Telemetry Transport

NFC – Near Field Communication

REST – Representational State Transfer

RSSI – Received Signal Strength Indication

SVM – Support Vector Machine

TSDB – Time series database

ВСТУП

У сучасних умовах людство постійно намагається по-новому осмислювати використання наявних об'єктів і технологій, прагнучи таким чином знаходити шляхи розв'язання нових завдань, що постають перед суспільством. Саме в подібному руслі виникла та була реалізована концепція Інтернету речей. Її суть полягає в тому, що інтелектуальні можливості вбудовуються у фізичні предмети, а між ними формується середовище для обміну даними, завдяки чому відбувається автоматизація значної кількості щоденних процесів у промисловості, побуті, медицині та багатьох інших сферах людської діяльності.

Одним із практичних напрямів застосування Інтернету речей є розпізнавання моделей поведінки людини в просторах, обладнаних пристроями з розширеним технічним функціоналом, що дозволяє під'єднати їх до мережі Інтернет і, відповідно, зробити ці пристрої «розумними». Прикладом подібного середовища виступає так званий розумний дім. У ньому різноманітні датчики можуть визначити, наприклад, момент, коли мешканець переходить до спальні, щоб відпочити. Коли система фіксує таку активність за допомогою сенсорів, вона застосовує алгоритм класифікації та відносить подію до категорії «сон». Після цього розумний комплекс здатен автоматично подбати про мешканця, наприклад, вимкнувши опалення або змінюючи інші параметри без будь-якої участі людини. У результаті подібна моніторингова система збирає інформацію з датчиків, визначає події, реагує на них та підтримує користувача, коли виникає потреба.

Актуальність теми дослідження. Водночас сучасні рішення для домашньої автоматизації досі не забезпечують повної автономності таких процесів, оскільки значною мірою ґрунтуються лише на заздалегідь визначених правилах формату «якщо – то», що обмежує гнучкість і можливості адаптації системи. Тому розробка інтелектуальної системи, яка забезпечить більший рівень автоматизації процесів у розумному будинку, є актуальним завданням.

Метою роботи є проектування інтелектуальної системи класифікації та ідентифікації діяльності людини для розумного будинку.

Об'єктом дослідження є процес класифікації діяльності людини інтелектуальною системою розумного будинку.

Предметом дослідження є інтелектуальна система класифікації діяльності людини у розумному будинку.

Методи дослідження. У процесі виконання роботи застосовувалися такі методи: аналітичні та математичні методи досліджень; теорія графів і множин.

Наукова новизна: отримала подальший розвиток модель розпізнавання людської активності, створена на основі наївного байєсового класифікатора. Ця модель дає змогу визначати найімовірнішу дію, яку виконує мешканець інтелектуального житла, спираючись на сукупність релевантних ознак. Такий підхід зробив можливим чітке формалізування ключових фаз процесу ідентифікації поведінкових патернів людини в умовах функціонування розумного середовища.

Практичне значення: представлена інтелектуальна система здатна збільшити рівень автоматизованого реагування на запити та звички мешканців smart-житла шляхом проектування механізму класифікації різновидів діяльності, які фіксуються в інтелектуальних домах і потребують оперативної інтерпретації для подальшого керування.

Апробації результатів роботи. Результати досліджень роботи оприлюднені на Всеукраїнській науково-практичній конференції молодих вчених та студентів «Інформаційні технології в освіті, техніці та промисловості» - 2025.

Структура і обсяг роботи. Магістерська робота складається зі вступу, 3 розділів, висновків і списку літератури, що включає 50 найменувань. Основна частина роботи викладена на 59 сторінках машинописного тексту. Робота містить 14 рисунків, 2 таблиці та 1 додаток на 2 сторінках.

1 АНАЛІЗ ТЕХНОЛОГІЙ ТА РІШЕНЬ, ЯКІ ЗАБЕЗПЕЧУЮТЬ ІНТЕЛЕКТУАЛЬНІСТЬ РОЗУМНИХ БУДИНКІВ

1.1 Архітектура системи IoT

Архітектуру Інтернету речей зазвичай описують як багаторівневу систему, у якій окремі функціональні шари взаємодіють між собою, створюючи єдиний простір для комунікації між фізичними об'єктами та інформаційним середовищем. До таких шарів належать рівень застосунків, сенсорний рівень, рівень комунікаційних шлюзів і мережевих модулів, а також серверна або хмарна частина, що разом забезпечують безперервний обмін даними між реальним світом і цифровою інфраструктурою (рис. 1.1). На базовому рівні розташовані різні сфери використання IoT – житлові простори, транспортні системи, розумне агровиробництво, офісні рішення, промислові підприємства, урбаністичні розумні комплекси і розумні будівлі. Вище знаходиться шар фізичних сенсорів та об'єктів, що демонструє типові групи пристроїв у структурі IoT [1].

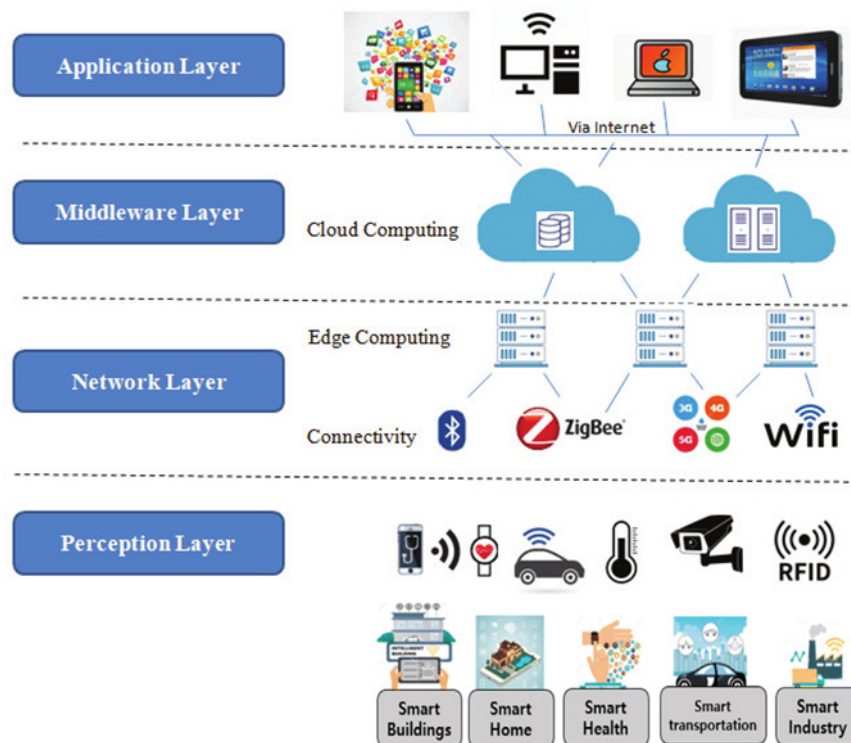


Рисунок 1.1 – Рівні IoT

Значний набір сенсорів конструктивно розрахований для прямого доступу до Інтернету та обміну інформацією з серверними й хмарними застосунками. Сюди належать охоронні відеокамери, терморегулятори, датчики пожежної безпеки, інтелектуальні лічильники ресурсів та інші модулі контролю. Інша категорія сенсорів функціонує через проміжне обладнання – шлюзові вузли або периферійні модулі туманних обчислень. Такі елементи мають вищу продуктивність, дозволяють об'єднувати багато сенсорів у локальній мережі та пов'язують їх із загальною IoT-системою через протоколи ZigBee, різні варіанти 802.15.4, Bluetooth, енергозберігаючі версії Wi-Fi чи інші бездротові рішення, що застосовуються для малопотужних пристроїв.

Функції шлюзів охоплюють широкий спектр завдань: перетворення між протоколами, фільтрацію первинних потоків, буферизацію даних, локальну аналітику та первинне виявлення подій. У сучасній екосистемі IoT значну роль відіграють смартфони, які фактично можуть діяти як сенсорні пристрої, локальні вузли для пересилання інформації або інтерфейси взаємодії з користувачем. Завдяки наявності великої кількості вбудованих модулів – акселерометрів, гіроскопів, датчиків освітленості, тиску, температури, магнітних полів, GPS-модулів, барометрів, а також камер і мікрофонів – смартфони здатні формувати багатий набір даних для різних застосунків. Програмний доступ до цих сенсорів сприяє інтеграції телефону у хмарні сервіси та робить його важливою частиною мережі IoT. Багато застосунків з доповненою реальністю або сервісів геолокації комбінують інформацію з кількох сенсорів для побудови точніших моделей поведінки користувача чи оточення.

Верхній шар архітектури уособлює хмарні технології, що виконують масштабну обробку даних і забезпечують роботу віддалених серверних систем. У хмарному середовищі агрегуються різні потоки інформації, здійснюється аналітична обробка, оптимізація процесів і знаходження закономірностей між численними подіями. Дані, що надходять від сенсорів, можуть оброблятися у форматі потоків у реальному часі або зберігатися для подальшого аналізу. Хмарна інфраструктура забезпечує додаткові функції: масштабовані репозиторії, системи

візуалізації, генерацію звітів, сервіси керування доступом, виявлення аномалій і модулі безпеки, критично важливі для захисту IoT-середовища від загроз та атак.

Алгоритми машинного навчання та штучного інтелекту часто реалізуються саме у хмарі, де є доступ до великих масивів даних та високої продуктивності. Завдяки цьому моделі можуть досягати значної точності прогнозування, аналізувати поведінку об'єктів, оцінювати ризики та формувати рекомендації. У новіших реалізаціях поширюється також підхід edge-AI, коли частина обчислень переноситься на периферію, що зменшує затримку і навантаження на мережу.

Основний принцип руху даних у системах IoT полягає в тому, що сенсорні модулі вимірюють параметри середовища і передають показники до програм, які реагують на ці значення. Такі програми можуть бути розподіленими між хмарою, шлюзами та окремими периферійними рішеннями. Це дозволяє забезпечити швидкість реакції, оптимальне використання ресурсів і надійність системи.

У підсумку архітектура Інтернету речей дозволяє здійснювати постійний моніторинг, аналіз і коригування стану фізичних процесів. Вона вирізняється гнучкістю, масштабованістю, економічністю, великим набором функціональних можливостей та здатністю підтримувати широкий спектр рішень у міській інфраструктурі, енергетиці, промисловій автоматизації, медицині, транспорті, агросекторі, логістиці та багатьох інших галузях, де доступ до точних даних і високий рівень автоматизації є критично важливими [1].

1.2 Інтелектуальні технології в розумних будинках

Штучні нейронні мережі в сучасному вигляді отримали початковий імпульс розвитку у другій половині XX століття. У 1958 році Френк Розенблатт представив концепцію, що отримала назву «одношаровий перцептрон» – структуру, яка заклала фундамент для подальшого становлення всієї галузі. Перцептрон став однією з перших моделей, що намагалася формалізувати принципи роботи біологічних нервових клітин. Ідея спиралася на уявлення про мозок як складну мережу нейронів, що обмінюються сигналами через синаптичні

контакти. Синапси передають імпульси за допомогою хімічних речовин, і реакція виникає тільки за достатнього комбінованого впливу. Саме такий механізм став натхненням для створення математичного аналога - штучного нейрону, який обробляє сигнали на основі їхніх числових ваг та порогового значення [1].

У штучній нейронній мережі кожен елемент може отримувати дані від кількох джерел і генерувати власний вихідний сигнал. Завдяки операціям зважування, підсумовування та порівняння з порогом модель визначає, чи «активувати» нейрон. Таким чином формується елементарний обчислювальний блок, з якого будуються складні мережеві архітектури. Персептрон реалізує ці принципи через множення вхідних сигналів на вагові коефіцієнти та подальше їх об'єднання. У разі перевищення порогового значення система видає одиничний результат, у протилежному випадку – нуль. Завдяки цій простій логіці модель здатна виконувати базові завдання класифікації.

Мережа складається з кількох типів шарів. Перший із них, вхідний, лише отримує дані та передає їх далі. Приховані шари виконують математичні перетворення, надаючи змогу виділяти нові ознаки, що не містяться у сирих даних. Кожен нейрон у прихованому шарі формує зважену суму, після чого застосовується функція активації – сигмоїдна, ReLU або інша, залежно від структури моделі. Вихідний шар перетворює результати останнього прихованого рівня на класи, ймовірності або числові значення, необхідні для розв'язання конкретного завдання. Модель з одним прихованим шаром часто називають неглибокою, на відміну від багат шарових конструкцій, що використовуються в сучасних системах штучного інтелекту.

З часом окремою галуззю машинного навчання стало глибинне навчання. На відміну від традиційних моделей, глибинні мережі складаються з великої кількості рівнів, що виконують послідовні перетворення, поступово формуючи складні відображення [1]. На кожному шарі система навчається виділяти нові властивості: від простих лінійних комбінацій до високорівневих абстракцій. Такий підхід забезпечує високу ефективність у сферах, де обсяг інформації величезний – комп'ютерний зір, прогнозування часових рядів, обробка сигналів,

Інтернет речей. Зростання популярності глибинного навчання пов'язане також з появою графічних процесорів і хмарних обчислень, які зробили тренування великих мереж значно швидшим. Багатошарові перцептрони є фундаментальною формою таких структур, хоча нині існують значно складніші моделі – згорткові, рекурентні, трансформери.

Метод *K*-найближчих сусідів (*k*-nearest neighbor method, KNN) належить до інтуїтивно зрозумілих підходів. Він не вимагає побудови складної математичної моделі: достатньо мати набір навчальних прикладів, кожен з яких описаний певною кількістю атрибутів. Уявити їх можна як точки в багатовимірному просторі. Новий об'єкт порівнюється з наявними прикладами за допомогою певної метрики – найчастіше евклідової, мангеттенської або косинусної. Алгоритм визначає, які *K* сусідів розташовані найближче, і на основі їхньої більшості присвоює об'єкту відповідний клас. При $K=1$ результат залежить лише від одного найбільш близького прикладу, що робить метод чутливим до шуму, тому збільшення *K* дозволяє отримати більш стабільні результати. KNN часто застосовується у пошуку схожих зображень, рекомендаційних системах та задачах визначення аномалій [1].

Support Vector Machine (SVM) пропонують інший підхід до класифікації. Вони намагаються знайти гіперплощину, яка максимально розділяє точки різних класів. Перевагою є здатність знаходити оптимальну межу, яка забезпечує найбільший запас між класами. Якщо дані важко розділити в їх початковому вигляді, застосовується нелінійне перетворення у простір більшої розмірності. Використовуючи ядрові функції – радіальні, поліноміальні чи сигмоїдні – SVM можуть моделювати складні поверхні розділення. Опорні вектори є ключовими точками, що визначають форму гіперплощини. Хоча час навчання може бути значним, модель має хорошу стійкість до перенавчання. SVM часто використовують у системах розпізнавання тексту, біоінформатиці та задачах з невеликим або середнім обсягом даних [1].

Random Forest базується на концепції ансамблевих методів, де багато моделей поєднуються задля покращення точності. Ліс складається з набору дерев

рішень, кожне з яких отримує випадкову підмножину ознак і фрагмент даних. Така випадковість зменшує кореляцію між деревами та забезпечує високу здатність до узагальнення. Кожне дерево видає свій прогноз, а підсумковий результат визначається голосуванням або усередненням, залежно від завдання. Важливими параметрами, що впливають на якість, є кількість дерев, глибина кожного дерева, розмір випадкової підмножини ознак. Алгоритм добре працює у задачах класифікації, регресії, визначення важливості ознак і навіть у подіях з великим рівнем шуму. Модель має високу стійкість, оскільки не покладається на рішення однієї структури, а комбінує багато варіантів [1].

Classification and Regression Tree (CART) є одним із найвідоміших алгоритмів побудови дерев рішень. Він розбиває дані на основі порогових значень ознак, прагнучи отримати максимально однорідні підмножини. Мірою неоднорідності виступає індекс Джині, який оцінює ступінь чистоти. Побудова дерева продовжується доти, доки не буде досягнуто максимальної чистоти або встановлених обмежень. Оскільки надмірно глибокі дерева можуть перенавчатися, застосовується «обрізка», яке видаляє слабкі гілки. CART універсальний: підходить як для числових, так і для категоріальних даних, а також легко інтерпретується – що робить його корисним у банківській сфері, медицині, маркетинговій аналітиці [1].

Наївний бейєсів класифікатор – є одним із найпростіших і водночас ефективних статистичних методів для класифікації. Він ґрунтується на теоремі Байєса та припущенні умовної незалежності ознак: кожна характеристика об'єкта впливає на вибір класу незалежно від інших. Завдяки цьому модель обчислює ймовірність належності до кожної категорії як добуток умовних ймовірностей ознак, що дозволяє швидко отримувати результат і для великих наборів даних.

Під час навчання алгоритм оцінює базові ймовірності класів та ймовірності появи кожної ознаки в межах цих класів. Для неперервних даних зазвичай використовується нормальний розподіл (Gaussian Naive Bayes), а для текстів або частотних даних – мультиноміальний чи бернуллівський варіанти. Класифікація відбувається шляхом вибору класу з максимальною апостеріорною ймовірністю.

Хоча припущення про незалежність ознак є спрощенням, яке не завжди відповідає реальним даним, наївний баєсів класифікатор часто демонструє високу точність, особливо у задачах аналізу текстів, фільтрації спаму, класифікації документів та медичній діагностиці. Модель проста, інтерпретована та стійка до «зайвих» ознак, що робить її корисною в інформаційному пошуку, маркетинговій аналітиці та інших сферах, де важливі швидкість і надійність результатів.

Приховані марковські моделі (Hidden Markov Model, HMM) використовуються для опису процесів, що мають приховані стани, недоступні прямому спостереженню. Основною ідеєю є те, що послідовність видимих подій залежить від внутрішніх станів, які формують імовірнісні розподіли. Модель визначає, з якою ймовірністю система переходить від одного прихованого стану до іншого, а також як кожен стан впливає на спостережувані результати. Це робить HMM надзвичайно корисними у задачах аналізу часових рядів, обробки звукових сигналів і розпізнавання мови. Існують три класичні задачі HMM: обчислення ймовірності послідовності, визначення найімовірнішої траєкторії прихованих станів та оцінювання параметрів моделі. Вони розв'язуються алгоритмами Вітербі, Баум-Велша та прямого-зворотного проходу [1].

HMM є частиною більш широкого класу графічних моделей, які дозволяють моделювати залежності між змінними у структурованій формі. Вони широко застосовуються там, де важливо описати взаємодію між неповністю спостережуваними процесами: біологічна еволюція, мовні моделі, фінансові часові ряди, системи прогнозування станів об'єктів [1].

1.3 Інтелектуальні рішення ідентифікації діяльності людини в IoT

Автоматичне розпізнавання людської активності за допомогою IoT-систем є однією з ключових задач сучасних досліджень у галузі розумних середовищ та електронного охорони здоров'я. Для її вирішення використовуються різноманітні методи, серед яких скінченні автомати (Finite State Machine, FSM), зважені скінченні автомати (Weighted Finite Automata, WFA), приховані ланцюги Маркова

(Hidden Markov Models, HMM), наївний баєсівський класифікатор, Support Vector Machine (SVM), Random Forest та бутстрепова агрегація. Ці підходи дозволяють моделювати поведінку людини на рівні послідовності дій, ймовірнісних залежностей між ними або за допомогою експертних правил.

У дослідженні Karvonen та Kleuko [1,2] запропоновано рішення, що базується на знаннях експертів. Для класифікації використовувалися лише дані від чотирьох двійкових датчиків. Автори відзначають, що традиційні алгоритми машинного навчання, зокрема SVM, не враховують часові залежності між видами діяльності, що є критично важливим для коректного прогнозування послідовностей. Запропонована експертна система функціонує як скінченний автомат і забезпечує точність класифікації 81,3%. Основною перевагою цього підходу є прозорість виведених правил, що дозволяє користувачам інтерпретувати логіку класифікації. Такий аспект особливо важливий у сфері електронного охорони здоров'я, де користувачі повинні розуміти процес прийняття рішень системою [1]. Серед недоліків – зниження точності при зміні щоденних звичок мешканця розумного будинку.

Окремим викликом є характер набору даних, який був зібраний у контрольованому середовищі. Тестовані особи могли поводитися неприродно через усвідомлення спостереження, а також поведінка могла змінюватися або перериватися, що призводить до некоректної класифікації послідовностей дій. Крім того, обмежений обсяг даних знижує надійність навчання моделей [1].

Підхід Lago та Inoue [3] поєднує ймовірнісну та описову моделі. Ймовірнісна частина базується на HMM, де кожна діяльність відображена окремим станом, а переходи між станами моделюють послідовність дій. Ймовірності виходу розраховувалися за допомогою нейронної мережі з урахуванням середньої тривалості кожної з 24 активностей. Такий підхід дозволяє оцінити ймовірність залишання у поточній діяльності або переходу до іншої. Описова частина моделі спирається на словесні описи дій, реалізовані у класах Java, що дозволяє визначати початок і кінець активностей. Загальна точність гібридної системи склала 45%. Проблеми виникали при класифікації короткотривалих дій, таких як

«покласти прання в пральну машину» або «відвідати SmartLab». Однак перевага описової частини полягає у здатності розпізнавати дії, які не зустрічалися у навчальному наборі. Ефективність моделі обмежена у разі неспрацювання датчиків, які позначають завершення активності [1].

У дослідженні Salomon та Tirnauce [4] запропоновано напівкерований метод на основі WFA та регулярних виразів. Для підвищення точності додатково використовувалися дані датчиків підлоги, що дозволяло визначати положення користувача у приміщенні. Для кожного сегмента дня навчався окремий автомат, а ваги переходів визначалися на основі частоти траєкторій між станами. Регулярні вирази описували активність датчиків та їх порядок для кожної діяльності. Комбінування ймовірностей переходів і регулярних виразів дозволяло класифікувати активності з точністю 90,65%. Основні помилки були пов'язані з неточним визначенням початку та кінця активностей через людський фактор та шум у даних [1].

У роботі Jimenez та Seco [5] застосовано наївний баєсівський класифікатор для оцінки виконаних дій. Метод аналізував дані всіх чотирьох джерел, досягнувши точності 60,5%.

У дослідженні [6] досягнуто точності 94,0% на навчальних даних та лише 47,0% на тестових наборах, застосувавши FilteredClassifier з Weka Tool Weka 3. Така суттєва різниця між навчанням та тестуванням свідчить про переобладнання моделі класифікації. Для навчання використовували всі доступні сенсорні дані. FilteredClassifier реалізовувався із стандартним фільтром StringToWordVector та Random Forests як базовим класифікатором. Фільтр Weka проводив попередню обробку, включаючи очищення та підготовку набору даних для класифікації.

Авторами роботи [7] запропоновано методику на основі Міжгалузевого стандартного процесу для інтелектуальних даних (CRISP-DM). За [8], вона складається з шести етапів:

- 1) розуміння бізнесу;
- 2) розуміння даних;
- 3) підготовка даних;

- 4) моделювання;
- 5) оцінка;
- 6) розгортання.

Система [7] використовує вхідні дані з потоків подій бінарних сенсорів, дані про близькість, прискорення зі смарт-годинника та інформацію про місцезнаходження зі смарт-підлоги. Для моделювання застосовувалися J48, I_b1, SVM, Random Forests, AdaBoostM1 та бутстреп-агрегація. Найкращі результати показав алгоритм AdaBoostM1.

Однак проблема підходу полягає у класовому дисбалансі набору даних: бракує достатньої кількості прикладів для діяльностей «мити посуд» та «грати у відеоігри», що ускладнює їх точне прогнозування. Наприклад, активності «сніданок», «обід» та «вечера» були об'єднані в єдину категорію «вживати їжу». Таке об'єднання також стосувалося підготовки страв для цих прийомів їжі. Фізичні рухи при цих діяльностях майже не відрізняються, різняться лише час доби, і об'єднання дозволило збільшити точність класифікації на 13%.

Незважаючи на те, що 10-кратна перехресна перевірка на навчальному етапі дала точність 92,1%, результати на тестовому наборі становили лише 60,1%, а без нульового класу – 62,77%. Результати [6] показали, що основною проблемою стандартних методів розпізнавання активності на основі сенсорів (SVM, приховані моделі Маркова, наївний баєсів класифікатор) є ручне добування ознак, що базується на досвіді та знаннях розробника. Такі ознаки виявляються поверхневими, наприклад, статистичні параметри – середнє або дисперсія, що недостатньо для складних видів діяльності [1].

Ще однією проблемою є невеликий маркований набір даних, адже для його створення потрібно вручну позначати активності. Для навчання моделей потрібно багато даних, а сенсорні вимірювання одного й того ж виду діяльності можуть змінюватися. Крім того, більшість стандартних алгоритмів розпізнавання образів працюють із статистичними даними, тоді як сенсорні дані надходять як послідовність подій, які мають сенс лише у певному порядку [1].

Враховуючи цю природу даних, моделі глибокого навчання дозволяють одночасно виконувати автоматичне вилучення абстрактних ознак та побудову моделі. Одним із варіантів є CNN, яка обробляє часові ряди та виявляє локальні шаблони [6]. Підхід [9] показав ефективність кількох алгоритмів глибокого навчання для розпізнавання людської активності за допомогою розумного одягу. Було порівняно нейронні мережі із прямим зв'язком, CNN та рекурентні мережі на основі LSTM, виявивши, що двонаправлені LSTM значно перевершують інші сучасні алгоритми класифікації.

Таким чином, сучасні методи розпізнавання людської активності об'єднують різні підходи: від експертних систем і скінченних автоматів до ймовірнісних та гібридних моделей. Це дозволяє підвищувати точність класифікації та адаптуватися до нових умов поведінки користувачів. Основні виклики включають непередбачувану поведінку людей, короткотривалі дії та обмеженість даних для навчання моделей.

1.4 Постановка завдання

Аналіз наявних підходів і технологій для розпізнавання людської поведінки в інтелектуальних домашніх системах продемонстрував, що сучасні рішення не забезпечують достатнього рівня автоматизації цього процесу та характеризуються невисокою точністю. У зв'язку з цим виникає потреба створення інтелектуальної системи та нового підходу до виявлення активностей користувача в «розумному» помешканні, які могли б суттєво підвищити автоматизацію роботи та якість реагування на потреби мешканців. Для досягнення поставленої мети слід виконати такі завдання:

- провести детальний огляд методів, моделей і технічних рішень, що використовуються для класифікації дій людини в розумних будинках, та визначити їх недоліки;

- створити модель, що описує процес класифікації різних видів активностей користувача в системі розумного дому;

- розробити інтелектуальну систему ухвалення рішень, яка регулюватиме роботу компонентів розумного будинку відповідно до класифікованих типів людської діяльності;

- на базі створеної системи здійснити оцінювання результативності класифікації дій та проаналізувати її ефективність.

2 КЛАСИФІКАЦІЯ ВИДІВ ДІЯЛЬНОСТІ ЛЮДИНИ У РОЗУМНОМУ БУДИНКУ

2.1 Модель розумного будинку для збору даних

Дослідження, представлене у цій роботі, базується на аналізі набору даних, зібраного у спеціалізованій розумній лабораторії під час Ubiquitous Computing and Ambient Intelligence Cup 2018 [10]. Ці дані використовуються для навчання моделей розпізнавання людської активності, а також для розробки комплексної системи, методів й інструментів, які забезпечують інтеграцію штучного інтелекту у сучасні розумні будинки, що дозволяє автоматизувати повсякденні процеси і покращити якість життя мешканців [1].

Предметом дослідження є розумна лабораторія, оснащена великою кількістю різних датчиків, розміщених таким чином, щоб можна було відслідковувати дії людей у всіх приміщеннях (рис. 2.1).

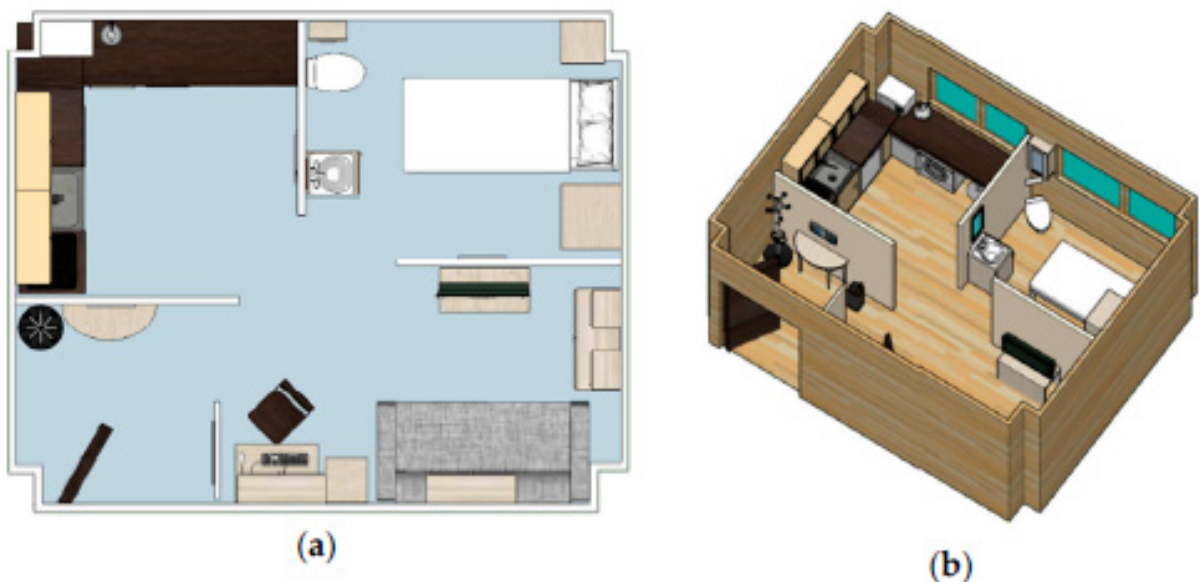


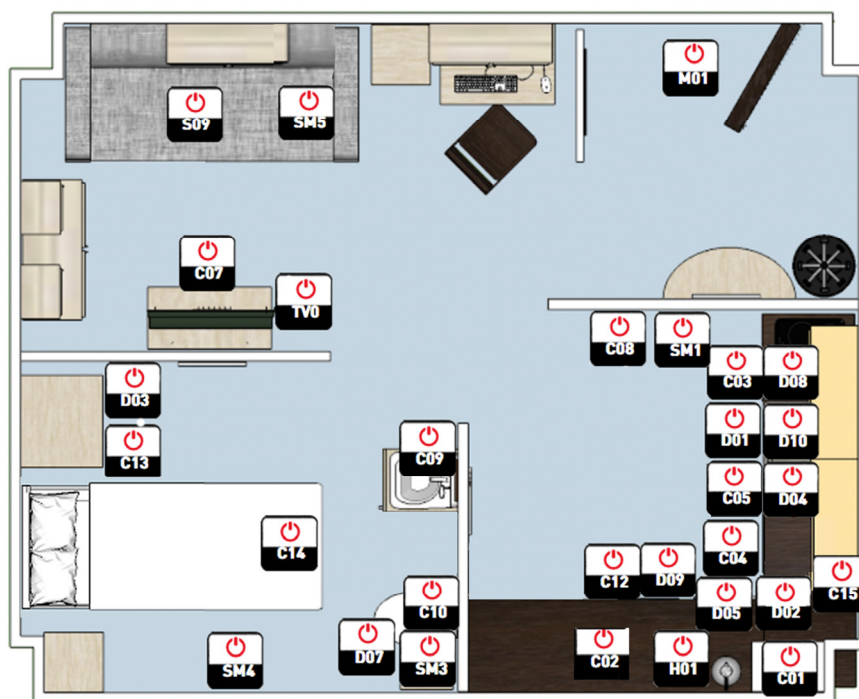
Рисунок 2.1 – Макет SmartLab: а) план; б) вигляд в 3-D

Лабораторія імітувала житлову квартиру та складалася з п'яти зон: передпокою, кухні, вітальні, спальні та ванної кімнати. У наборі даних представлені 24 різні види повсякденної діяльності, від приготування їжі та

прибирання до використання меблів і пересування по приміщеннях. Датчики були розташовані поблизу об'єктів, що контролюються, або прикріплені до них для точного збору інформації про дії мешканців.

Дані для аналізу надходять із чотирьох основних джерел:

- 30 бінарних датчиків, які передають двійкову інформацію про магнітний контакт, рух або тиск (рис. 2.2);
- інформація про наближення між розумним наручним годинником і 15 маяками Bluetooth Low Energy (BLE), встановленими в лабораторії (рис. 2.3);
- дані прискорення у трьох осях, які генерує смарт-годинник;
- інформація про місцезнаходження, що надходить із сенсорної підлоги, розбитої на модулі (рис. 2.4).



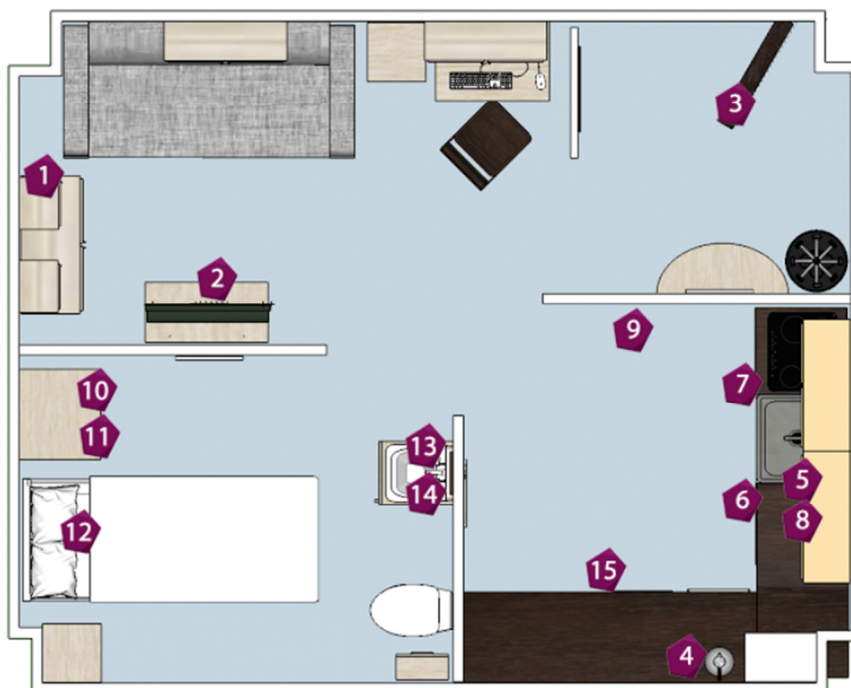
C01-Аптечка; C02-Блюдо з фруктами; C03-Столове приладдя; C04-Каструлі; C05-Пляшка з водою; C07-ХВОХ; C08-Відро для сміття; C09-Кран; C10-Резервуар; C12-Корзина для прання; C13-Ящик для піжами; C14-Ліжко; C15-Кухонний кран; D01-Холодильник; D02-Мікрохвильовка; D03-Шафа; D04-Сервіз; D05-Машина для миття посуду; D07-Туалет; D08-Кухонна шафа; D09-Пральна машина; D10-Буфетна; H01-Чайник; M01-Двері; S09-Диван; SM1-Датчик руху кухні; SM3-Датчик руху ванної кімнати; SM4-Датчик руху спальні; SM5-Датчик руху дивану; TV0-Телевізор

Рисунок 2.2 – Розташування бінарних датчиків

Бінарні датчики працюють бездротово за протоколом Z-Wave. Наприклад, якщо пляшка води використовується, її стан фіксується як «Відкрита», а в іншому випадку – «Закрита». Переміщення мешканців контролюють PIR-сенсори, які використовують протокол ZigBee і передають двійкові значення: «Рух» або «Без

руху». М'які меблі обладнані текстильними сенсорними шарами для фіксації тиску, що передають значення «Тиск присутній» або «Тиску немає» (рис. 2.2).

Дані про наближення надходять від 15 BLE-маяків, що передають сигнали на розумний годинник, обладнаний спеціальним програмним забезпеченням для обробки цих сигналів (рис. 2.3). Маяки також дозволяють запускати дії на основі місцезнаходження, наприклад автоматично включати певний пристрій або змінювати стан системи, коли людина входить у визначену зону [1].



1-Книга; 2-Контролер телевізора; 3-Вхідні двері; 4-Коробка з ліками; 5-Кухонна шафа; 6-Холодильник; 7-Ящик для каструль; 8-Пляшка з водою; 9-Відро для сміття; 10-Двері шафи; 11-Шухляда для піжами; 12-Ліжко; 13-Вмивальник; 14-Зубна щітка; 15-Корзина з пранням

Рисунок 2.3 – Розташування BLE маяків

Підлога обладнана технологією SensFloor, яка дозволяє відслідковувати пересування людини під будь-яким типом покриття. Набір даних SensFloor включає інформацію про вісім сенсорних полів, кожне з яких має індивідуальний ідентифікатор (рис. 2.4). Система працює шляхом подачі невеликого електричного заряду через провідну тканину з вбудованими сенсорними пластинами (32 пластини на квадратний метр), а проходження людини змінює ємність заряду, що активує сигнал.

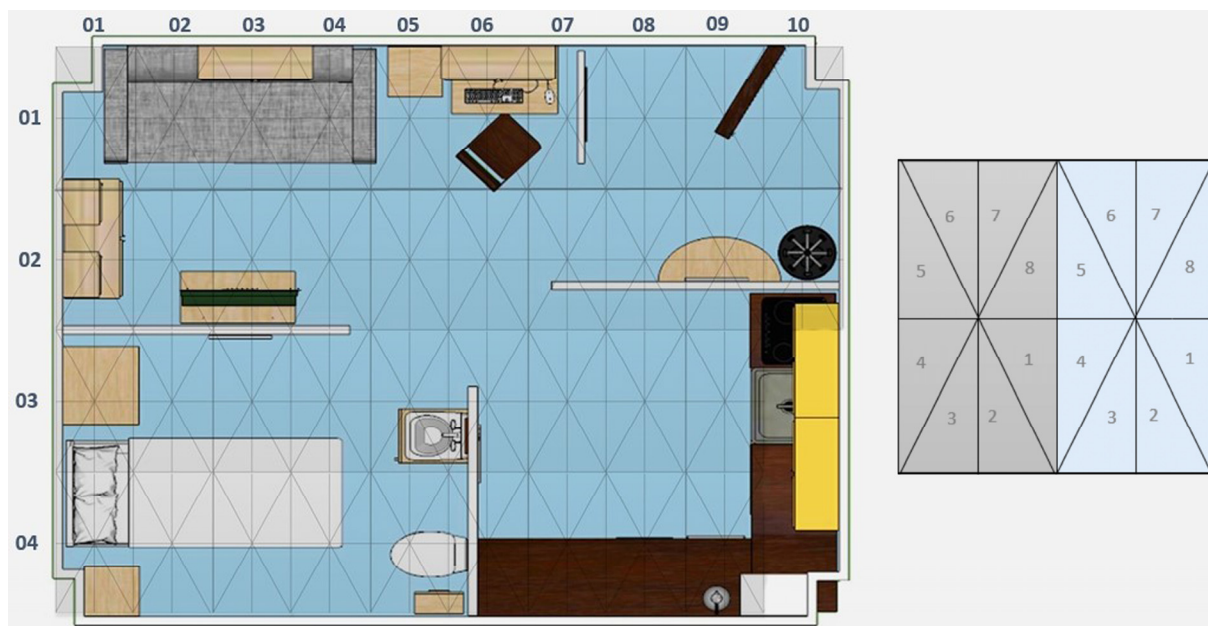


Рисунок 2.4 – Розташування модулів розумної підлоги

Хоча раніше проводилися дослідження з цим набором даних, більшість існуючих систем стикаються з проблемами при класифікації складних сценаріїв, особливо коли дії повторюються або накладаються одна на одну. У даній роботі пропонується алгоритм, що підвищує точність розпізнавання та оптимізує класифікацію людської активності в IoT-середовищах [1].

Система повинна підтримувати персоналізовані налаштування для кожного мешканця розумного будинку, що дозволяє підвищити якість життя за рахунок автоматизації. Один із практичних сценаріїв – нагадування про прийом ліків. Наприклад, мешканець може регулярно приймати гормональні препарати для щитовидної залози або засоби для контролю рівня цукру в крові. Система відстежує ці дії за допомогою сенсорної підлоги, магнітних контактів, датчиків наближення і даних прискорення годинника. Якщо прийом пропущено, надсилається автоматичне сповіщення на пристрій користувача.

IoT-платформа повинна виконувати низку нефункціональних вимог: надійність алгоритму класифікації, точність рішень, здатність обробляти дані в реальному часі та забезпечувати сумісність різних протоколів і стандартів. Додатки IoT формують єдину мережу, що дозволяє забезпечити взаємозв'язок між пристроями, сервісами зберігання даних і прикладними сервісами.

У розумній лабораторії датчики фіксують дії мешканців та надсилають інформацію до центральної системи. Система повинна ефективно керувати великою кількістю пристроїв, а у разі несправностей або низького заряду акумулятора – виявляти помилки і сигналізувати про них [1].

Основні вимоги безпеки для IoT-систем включають [11]:

- доступність сервісу та захист від DoS-атак;
- аутентифікація та авторизація для уповноважених користувачів;
- конфіденційність і цілісність даних завдяки шифруванню;
- енергоефективність пристроїв для тривалого використання.

Ще одним важливим аспектом є масштабованість, адже з ростом кількості пристроїв та обсягів даних збільшується складність їх обробки. Для прискорення навчання моделей глибокого навчання використовуються GPU, що забезпечує швидшу обробку великих обчислювальних задач.

Зручність користування панелями керування та іншими інтерфейсами тестується за допомогою експериментів із реальними користувачами, щоб забезпечити ефективну взаємодію мешканців із розумним будинком і підвищити комфорт користування.

2.2 Опрацювання набору даних

Підготуємо отримані дані для їх застосування в моделі машинного навчання. Для цього здійснимо форматування та впорядкування вхідної інформації, що дозволяє підвищити якість подальшого аналізу.

Процес підготовки даних можна подати як послідовність конкретних етапів.

- 1 Синхронізація сигналів датчиків у вибірки.
- 2 Верифікація даних за категорійними значеннями.
- 3 Масштабування ознак (нормалізація).
- 4 Очищення та фільтрація інформації.
- 5 Розподіл набору даних на підмножини.

Розглянемо кожен етап докладніше.

Крок 1. Синхронізація сигналів датчиків. Дані з різних сенсорів об'єднуються у вибірки за певний часовий проміжок. Це важливо, оскільки частоти збору інформації різняться, а бездротові мережі іноді створюють затримки. Для уникнення проблем обираємо оптимальний інтервал часу, протягом якого агрегуємо кілька вимірювань у одне значення для кожного джерела, забезпечуючи узгодженість даних.

Крок 2. Перевірка відповідності категорійним ознакам. Бінарні сенсорні дані вже представлені у двійковому форматі й можуть бути частково використані без змін. Інші дані потребують перетворення. Наприклад, інформацію про підлогу структуруємо за модулями плитки: значення встановлюємо в 1, якщо датчик зафіксував активність у проміжку часу, і 0 – якщо активності не було.

Додатково, для подій, пов'язаних із прийомами їжі, визначаємо сегмент дня: ранок, день і вечір, який кодуємо числами 0, 1 або 2 відповідно. Це дозволяє моделі розрізнити сніданок, обід та вечерю.

Ще один приклад – дані про наближення, отримані за допомогою RSSI (Received Signal Strength Indicator). Значення вище -97 вказують на близькість до об'єкта всередині квартири. Для уніфікації показник масштабуємо від 0 до 1, де 1 відповідає максимальному наближенню.

Крок 3. Масштабування (нормалізація) ознак. Небінарні дані, наприклад прискорення, мають різну шкалу, що може знизити ефективність моделі. Застосовуємо MinMaxScaler, який переводить значення у діапазон [0,1]:

$$x_{\text{min}} = \frac{x - \min(x)}{\max(x) - \min(x)}, \quad (2.1)$$

де x - значення даних.

Формула нормалізації (2.1) дозволяє забезпечити, що мінімум стане 0, а максимум – 1. Це прискорює процес навчання і робить модель більш чутливою до всіх ознак.

Крок 4. Очищення інформації. Деякі вибірки позначені як «без активності», але все ще містять дані. Такі записи видаляються, оскільки вони можуть

спотворити процес навчання і знизити точність прогнозів. Очищення підвищує якість набору даних і стабільність моделі.

Крок 5. Розподіл набору даних. Для підготовки до навчання здійснюємо поділ на навчальну і тестову підмножини методом k -кратного перехресного підтвердження. Дані поділяємо на k приблизно рівних частин. $k-1$ частину використовуємо для навчання, а одну – для тестування моделі. Процедуру повторюємо k разів, отримуючи k оцінок продуктивності. Остаточну ефективність обчислюємо як середнє цих оцінок.

Після виконання цих кроків отримуємо підготовлений набір даних без суперечливих або некоректних записів. Така обробка дозволяє підвищити точність прогнозів і забезпечує надійну основу для побудови моделей машинного навчання. Правильно підготовлені дані зменшують ризик перенавчання та підвищують стабільність моделі при роботі з новими даними.

2.3 Класифікація видів діяльності людини

Оскільки досліджується процес класифікації видів діяльності, які фіксуються в розумних будинках, важливою задачею стає створення відповідної моделі. Така модель дозволяє формалізувати ключові елементи та встановити взаємозв'язки між ними [1]. Для побудови структури класифікації людської діяльності в розумних будинках застосуємо підхід теорії множин, який забезпечує математичну точність опису процесу.

Модель розпізнавання активності людини на базі Naive Bayes Classifier можна представити як кортеж, що включає наступні компоненти:

$$M = \langle K, A, T, S, \gamma \rangle, \quad (2.2)$$

де K – класифікатор;

A – множина всіх видів діяльності;

T – багатовимірний часовий ряд, що фіксує події;

S – набір сенсорів, розміщених у розумному будинку;

γ – функція, що визначає відповідність конкретній діяльності.

Детально розглянемо складові цієї моделі. Визначення виду діяльності відбувається відповідно до одного з 24 типів щоденної активності людини, отриманої з вхідних даних [1]. Множину таких видів можна представити як

$$A = \{A_i\}_{i=1}^{24}, \quad (2.3)$$

Сенсори розумного будинку для збору даних представлені наступним чином:

$$S = \{S_i\}_{i=1}^4 = \langle s_f, s_o, s_a, s_m \rangle, \quad (2.4)$$

де s_f – датчики на «розумній» підлозі;

s_o – датчики, інтегровані в об'єкти всередині приміщень;

s_a – акселерометри з розумного годинника на руці користувача;

s_m – бінарні магнітні сенсори для контролю відкривання дверей або вікон.

Модель класифікатора у вигляді кортежу включає:

$$K = \langle \gamma, X, v \rangle, \quad (2.5)$$

де γ – функція визначення діяльності за ознаками;

X – вхідний вектор ознак;

v – індикатор присутності людини (0 або 1), що обов'язково для класифікації.

Функція активності базується на теоремі Баєса:

$$P(A|X) = \frac{P(X|A)P(A)}{P(X)}, \quad (2.6)$$

де $P(A|X)$ – апостеріорна ймовірність виду діяльності за даним вектором ознак;

$P(A)$ – апіорна ймовірність;

$P(X|A)$ – ймовірність ознак для виду;

$P(X)$ – апіорна ймовірність ознак.

Вхідний вектор формується на основі даних сенсорів:

$$X = \{x_i\}_{i=1}^5 = \langle x_z, x_o, x_D, x_{mv}, x_{mg} \rangle, \quad (2.7)$$

де x_z – зона розташування людини;

x_O – об’єкт взаємодії (чайник, телевізор, ліжка тощо);

x_D – період доби (ранок, день, вечір);

x_{mv} – наявність або відсутність руху;

x_{mg} – стан магнітного контакту (бінарний).

Дані про місцезнаходження людини представлені як кортеж:

$$x_Z = \langle F, \delta \rangle, \quad (2.8)$$

F – інформація про модулі «розумної підлоги»;

δ – функція, що визначає приналежність до певного приміщення (кухня, спальня, ванна тощо).

Сигнали з підлогових сенсорів надходять із 38 модулів, кожен з яких фіксує пересування людини, представлене як:

$$F = \{f_i\}_{i=1}^{38}, \quad (2.9)$$

де f_i – модуль розумної підлоги.

Параметри часу доби (ранок - 0, обід - 1, вечір - 2) визначаються функцією:

$$x_D = v(t), \quad (2.10)$$

де t – час діяльності.

Розширена теорема Баєса дозволяє враховувати всі ознаки одночасно, при цьому знаменник залишається сталим:

$$P(A | x_1, \dots, x_n) = \frac{P(x_1 | A)P(x_2 | A) \dots P(x_n | A)P(A)}{P(x_1)P(x_2) \dots P(x_n)}, \quad (2.11)$$

і вводимо пропорційність:

$$P(A | x_1, \dots, x_n) \propto P(A) \prod_{i=1}^n P(x_i | A). \quad (2.12)$$

Результатом класифікатора стає вид діяльності з максимальною ймовірністю:

$$\gamma = \arg \max_A P(A) \prod_{i=1}^n P(x_i | A). \quad (2.13)$$

Ймовірність появи ознаки в певному класі визначається як:

$$P(x_i | A) = \frac{N_i}{N}, \quad (2.14)$$

де N_i – кількість входжень ознаки x_i у класі A ;

N – загальна кількість входжень класу A .

Для уникнення проблеми нульової частоти застосовують згладжування Лапласа, додаючи одиницю до кожної частоти:

$$P(x_i | A) = \frac{N_i + 1}{N + k}, \quad (2.15)$$

де k – кількість можливих значень ознаки.

Алгоритм навчання класифікатора формалізовано як кортеж:

$$L = \langle D, \tau, \vartheta \rangle, \quad (2.16)$$

де D – початковий набір даних із відомими парами (X, A) ;

τ – функція обробки даних,

ϑ – функція побудови моделі та навчання.

Для перетворення даних застосовується k -кратне перехресне затвердження:

$$\tau = \frac{D}{k} = \langle B, C \rangle, \quad (2.17)$$

де k – кількість частин даних;

B – тренувальна множина, $k-1$ частина;

C – тестова множина.

Функція навчання виконує k ітерацій, використовуючи кожну підвибірку як перевірочну:

$$\vartheta = \langle \phi, \mu \rangle, \quad (2.18)$$

де ϕ – перехресне затвердження (повторюється k разів);

μ – функція ефективності моделі.

Ефективність моделі обчислюється середньоквадратичною похибкою:

$$\mu = \frac{1}{k} \sum_i MSE_i, \quad (2.19)$$

де MSE_i – похибка i -тої ітерації:

$$MSE_i = \frac{1}{n} \sum_i (y_i - f(x_i))^2, \quad (2.20)$$

де n – кількість спостережень;

y_i – отримані значення;

$f(x_i)$ – прогнозовані.

Ця модель дозволяє систематично оцінювати точність класифікації, інтегруючи дані з різних сенсорів і зберігаючи гнучкість для майбутніх розширень, наприклад, додавання нових типів діяльності або сенсорів.

Для ідентифікації людських дій у системі «розумний будинок» запропоновано підхід, результатом якого є визначення типу активності на основі набору вхідних параметрів. Узагальнений вигляд цього методу представлено на рисунку 2.5.

Датасет містить кілька часових потоків, які об'єднано у єдиний багатовимірний часовий масив, що охоплює різні змінні. Часовий ряд описує зміну однієї чи кількох випадкових величин протягом вибраного проміжку часу. Дані фіксуються з однаковим кроком, який становить 5 секунд. На кожному кроці зберігається значення $x(t)$, що відображає показник певного сенсора в момент t .

Багатовимірний часовий ряд T є впорядкованою послідовністю спостережень, зібраних у часі, і формально задається так:

$$T = X_i(t); [i = 1, \dots, n; t = 1, \dots, m], \quad (2.21)$$

де i – індекс окремих вимірів, що фіксуються в час t ;

n – кількість параметрів, що безперервно відстежуються;

m – обсяг вибірки.

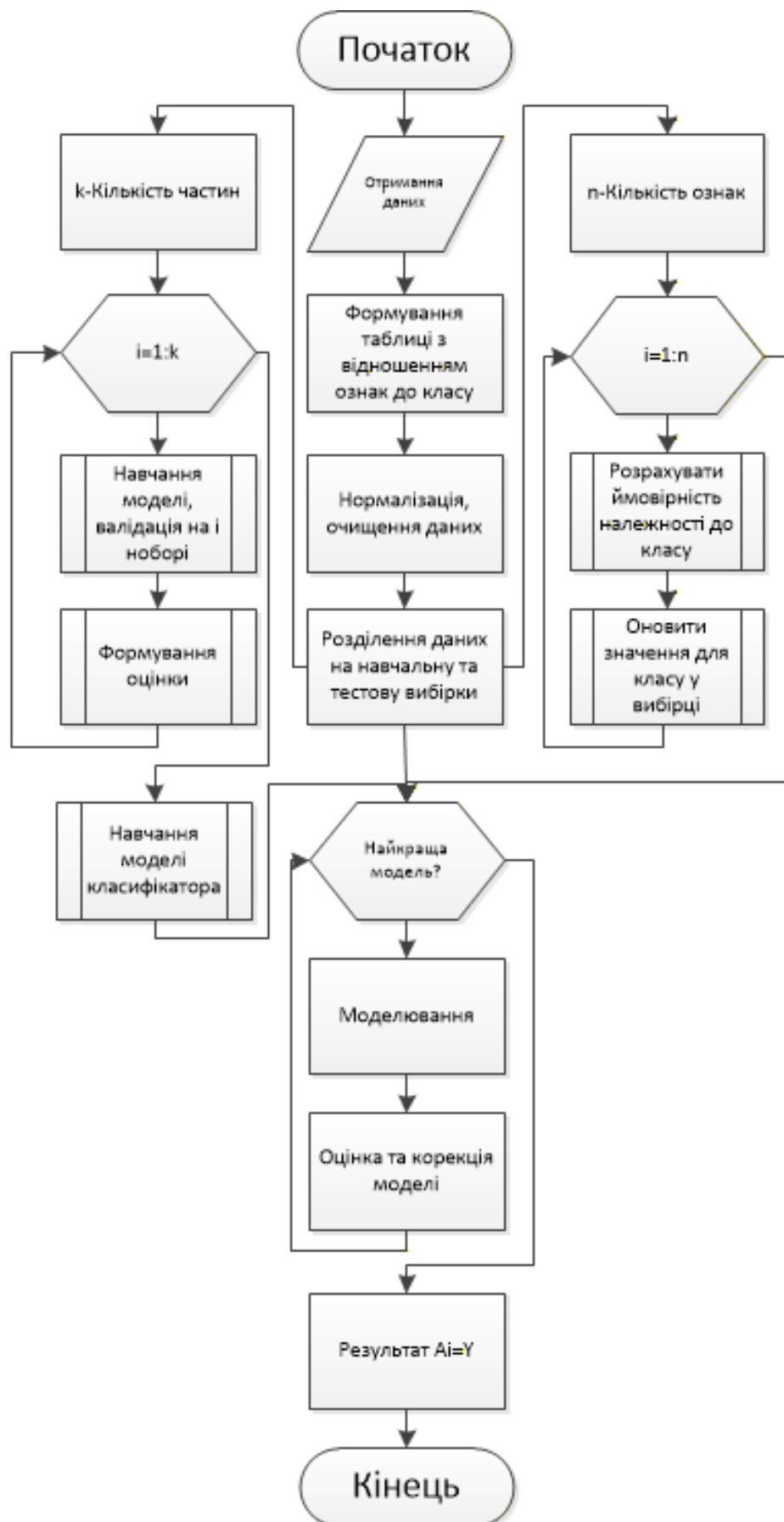


Рисунок 2.5 – Алгоритм ідентифікації активності людини в розумному будинку

Ряди, отримані від бінарних сенсорів, містять лише два можливі значення: 1 повідомляє про наявність руху, тиску або магнітного контакту, а 0 означає їхню відсутність. Якщо протягом п'яти секунд датчик видає кілька значень, підсумковим записом буде 1, якщо принаймні один показник дорівнює 1, інакше фіксується 0. Бінарний часовий ряд $S(t)$ включає послідовність змін S_1, S_2, \dots , що вибираються з однаковим часовим інтервалом.

Для небінарних датчиків прикладом є сенсор наближення, де вимірювання базуються на рівні RSSI. Такі спостереження представляють собою послідовність негативних числових значень $S(t)$, що змінюються відповідно до переміщення людини в приміщенні (рис. 2.6).

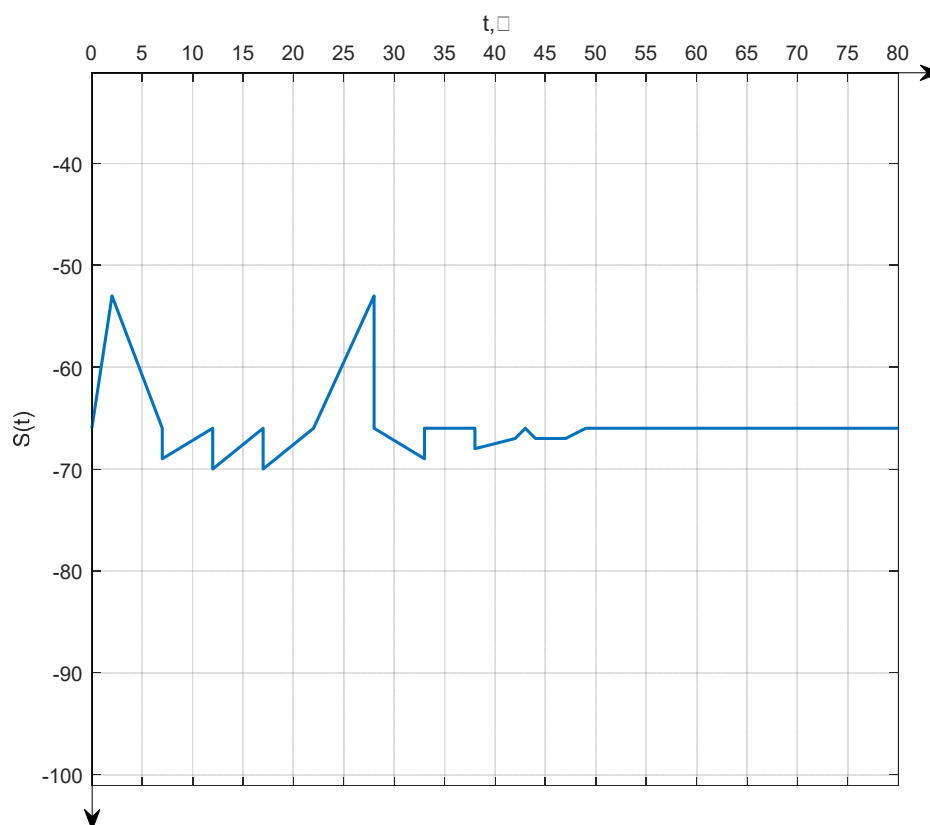


Рисунок 2.6 – Часовий ряд небінарного датчика $S(t)$

Щоб підвищити якість роботи моделі, виконується нормалізація – перенесення всіх числових значень у діапазон від 0 до 1 (рис. 2.7). Маючи однакову шкалу для всіх джерел даних, багатовимірні ряди об'єднують в єдину структуру. Перед процедурою класифікації виконується поділ на навчальну та тестову частини.

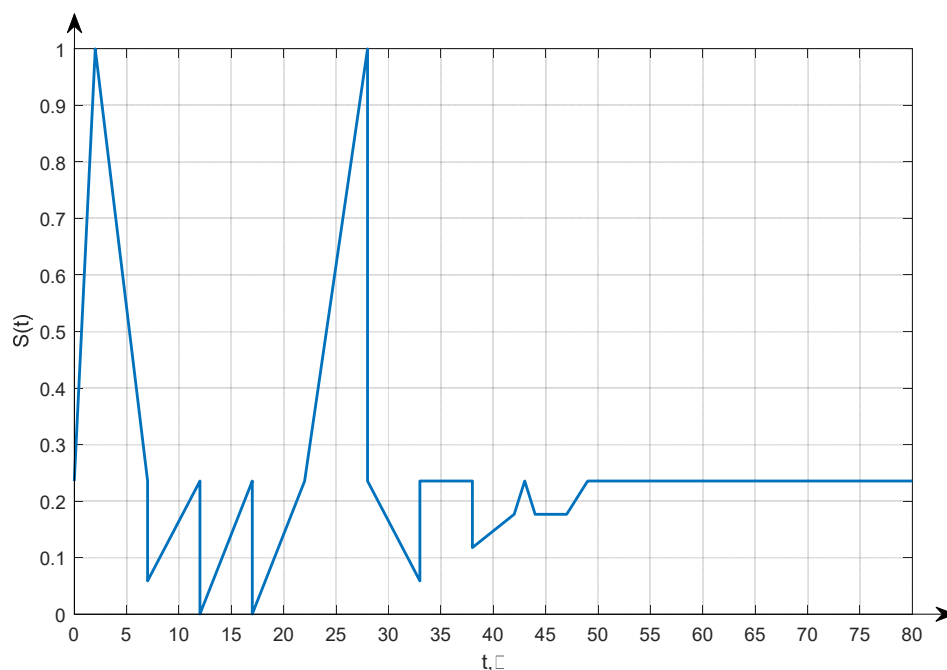


Рисунок 2.7 – Нормалізований часовий ряд небінарного датчика $S(t)$

Основне завдання машинного навчання полягає у створенні моделі, здатної узагальнювати інформацію та працювати з прикладами, яких вона не бачила під час тренування. Для раціонального використання вибірки застосовують k -кратне перехресне затвердження як метод розподілу даних.

Перед тренуванням моделі задаються параметри, формуючи простір гіперпараметрів. Під час виконання алгоритму послідовно обирається одна з можливих комбінацій цих значень. Для кожного набору параметрів модель навчається та оцінюється за допомогою k -кратної валідації, а отриманий показник (наприклад, MSE) зберігається.

Після перебору всіх комбінацій алгоритм накопичує отримані оцінки та визначає найбільш ефективну. На основі найкращого результату формується оптимізована модель, яка містить гіперпараметри, що забезпечили максимальну продуктивність (рис. 2.8).

Використання k -кратного перехресного затвердження дозволяє підвищити точність, стабільність і надійність моделі. Додатково цей метод допомагає уникати перенавчання, забезпечуючи краще узагальнення у задачах розпізнавання активностей у розумних середовищах.

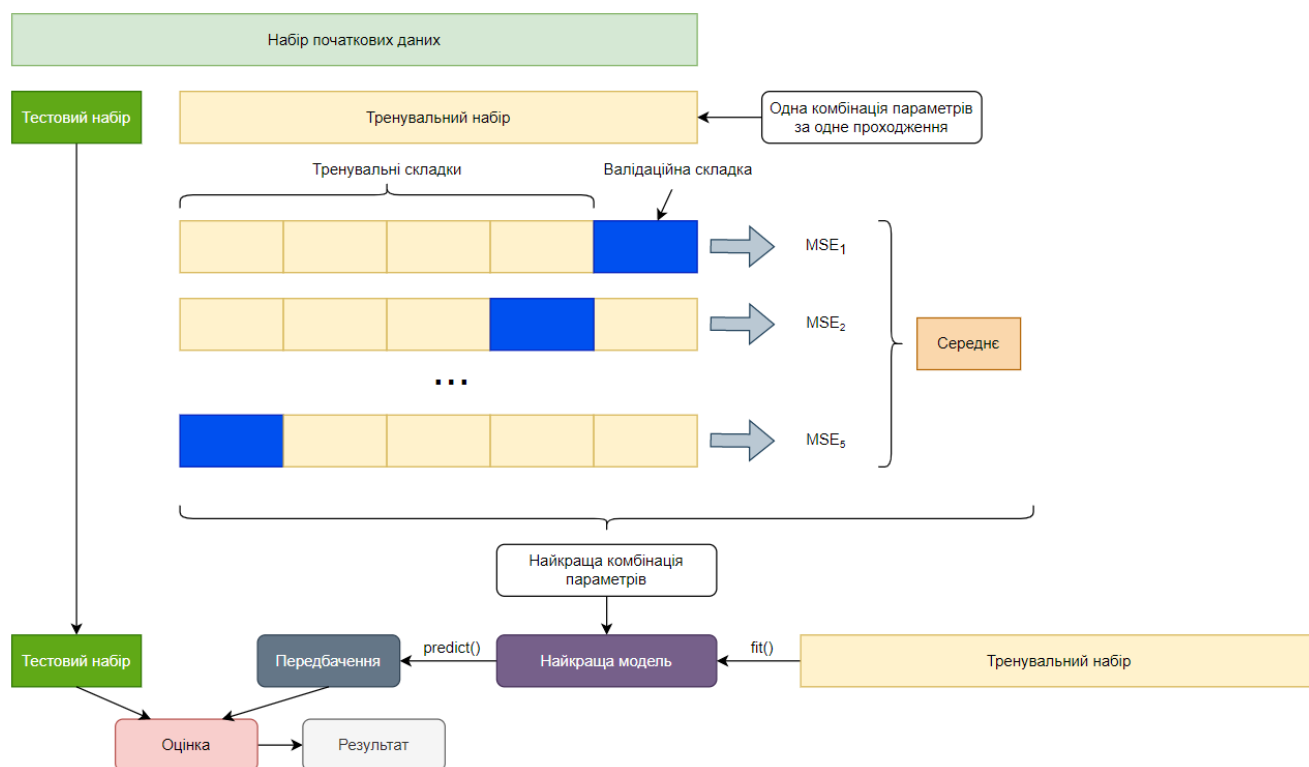


Рисунок 2.8 – k-кратне перехресне затвердження при $k = 5$

Аналіз часових рядів буде виконувати Байєсівська мережа безперервного часу. Ця модель є графічною структурою, вузли якої відповідають випадковим величинам, а стан системи змінюється безперервно у часі. Відповідно, еволюція кожної змінної визначається станом її батьківських вузлів на графі. Мережа включає дві основні складові:

1 початковий розподіл ймовірностей, який визначає базові значення для всіх змінних;

2 динамічний компонент, що керує розвитком ймовірностей у часі та описує, як стан вузлів змінюється в залежності від інших елементів мережі.

Розглянемо Байєсівську мережу безперервного часу більш детально. Нехай $X = \{X_1, X_2, \dots, X_N\}$ – набір випадкових величин. Кожна змінна X_n має скінченний простір значень $Val(X_n) = \{x_1, x_2, \dots, x_n\}$. Мережа η включає дві ключові компоненти: початковий розподіл ймовірностей P_x^0 , який задається традиційною Байєсівською мережею β для X , та модель безперервного переходу, що визначається через:

- орієнтований (можливо циклічний) граф δ із вузлами X_1, X_2, \dots, X_N ;

- матрицю умовної інтенсивності $Q_{X_n}^{pa(X_n)}$ для кожної змінної $X_n \in X$, яка складається з набору матриць для всіх можливих комбінацій станів $pa(x_n)$ батьків $pa(X_n)$ вузла X_n .

Кожна матриця відображає миттєву ймовірність переходу між станами змінної X_n для конкретного набору батьківських станів:

$$Q_{X_n}^{pa(x_n)} = \begin{bmatrix} -q_{x_1}^{pa(x_n)} & q_{x_1x_2}^{pa(x_n)} & \dots & q_{x_1x_1}^{pa(x_n)} \\ q_{x_2x_1}^{pa(x_n)} & -q_{x_2}^{pa(x_n)} & \dots & q_{x_2x_1}^{pa(x_n)} \\ \dots & \dots & \dots & \dots \\ q_{x_1x_1}^{pa(x_n)} & q_{x_1x_2}^{pa(x_n)} & \dots & -q_{x_1}^{pa(x_n)} \end{bmatrix}. \quad (2.22)$$

Байєсівська мережа дозволяє використовувати два типи доказів: точкові та безперервні. Точковий доказ означає спостереження конкретного значення x змінної X_n у визначений момент часу, тобто $X_n(t) = x$. Безперервний доказ надає інформацію про значення змінної протягом інтервалу $[t_1, t_2)$, де крайні точки можуть бути включені або виключені. Мережа створюється на основі послідовності доказів, яка називається потоком доказів.

Для опису потоку доказів потрібен часовий розподіл. J -часовий потік визначається як розбиття інтервалу $[0, T)$ на J підінтервалів: $[0, t_1)$, $[t_1, t_2)$, ..., $[t_{j-1}, T)$. J -потік доказів включає спільні екземпляри $X = x$ для будь-якої підмножини змінних X_n , пов'язаних із кожним із J відрізків часу. Його позначають як $(X^1 = x^1, X^2 = x^2, \dots, X^J = x^J)$ або скорочено (x^1, x^2, \dots, x^J) . Потік вважається повністю спостережуваним, якщо стан усіх змінних X_n відомий на всьому інтервалі $[0, T)$, і частково спостережуваним у протилежному випадку.

Для класифікації людської активності за багатовимірними часовими рядами X_n , $n=1,2,\dots,N$, застосовують Bayesian Network Classifier. Для визначення нового класу додається вузол Y , який представляє клас об'єкта. Модель Bayesian Network Classifier безперервного часу описується як графічна структура:

$$C = \{\eta, P(Y)\}, \quad (2.23)$$

де η – модель мережі з вузлами атрибутів X_1, X_2, \dots, X_N ;

$P(Y)$ – гранична ймовірність для станів $ValY = \{y_1, y_2, \dots, y_k\}$.

Граф G , що представляє мережу, повинен бути зв'язним і відповідати умовам:

- $pa(Y) = \emptyset$, тобто вузол класу Y є кореневим;
- Y повністю визначає $P(Y)$ та не залежить від часу.

Модель може містити цикли (рис. 2.9). Наприклад, один цикл охоплює вузли X_3, X_2, X_4 та X_5 , а інший вузли X_1 і X_3 , де X_1 впливає на X_3 і потім повертається до X_3 .

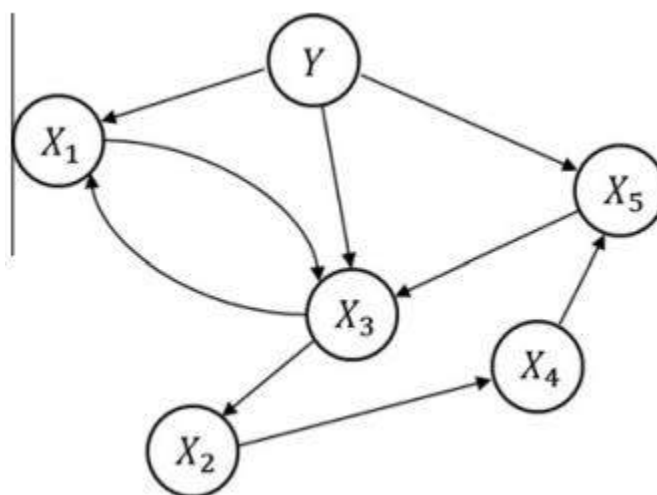


Рисунок 2.9 – Граф Bayes Network Classifier

Маючи навчальний набір D з повністю спостережуваними потоками доказів, класифікатор тренується шляхом максимізації функції оцінки,

$$S_{\eta}(G : D) = \ln P(D | G) + \ln P(G), \quad (2.24)$$

де D – дані навчання;

G – граф класифікатора.

Для ефективного навчання потрібно обмежити максимальну кількість батьків L для вузлів X_1, X_2, \dots, X_N . Високе значення L збільшує обчислювальні ресурси, необхідні для пошуку оптимальної структури G , що максимізує функцію оцінки (2.24).

До класу Bayesian Network Classifier безперервного часу належать такі варіанти: Naive Bayes Classifier безперервного часу та дерево-класифікатор з доповненням Naive Bayes.

Naive Bayes Classifier безперервного часу – такий класифікатор (2.23), коли $pa(X_n) = Y$, $n = 1, 2, \dots, N$.

У дерево-класифікаторі, доповненому Naive Bayes, вузли формують дерево з такими умовами:

- 1 $Y \in pa(X_n)$, $n = 1, 2, \dots, N$;
- 2 $\exists j: |pa(X_j)| = 1$, $i \neq j$, $|pa(X_i)| = 2$.

Вузли X_n формують дерево, де кожна змінна має батьківський вузол Y та максимум одного іншого атрибута (рис. 2.10).

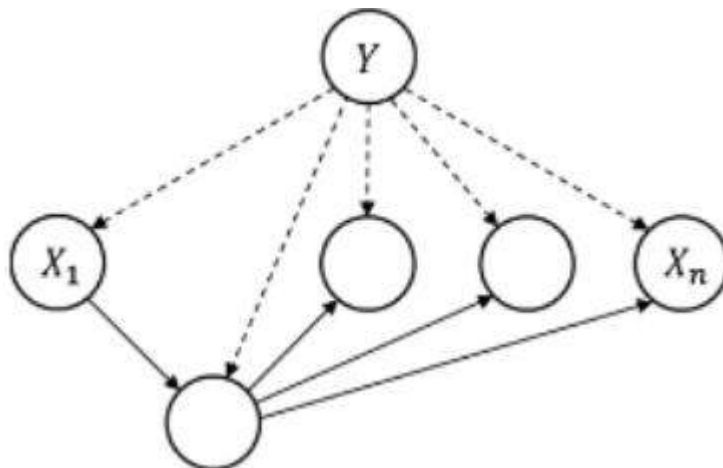


Рисунок 2.10 – Класифікатор безперервного часу з Naive Bayes Classifier

Коли класифікатор працює з повністю спостережуваним J -потокком доказів (x^1, x^2, \dots, x^J) , прогнозування класу здійснюється на основі стандартного Naive Bayes Classifier, забезпечуючи точну оцінку ймовірності належності до класу у динамічному часовому інтервалі.

Для виконання класифікації обрано Naive Bayes Classifier. Він вирізняється високою швидкістю роботи, здатністю точно передбачати класи для тестових наборів даних та ефективністю при малих обсягах даних з великою кількістю ознак. У порівнянні з іншими алгоритмами, Naive Bayes часто показує кращі результати саме в умовах обмеженого набору даних.

Додатково, цей класифікатор добре справляється з випадками, коли навчальні дані не містять певних сценаріїв, легко оновлюється при появі нових даних, має високу масштабованість та дозволяє робити ймовірнісні прогнози. Він не потребує великих обсягів навчальних прикладів, нечутливий до шумових або малозначущих ознак та здатний ефективно працювати у реальному часі. На великих масивах даних Naive Bayes демонструє високу точність і швидкість обчислень. Саме ці характеристики зумовили його вибір для класифікації.

Naive Bayes є ймовірнісним алгоритмом, що базується на теоремі Байєса і передбачає, що всі предиктори незалежні між собою. Це означає, що конкретна ознака не залежить від інших у межах класу. Основне завдання класифікатора полягає у визначенні ймовірності класу за значеннями ознак, тобто обчисленні $P(Y|X)$:

$$f: X \rightarrow Y, \quad (2.25)$$

де Y – змінна, що позначає клас;

X – вектор з n ознак, отриманих від сенсорів або інших джерел.

Класами виступають двадцять чотири попередньо визначені активності людини, а вектор X формується з даних різних датчиків розумного будинку

$$X = \langle X_1, X_2, \dots, X_n \rangle, \quad (2.26)$$

де X_i – конкретна ознака вектора X .

Використовуючи теорему Байєса, ймовірність $P(Y = y_i | X)$ можна виразити через розподіл $P(X | Y)$ та апіорну ймовірність $P(Y)$:

$$P(Y = y_i | X = x_k) = \frac{P(X = x_k | Y = y_i)P(Y = y_i)}{P(X = x_k | Y = y_j)P(Y = y_j)}, \quad (2.27)$$

де y_i – i -те можливе значення Y ;

y_j – j -те можливе значення Y ;

x_k – k -те можливе значення вектора X .

Для цього алгоритм зберігає розподіли ймовірностей для кожного класу, що дозволяє робити точні прогнози $P(Y|X = x_k)$ для нових спостережень x_k .

Для повного обчислення $P(Y|X)$ зазвичай потрібні великі набори даних, що дозволяють оцінити ймовірності для всіх комбінацій ознак. Naive Bayes спрощує цю задачу завдяки припущенню незалежності ознак: кожна X_i умовно незалежна від будь-якої іншої X_k при відомому Y . Це значно спрощує оцінку $P(Y|X)$ з навчальних даних. Таким чином, функція ймовірності з n ознак приймає простішу форму, що робить алгоритм обчислювально ефективним:

$$P(X_1 \dots X_n | Y) = \prod_{i=1}^n P(X_i | Y). \quad (2.28)$$

Алгоритм класифікації будується так: на основі дискретної змінної Y та атрибутів $X_1 \dots X_n$ навчальна модель обчислює розподіл ймовірності для кожного можливого значення Y згідно з теоремою Байеса:

$$P(Y = y_k | X_1 \dots X_n) = \frac{P(Y = y_k) \prod_{i=1}^n P(X_i | Y = y_k)}{\sum_j P(Y = y_j) \prod_{i=1}^n P(X_i | Y = y_j)}. \quad (2.29)$$

Найімовірніше значення Y обирається як результат класифікації:

$$Y \leftarrow \arg \max_{y_k} \frac{P(Y = y_k) \prod_{i=1}^n P(X_i | Y = y_k)}{\sum_j P(Y = y_j) \prod_{i=1}^n P(X_i | Y = y_j)}. \quad (2.30)$$

Для спрощення обчислень знаменник виразу можна опустити, оскільки він не залежить від y_k :

$$Y \leftarrow \arg \max_{y_k} P(Y = y_k) \prod_{i=1}^n P(X_i | Y = y_k). \quad (2.31)$$

Вихід нового прикладу визначається максимальною ймовірністю $C = Y$ на основі заданих ознак (рис. 2.11).

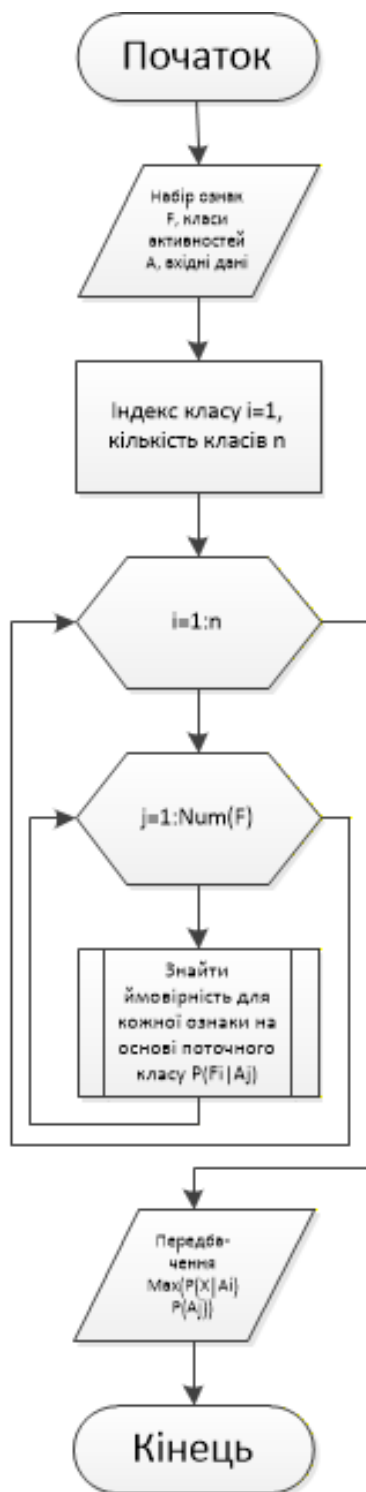


Рисунок 2.11 – Блок-схема роботи алгоритму ідентифікації класу діяльності

Якщо ознаки неперервні, варто обрати оптимальний тип класифікатора. Для таких випадків найбільш ефективним є Gaussian Naive Bayes Classifier. Він припускає, що для кожного значення y_k ймовірнісний розподіл X_i відповідає нормальному (гаусовому) розподілу з певним математичним сподіванням і середнім квадратичним відхиленням характерними для X_i і y_k .

Для навчання цього типу класифікатора оцінюються параметри μ та σ для кожної ознаки X_i та класу y_k . Математичне сподівання визначається як середнє значення ознак:

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i, \quad (2.32)$$

де n – кількість ознак;

x_i а – i -те значення ознаки x ;

а середнє квадратичне відхилення – як стандартне відхилення від середнього:

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)^2}. \quad (2.33)$$

Функція щільності ймовірності гаусівського розподілу використовується для оцінки ймовірності $P(X_i | y_k)$, що стає основою прогнозування:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (2.34)$$

Таким чином, Gaussian Naive Bayes дозволяє ефективно працювати з неперервними ознаками, обчислювати ймовірності для нових спостережень та забезпечує швидке та точне передбачення класів у системах з багатьма датчиками та складними наборами даних.

2.4 Висновок до розділу

Було здійснено детальний аналіз початкових даних, отриманих у спеціалізованій розумній лабораторії, обладнаній різноманітними датчиками та маяками для спостереження за діями людини. Проведено первинну обробку отриманої інформації, сформовано навчальну вибірку та підготовлено окремі тестові набори даних.

На основі опрацьованих значень сенсорів побудовано модель розпізнавання активності користувача за допомогою Naive Bayes Classifier. Такий підхід забезпечує визначення найбільш імовірної дії мешканця розумного середовища,

беручи до уваги наявний вектор характеристик. Класифікатор дозволяє здійснювати прогнозування навіть за умов багатовимірності та неповноти деяких спостережень.

Також розроблено метод розпізнавання людської активності в умовах інтелектуального житла, який ґрунтується на аналізі багатовимірних часових рядів та застосуванні Bayesian Network Classifier. Запропонована модель відноситься до класу ймовірнісних графічних структур, здатних відображати часову динаміку станів. Використання такої мережі робить можливим відтворення ймовірнісних залежностей у моменти появи певних дій або подій.

Процедура отримання класифікаційного висновку базується на застосуванні Naive Bayes Classifier, оскільки його стабільність, точність та невибагливість до обсягів даних забезпечують надійну роботу у широкому спектрі сценаріїв. Завдяки цьому класифікатор можна ефективно використовувати як у моделях реального часу, так і при значних масивах історичних даних.

Запропонований метод успішно долає складність задачі розпізнавання великої кількості можливих класів людської діяльності. Він дозволяє ідентифікувати одну з численних потенційних активностей, досягаючи високого рівня точності навіть за умов значної варіативності поведінки користувача та неоднорідності сенсорних даних.

3 ІНТЕЛЕКТУАЛЬНИЙ МОДУЛЬ СИСТЕМИ РОЗУМНОГО БУДИНКУ НА ОСНОВІ КЛАСИФІКАЦІЇ ВИДІВ ДІЯЛЬНОСТІ ЛЮДИНИ

3.1 Структура інтелектуальної системи

На основі створених механізмів та побудованої моделі ідентифікації людської поведінки у середовищі «розумного дому», що використовує класифікатор Naive Bayes, було сформовано модуль інтелектуальної системи підтримки прийняття рішень щодо керування компонентами домашньої автоматизації. Система працює, спираючись на результати класифікації типів активності користувача. Узагальнена структурна схема такої платформи схематично подана на рисунку 3.1.

Розроблювана система має забезпечувати сумісність із широким спектром сенсорів, що можуть походити від різних виробників та підтримувати різні протоколи. Передбачено гнучкий модульний веб-інтерфейс, який дозволяє відображати значення вимірювань, а також окремий застосунок для керування інформаційними потоками. Зібрані дані використовуються для аналітики та навчання моделі ML, що автоматично визначає характер поведінки мешканця [1].

Платформа створена таким чином, щоб забезпечити високу продуктивність і масштабованість, а також спирається на сучасні, надійні та перевірені інструменти з відкритим кодом.

Основні набори даних від сенсорів, а також метадані, пов'язані з роботою системи, зберігаються у високошвидкісному сховищі з малою затримкою. Воно побудоване на базі двох систем управління даними – MongoDB і InfluxDB, що разом формують гнучку та продуктивну інфраструктуру.

InfluxDB є спеціалізованою базою часових рядів, оптимізованою для роботи з великими потоками даних, у яких ключовим індексом виступає часова позначка. Вона має високу роздільну здатність індексації (наносекунди), і завдяки цьому ефективно обробляє масові послідовні операції читання та запису.

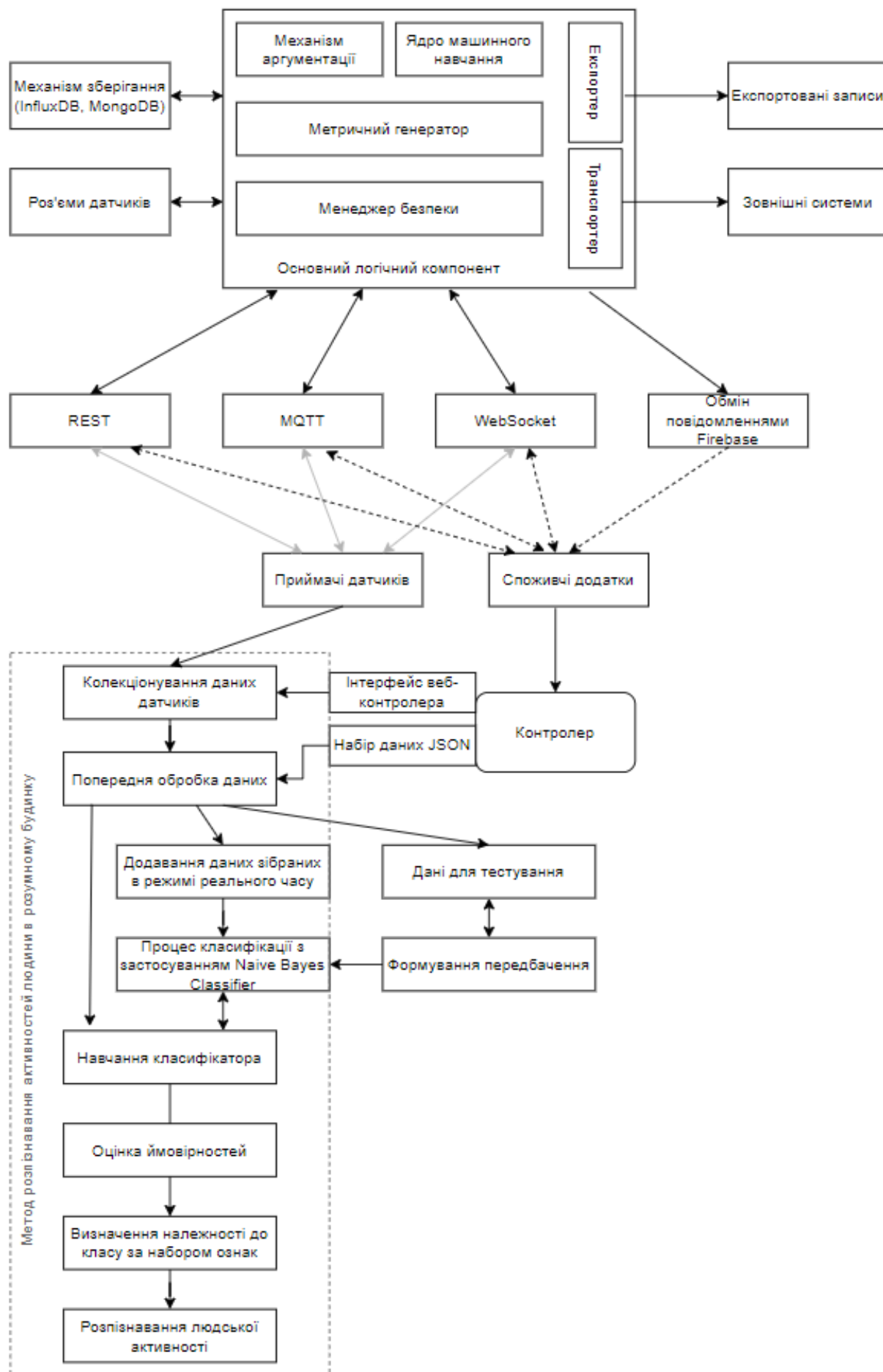


Рисунок 3.1 – Структура модуля інтелектуальної системи

База дозволяє зберігати довільні параметри в кожному записі, а кластеризація її інфраструктури забезпечує подальше горизонтальне масштабування. Сукупність цих характеристик робить InfluxDB придатною для сенсорних систем, що генерують великі обсяги даних у реальному часі.

Водночас ця СУБД не оптимізована для довільних змін інформації або випадкових вставок. Тому для роботи з даними іншого типу використовується окрема система – MongoDB. У ній записи зберігаються у документно-орієнтованій моделі, формованій у BSON-структурах, які можна зручно маніпулювати через JSON-подібний формат. MongoDB використовується для збереження метаданих, профілів користувачів, ключів API, параметрів експериментів та іншої інформації, що часто змінюється. Її масштабованість та гнучкість чудово підходять для потреб такої системи [1].

Обидві бази даних є повністю відкритими й не потребують додаткових витрат на ліцензії, що зменшує ризики залежності від окремого постачальника та підтримує стабільний розвиток системи без фінансових обмежень. Система зберігання інтегрована у загальну архітектуру та взаємодіє з головним логічним модулем платформи [1].

Центральний логічний компонент включає низку підсистем: модуль безпеки, засоби формування показників, механізм роботи з правилами, ядро машинного навчання, а також блок експорту та передавання даних.

Засіб безпеки забезпечує аутентифікаційні процедури, включно з SHAР-перевіркою, роботою ключів API (512-бітних), керуванням обліковими записами та валідацією токенів. Криптографічні функції реалізовані за допомогою бібліотеки Bouncy Castle, яка забезпечує надійний генератор випадкових чисел та широкий набір алгоритмів шифрування [1].

Обчислювальна логіка формування показників на стороні сервера мінімізує навантаження на пристрої користувачів і зменшує обсяг переданих даних, що позитивно впливає на швидкість обробки. Для класифікації вікон значень використовується rule-based engine, у якому застосовуються оператори логічного

порівняння (більше, менше, дорівнює) та їх заперечення. Правила можуть задаватися через веб-інтерфейс або спеціальний Java API.

Підсистема машинного навчання побудована на основі бібліотеки Scikit-learn, що містить потужні інструменти для прогнозування аналітики даних. Оскільки цей пакет базується на NumPy, SciPy та matplotlib, він забезпечує гнучке моделювання, швидке створення прототипів та високу сумісність із типовими ML-процесами.

Основний модуль опрацьовує передавання та експорт даних сенсорів. Передача даних у реальному часі здійснюється через REST-запити та WebSocket-з'єднання, а експортування дозволяє сформувати цілісний JSON-файл, який містить всі вимірювання та експериментальні дані, пов'язані з конкретною подією.

Головна логіка реалізована у вигляді Java-бібліотеки, яка дозволяє зовнішнім системам інтегруватися з платформою через три типи кінцевих точок: REST, MQTT та WebSocket. REST-ендпоінт забезпечує взаємодію із веб-додатками, мобільними програмами і приймачами сенсорів. MQTT використовується як pub-sub канал для сенсорних систем, а WebSocket – як додатковий інтерфейс для інтерактивного обміну даними.

Система також містить інтеграцію з сервісом Firebase Cloud Messaging, що забезпечує стабільну доставку push-повідомлень у мобільні та десктопні застосунки.

Приймачі сенсорів виконують збір низькорівневих вимірювань, перетворення їх у формат JSON та передачу далі по системі. Аналогічні функції мають інтерфейси датчиків, які забезпечують взаємодію з іншими платформами. Наприклад, інтерфейс RaZberry підтримує сенсори стандарту Z-Wave та забезпечує повну передачу даних, отриманих від зареєстрованих пристроїв.

Для адміністрування платформи пропонується використання модульного веб-додатка, що містить системи управління сенсорами, інструменти візуалізації їхніх показників, керування доступом, API-ключами та дослідницькими конфігураціями. Веб-застосунок будується за допомогою сучасного фреймворку

Angular та бібліотеки інтерфейсів Bootstrap. Така архітектура дозволяє легко розширювати систему новими модулями й інтегрувати сторонні інструменти.

Крім того, інтерфейс може бути доступним із серверів, які відрізняються від того, що надає REST-ендпоінт, за умови коректного налаштування політик CORS, які зазвичай використовуються для підвищення безпеки системи.

3.2 Збір та опрацювання даних від сенсорів

Датчики взаємодіють із системою за допомогою різних комунікаційних протоколів, серед яких Bluetooth, RF-радіозв'язок, LoRaWAN, Wi-Fi, Ethernet, IEEE 802.15.4 та Z-wave (табл. 3.1). Наведений перелік демонструє модулі, здатні інтегруватися в архітектуру, але його легко доповнювати новими елементами. До потенційно сумісних пристроїв належать монітори тиску, пульсоксиметри, сенсори витоку води, смарт-годинники, ваги та інші побутові або медичні сенсори [1].

Для зручного підключення всієї інфраструктури запропоновано два універсальні інтерфейси, призначені для безперешкодного під'єднання датчиків, що працюють на двох взаємопов'язаних платформах: локальній мережі та контролері Z-wave RaZberry. Останній функціонує як приймальний вузол, що фіксує сигнали сенсорів, котрі передають дані через протокол Z-wave.

Підтримка обладнання досягається завдяки двом ключовим підходам:

- схеми формування глобальних унікальних ідентифікаторів (UUID);
- застосуванню єдиного структурованого формату опису датчика на основі принципів уніфікованого читання.

Механізм присвоєння UUID гарантує правильне розрізнення потоків даних і запобігає помилковому змішуванню інформації з різних джерел. Генерація таких ідентифікаторів здебільшого здійснюється програмними засобами приймачів, які під час реєстрації створюють відповідні метадані. Після занесення сенсорів у систему приймачі можуть передавати їхні вимірювання у встановленому форматі.

Таблиця 3.1 – Сенсори для системи розумного будинку

Тип датчика	Виробники	Протокол зв'язку
Акселерометр	Bosch Microchip Sun Microsystems Texas Instruments	I2C з Wi-Fi LoRaWAN IEEE 802.15.4 Bluetooth
Якості повітря	Elsys	LoRaWAN
Аналогова напруга	Adeunis	LoRaWAN
Bluetooth маяк	Various	LoRaWAN
Контактний датчик	Everspring Nexa Tynetec	Z-Wave RF (433MHz) RF (169MHz)
GPS	Adeunis RF GlobalSat Ulster University	LoRaWAN Wi-Fi/4G (через додаток)
Вологості повітря	Adeunis RF Microchip Texas Instruments Ulster University	LoRaWAN Bluetooth Wi-Fi
Інтенсивності світла	Sun Microsystems Texas Instruments	IEEE 802.15.4 Bluetooth
Магнітометри	Bosch Texas Instruments	Wi-Fi/Bluetooth Bluetooth
Теги NFC	Various	Wi-Fi/4G/Ethernet
Пасивні інфрачервоні датчики руху	Belkin Elsys Nexa	Wi-Fi LoRaWAN RF (433MHz)
Споживання електроенергії	Belkin NKE Wattecco	Wi-Fi LoRaWAN
Натискання кнопки	FLIC	Bluetooth
Розумна підлога	Future-Shape GMBH	RF (868MHz)
Звуковий тиск	Ulster University	Wi-Fi/Bluetooth
Температура	Adeunis RF Microchip Sun Microsystems Texas Instruments Ulster University	LoRaWAN IEEE 802.15.4 Bluetooth Wi-Fi
Теплове бачення	Heimann GMBH IOTTech Ulster University	Ethernet USB Wi-Fi/Bluetooth

Метадані, як і потокові показники, передаються у JSON-структурі. Результати експериментальних вимірювань об'єднуються в один JSON-файл для спрощення обміну. Отриманий масив даних доступний через REST-інтерфейс. На першому етапі впроваджується JSON-парсер, який здійснює читання інформації, сформованої під час обробки сенсорних показників.

У наборі міститься 24 типи людської поведінки (табл. А.1, додаток А), і система здатна визначати лише один вид активності за раз. Процес маркування виконуваних дій відбувається через NFC-мітки, які зчитуються смартфоном. Застосування цієї технології дозволяє фіксувати початок та завершення кожної активності.

NFC є короткодіапазонною бездротовою технологією, що забезпечує обмін невеликими пакетами даних через індуктивний зв'язок між двома пристроями, розташованими поруч. Коли зчитувач із власним живленням наближається до пасивної мітки, у ній індукується енергія, після чого мікрочіп передає збережену інформацію назад на пристрій. Електромагнітне поле створюється котушкою зчитувача, а при наближенні тегу формується індуктивна взаємодія згідно з законом Фарадея. Після активації мітки зчитувач може фіксувати зміни поля, спричинені модуляцією сигналу NFC-тегом.

Наступний етап – сегментація даних, тобто поділ суцільної послідовності подій на рівномірні часові відрізки. Так утворюються послідовності часових рядів, що дає можливість застосовувати алгоритми класифікації вже в реальному часі. За аналізом тривалості дій і наборів характеристик у початковій вибірці формується база закономірностей, які підвищують імовірність визначення виконаної діяльності за поточними значеннями.

Подальший алгоритм працює як динамічна система класифікації: враховується ймовірність переходів між активностями та ймовірність надходження подій певного типу. Якщо поведінка не відповідає жодному з 24 попередньо визначених класів, у додатку створюється окрема категорія під назвою «Без діяльності», яка згодом відкидається під час препроцесингу.

Класифікатор визначає активність на основі формалізованого набору ознак. Бінарні параметри кодуються 0 або 1, часові сегменти – числами від 0 до 2, а інші характеристики отримують власні цифрові значення.

Дані прискорення надходять зі смарт-годинника, що містить акселерометр, який обчислює величину прикладеного прискорення. Інформація про відстань або наближення формується вимірювачами RSSI – показником рівня сигналу, який

завжди від'ємний. Чим ближче значення до нуля, тим кращий зв'язок: наприклад, -50 означає сильний сигнал, а -100 – практично відсутній. Показники прискорення та RSSI обов'язково нормалізуються, як було описано раніше.

Зібраний та опрацьований датасет містить записи ознак і значення відповідної активності. Для прикладу: система зафіксувала перебування мешканця на кухні завдяки сенсорам підлоги, спрацювання магнітного датчика чайника, ранішній час доби та горизонтальний рух за акселерометром — такий набір відповідає класу A_2 «Приготування сніданку». Модель навчається на подібних зразках. Надалі застосовується k-кратна схема перехресної валідації для поділу даних на навчальну й тестову частини [1].

3.3 Класифікація активності та оцінка ефективності системи

У запропонованій системі застосовано підхід контрольованого машинного навчання. У цьому процесі модель отримує зворотний сигнал, який повідомляє, чи були передбачення коректними, чи вони містили помилки. Набір даних має чіткі позначки класів, завдяки яким алгоритм може обчислити величину похибки для кожного зробленого припущення [1].

Формування моделі відбувається через побудову функції, що здатна навчатися на попередньо підготовленому масиві прикладів і надалі використовуватися для інтерпретації нових, раніше невідомих даних. Головна мета полягає у створенні узагальнювальної функції, яка надійно працюватиме навіть у тих випадках, коли їй подають інформацію, якої не було у навчальному наборі [1].

Після виконання підготовчого етапу дані поділяють на дві частини: тренувальну та тестову. На першій частині модель поступово вдосконалюється, коригуючи параметри доти, доки не досягне потрібного рівня якості. Це здійснюється для кожного прикладу окремо: помилка, тобто різниця між фактичним і очікуваним результатом, використовується для оновлення параметрів

алгоритму. Далі отриману модель перевіряють на другій частині даних, яка не була доступною під час навчання [1].

Тестовий набір відіграє важливу роль, оскільки демонструє здатність моделі правильно класифікувати нові випадки та слугує об'єктивним критерієм якості прогнозування. Після проходження процедури тренування класифікатор може визначати вид активності людини на основі поданих ознак. Результати класифікації відображають у матриці невідповідностей, що дає можливість оцінити ефективність роботи алгоритму (табл. 3.2).

Таблиця 3.2 – Матриця невідповідності

		Реальний клас	
		Позитивний (P)	Негативний (N)
Прогнозований клас	Позитивний (P)	Істинно позитивний (TP)	Хибно позитивний (FP)
	Негативний (N)	Хибно негативний (FN)	Істинно негативний (TN)

У матриці відображаються такі показники: істинно позитивні – кількість позитивних прикладів, які алгоритм ідентифікував правильно; істинно негативні – кількість випадків, коректно віднесених до негативного класу; хибно позитивні – ситуації, коли негативні приклади були помилково класифіковані як позитивні; хибно негативні – позитивні приклади, визначені як негативні. На підставі цих значень обчислюють точність – один із найпоширеніших критеріїв якості класифікаторів [1]:

$$acc = \frac{TN + TP}{TP + TN + FP + FN}. \quad (3.1)$$

Крім того, знаходять позитивне прогнозоване значення, яке показує частку правильно визначених позитивних випадків серед усіх прикладів, що модель визнала позитивними [1]:

$$rec = \frac{TP}{TP + FN}. \quad (3.2)$$

Окремо визначають справжній позитивний коефіцієнт – відношення правильно класифікованих позитивних прикладів до їх загальної кількості, що дає змогу оцінити здатність алгоритму виявляти потрібні класи [1]:

$$pr = \frac{TP}{TP + FP} \cdot \quad (3.3)$$

На основі цих двох показників обчислюється F1-міра, яка поєднує їх у єдиний інтегральний критерій і відображає збалансованість точності та повноти:

$$F1 = 2 \cdot \frac{pr \cdot rec}{pr + rec} \cdot \quad (3.4)$$

Графічне представлення результатів розпізнавання активностей методом Naive Bayes наведено на рисунку 3.2. Значення по горизонталі відповідають передбаченим варіантам дій, а по вертикалі – реальним класам, визначеним алгоритмом. Діагональний ряд демонструє правильно класифіковані випадки. Для більшості типів активностей система показала високий рівень точності [1].

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Сума	TP	FP	FN	TN	pr	rec	F1	acc	
1	152	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	152	152	0	0	0	100,00	100	100,00	100,00	
2	0	145	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	148	145	3	0	0	97,97	100	98,98	97,97	
3	0	0	153	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	161	153	8	0	0	95,03	100	97,45	95,03	
4	0	0	0	139	0	0	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	177	139	38	0	0	78,53	100	87,97	78,53	
5	0	18	0	0	143	0	12	0	0	0	0	0	0	0	6	0	0	0	0	0	0	5	0	0	184	143	41	0	0	77,72	100	87,46	77,72	
6	5	0	22	0	0	99	0	0	0	0	0	0	0	0	0	0	0	18	0	0	0	0	0	0	144	99	45	0	0	68,75	100	81,48	68,75	
7	42	0	0	14	0	0	130	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	190	130	60	0	0	68,42	100	81,25	68,42	
8	0	0	0	0	33	0	0	153	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	188	153	35	0	0	81,38	100	89,74	81,38	
9	0	0	0	0	0	0	0	0	58	0	35	0	0	0	0	0	0	0	0	0	0	0	31	0	124	58	66	0	0	46,77	100	63,74	46,77	
10	0	0	0	0	0	0	4	0	123	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	127	123	4	0	0	96,85	100	98,40	96,85	
11	0	0	0	0	0	0	0	0	0	152	0	0	0	0	0	3	0	0	0	0	15	0	0	0	170	152	18	0	0	89,41	100	94,41	89,41	
12	0	0	0	0	0	24	0	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0	0	144	120	24	0	0	83,33	100	90,91	83,33	
13	0	0	0	0	0	0	0	0	0	21	0	0	139	0	0	0	0	0	0	0	0	0	0	0	160	139	21	0	0	86,88	100	92,98	86,88	
14	0	0	0	0	0	0	0	0	0	0	0	0	115	0	0	0	0	0	0	0	0	10	0	0	125	115	10	0	0	92,00	100	95,83	92,00	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0	0	0	2	0	0	0	0	0	0	125	123	2	0	0	98,40	100	99,19	98,40	
16	0	0	0	1	0	0	0	0	0	0	0	0	0	0	26	0	0	0	0	0	0	0	0	0	27	26	1	0	0	96,30	100	98,11	96,30	
17	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	120	0	0	0	0	0	0	0	0	127	120	7	0	0	94,49	100	97,17	94,49	
18	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	120	0	0	0	0	0	0	0	123	120	3	0	0	97,56	100	98,77	97,56	
19	3	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	153	0	0	0	0	0	176	153	23	0	0	86,93	100	93,01	86,93	
20	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	26	89	0	27	0	0	144	89	55	0	0	61,81	100	76,39	61,81	
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	60	0	0	0	61	60	1	0	0	98,36	100	99,17	98,36	
22	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	151	0	0	156	151	5	0	0	96,79	100	98,37	96,79	
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	133	0	168	133	35	0	0	79,17	100	88,37	79,17
24	0	0	0	0	0	24	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	169	140	29	0	0	82,84	100	90,61	82,84	

Рисунок 3.2 – Матриця невідповідності результатів тестування

Найменш достовірними виявилися результати для видів діяльності під номерами 9 і 20 – «Перегляд телевізора» та «Прання». Перша активність часто плуталася з «Грою у відеоігри», оскільки в обох випадках активними були однакові сенсори вітальні – телевізор, підлога, диван. Аналогічно, категорія «Прання» іноді сприймалася як «Миття посуду» через розташування пральної машини на кухні або як «Переодягання» – через близькість до корзини для білизни. Найкращі показники, що сягнули 98-100%, отримано для класів 1, 15 і 21 – «Прийом ліків», «Викидання сміття» та «Робота за столом», адже ці типи активностей мають унікальні набори ознак, які майже не перетинаються з іншими діями [1].

На основі наведених прикладів можна дійти висновку, що алгоритм функціонує стабільно, забезпечує правильне шифрування та успішне відновлення вихідних даних. Інформацію, зашифровану цим методом, можна безпечно передавати відкритими каналами зв'язку, адже спроби відновлення вмісту за допомогою криптоаналізу є надзвичайно складними для сучасних методів і технічних засобів, на яких здійснюється обчислення задач цього типу.

3.4 Висновок до розділу

У ході дослідження було розроблено інтелектуальну систему підтримки рішень для керування елементами «розумного дому» на основі визначення щоденної активності людини. Відмінністю цієї системи є використання сучасних методів машинного навчання, що дозволило значно підвищити рівень автоматизації та точності, оптимізувати використання енергії і зробити взаємодію з домашніми пристроями більш адаптивною до індивідуальних потреб користувача. Результати свідчать, що інтеграція інтелектуальних алгоритмів у системи розумного житла підвищує їхню ефективність та відкриває можливості для подальшої персоналізації та розвитку.

ВИСНОВКИ

У процесі підготовки та написання магістерської роботи було розроблено модуль інтелектуальної системи підтримки ухвалення рішень щодо роботи складових розумного будинку. Його функціонування ґрунтується на розпізнаванні дій і поведінкових патернів людини у середовищі “smart home”.

У першому розділі подано огляд концепції Інтернету речей, наведено узагальнену структуру типової IoT-системи. Також проведено аналіз наявних підходів до обробки даних для визначення людської активності. Розглянуто сильні та слабкі сторони сучасних рішень із використанням алгоритмів штучного інтелекту в галузі розпізнавання життєдіяльності в контексті IoT-інфраструктури.

Другий розділ містить опис предметної сфери та формулювання ключових вимог до майбутньої системи. Подано модель процедури класифікації повсякденних дій користувача в розумному житлі. Ця модель ґрунтується на алгоритмі NBC і надає можливість передбачати найбільш ймовірну активність мешканця на основі аналізу набору вхідних характеристик.

У третьому розділі викладено метод виявлення людської поведінки у “smart home”, який використовує байєсівські мережі. Первинні дані було поділено на тренувальний та тестовий сегменти, сформульовано покроковий алгоритм класифікації. Після застосування підходу система визначає тип виконуваної людиною діяльності. Перевага такого методу полягає у його стійкості та здатності забезпечувати достатньо коректний результат незалежно від масштабів даних.

У четвертому розділі, спираючись на попередньо сформовані модель і методи розпізнавання активностей, розроблено покращену інтелектуальну систему ухвалення рішень щодо управління різними елементами розумного будинку. Удосконалена система застосовує алгоритми машинного навчання, що дозволяє збільшити рівень автоматизації процесів. Вона може адаптувати параметри роботи “smart”-пристроїв під поведінку користувачів, а також оптимізувати енергоспоживання завдяки аналізу виявлених типів діяльності.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1 Стрелкова О. Г. Інтелектуальна система прийняття рішення щодо функціонування елементів розумного будинку на основі класифікації видів діяльності людини : кваліфікаційна робота за освітнім рівнем «магістр» : 123 Комп'ютерна інженерія / Стрелкова О. Г. – Хмельницький, 2022. – 97 с.
- 2 Karvonen, Niklas and Denis Kleyko. A Domain KnowledgeBasedSolution for Human Activity Recognition: The UJA Dataset Analysis. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1261. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191261> (дата звернення: 14.10.2025)
- 3 Lago, Paula and Sozu Inoue. A Hybrid Model Using Hidden Markov Chain and Logic Model for Daily Living Activity Recognition. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1266. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191266> (дата звернення: 14.10.2025)
- 4 Salomon, Sergio and Cristina Tirnauce. Human Activity Recognition through Weighted Finite Automata. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1263. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191263> (дата звернення: 14.10.2025)
- 5 Jimenez, Antonio R. and Fernando Seco. Multi-Event Naive Bayes Classifier for Activity Recognition in the UCAmI Cup. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1264. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191264> (дата звернення: 14.10.2025)

- 6 Razzaq, Muhammad Asif et al. Multimodal Sensor Data Fusion for Activity Recognition Using Filtered Classifier. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1262. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191262> (дата звернення: 15.10.2025)
- 7 Ceron, Jesus, Diego Lopez, and Bjoern Eskofier. Human Activity Recognition Using Binary Sensors, BLE Beacons, an Intelligent Floor and Acceleration Data: A Machine Learning Approach. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1265. Punta Cana, Dominican Republic: MDPI. URL : <https://doi.org/10.3390/proceedings2191265> (дата звернення: 15.10.2025)
- 8 Wirth, Rudiger and Jochen Hipp. CRISP-DM: Towards a Standard Process Model for DataMining. Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining, pp. 29–39.
- 9 Hammerla, Nils Y. and Shane Halloran Thomas Plotz. Deep, Convolutional, and Recurrent Models for Human Activity Recognition using Wearables. Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16), pp. 1533–1540, 2016. URL : <https://www.ijcai.org/Proceedings/16/Papers/220.pdf> (дата звернення: 24.10.2025)
- 10 Espinilla, Macarena, Javier Medina, and Chris Nugent. UCAmI Cup. Analyzing the UJA Human Activity Recognition Dataset of Activities of Daily Living. The 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018). Vol. 2. 19 1267. Punta Cana, Dominican, Dec. 2018. URL : <https://www.mdpi.com/2504-3900/2/19/1267> (дата звернення: 04.11.2025)
- 11 Priyadarshini, Sushree Bibhuprada B., Amiya Bhusan Bagjadab, and Brojo Kishore Mishra. The Role of IoT and Big Data in Modern Technological Arena: A Comprehensive Study. Intelligent Systems Reference Library. Springer International Publishing. Chap. 2, vol. 154, pp. 13–25, 2019. isbn:

9783030042035. URL : https://link.springer.com/chapter/10.1007/978-3-030-04203-5_2 (дата звернення: 05.11.2025)
- 12 Pantazis, N. A., Stefanos, A. N., Dimitrios, D. V. Energy-efficient routing protocols in wireless sensor networks: A survey. *IEEE Commun. Surveys Tutor.* 2013, 15, 551–591.
- 13 Liu, X. A survey on clustering routing protocols in wireless sensor networks. *Sensors* 2012, 12, 11113–11153.
- 14 De Silva, L.C.; Morikawa, C.; Petra, I.M. State of the art of smart homes. *Eng. Appl. Artif. Intell.* 2012, 25, 1313–1321.
- 15 Zhang, D.; Shah, N.; Papageorgiou, L.G. Efficient energy consumption and operation management in a smart building with microgrid. *Energy Convers. Manag.* 2013, 74, 209–222.
- 16 Pedrasa, M.; Spooner, T.; MacGill, I. Coordinated Scheduling of Residential Distributed Energy Resources to Optimize Smart Home Energy Services. *IEEE Trans. Smart Grid* 2010, 1, 134–143.
- 17 United States Department of Commerce, National Telecommunications and Information Administration, United States Department of Commerce. URL : <http://www.its.bldrdoc.gov> (дата звернення: 04.11.2025)
- 18 Gu, H.; Diao, Y.; Liu, W.; Zhang, X. The design of smart home platform based on Cloud Computing. *Proceeding of the International Conference on the Design of Smart Home Platform Based on Cloud Computing, Harbin, China, 12–14 August 2011.*
- 19 Płaczek, B.; Bernaś, M. Uncertainty-based information extraction in wireless sensor networks for control applications. *Ad Hoc Netw.* 2014, 14, 106–117.
- 20 Saponara, S.; Bacchillone, T. Network Architecture, Security Issues, and Hardware Implementation of a Home Area Network for Smart Grid. *J. Comput. Netw. Commun.* 2012
- 21 Gonzalez-Tablas, A.; Ferreres, A.I.; Ramos Alvarez, B.; Garnacho, A. Guaranteeing the Authenticity of Location Information. *IEEE Pervasive Comput.* 2008, 7, 72–80.

- 22 Xing, F.; Wang, W. Analyzing Resilience to Node Misbehaviors in Wireless Multi-Hop Networks. Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC 2007, Kowloon, Hong Kong, China, 11–15 March 2007.
- 23 Liu, X. A survey on clustering routing protocols in wireless sensor networks. *Sensors* 2012, 12, 11113–11153.
- 24 Wang, P.; Yao, C.; Zheng, Z.; Sun, G.; Song, L. Joint Task Assignment, Transmission, and Computing Resource Allocation in Multilayer Mobile Edge Computing Systems. *IEEE IoT J.* 2018, 6, 2872–2884.
- 25 Ren, J.; He, Y.; Huang, G.; Yu, G.; Cai, Y.; Zhang, Z. An edgecomputing based architecture for mobile augmented reality. *IEEE Netw.* 2019, 33, 162–169.
- 26 Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000.
- 27 Younis, O.; Sonia, F. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Trans. Mobile Comput.* 2004, 3, 366–379.
- 28 Cherchi, R.; Colistra, G.; Pilloni, V.; Atzori, L. Energy consumption management in Smart Homes: An M-Bus communication system. Proceedings of the 2014 International Conference on Telecommunications and Multimedia (TEMU), Heraklion, Greece, 28–30 July 2014
- 29 Hu, Q.; Li, F. Hardware Design of Smart Home Energy Management System with Dynamic Price Response. *IEEE Trans. Smart Grid* 2013, 4, 1878–1887.
- 30 Jo, H.; Kim, S.; Joo, S. Smart heating and air conditioning scheduling method incorporating customer convenience for home energy management system. *IEEE Trans. Consum. Electron.* 2013, 59, 316–322.
- 31 Peruzzini, M.; Germani, M.; Papetti, A.; Capitanelli, A. Smart Home Information Management System for Energy-Efficient Networks.

- Collaborative Systems for Reindustrialization; Springer: Berlin, Germany; Heidelberg, Germany; Dresden, Germany, 2013; Volume 408, pp. 393–401.
- 32 Han, D.M.; Lim, J.H. Smart home energy management system using IEEE 802.15.4 and ZigBee. *IEEE Trans. Consum. Electron.* 2010, 56, 1403–1410.
- 33 Han, D.M.; Lim, J.H. Design and implementation of smart home energy management systems based on ZigBee. *IEEE Trans. Consum. Electron.* 2010, 56, 1417–1425.
- 34 R. Want. An introduction to RFID technology. *IEEE Pervasive Computing*, vol. 5, no. 1, pp. 25–33, Jan 2006.
- 35 L. Atzori, A. Iera, and G. Morabito. The Internet of Things: A Survey. *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- 36 M. Friedli, L. Kaufmann, F. Paganini, and R. Kyburz. Energy Efficiency of the Internet of Things - Technology and Energy Assessment Report. International Energy Agency, vol. 1, 2016.
- 37 M. S. Mahdavinejad, M. Rezvan, M. Barekatin, P. Adibi, P. Barnaghi, A. P. Shath. Machine learning for Internet of Things data analysis: A survey. *Digital Community Network*, pp. 163-167, 2017.
- 38 E. M. Tapia, S. S. Intille, K. Larson. Activity recognition in the home using simple and ubiquitous sensors. *Pervasive*, vol. 4, pp. 158-175, 2004.
- 39 D. Pavithra, R. Balakrishnan. IoT based monitoring and control system for home automation. *Global Conference on Communication Technologies-GCCT*, pp. 169-173, Apr 2015.
- 40 M. Pandurov, B. Petelj, R. Pavlovic, N. Teslic. Platform for extending home automation gateway's functionality with plugin mechanism. *IEEE 5th International Conference on Consumer Electronics*, pp. 354-357, 2015.
- 41 J. Wei. How Wearables Inersect with the Cloud and the Internet of Things: Considerations for the developers of wearables. *IEEE Consumer Electronics Magazine*, 3(3), 53-56, 2014.

- 42 J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), pp. 1645-1660, 2013.
- 43 N. Li, N. Zhang, S. Das, B. Thuraisingham. Privacy preservation in wireless sensor networks: A state-of-the-art survey. *Ad Hoc Networks*, vol. 7, no. 8, pp. 1501-1514, 2009.
- 44 P. Kinney. Zigbee technology: Wireless control that simply works. *Communications design conference*, vol. 2, pp. 1-7, October, 2003.
- 45 D. Minoli, K. Sohraby, B. Occhiogrosso. IoT Considerations, Requirements, and Architectures for Smart Buildings – Energy Optimization and Next-Generation Building Management Systems. *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 269-283, Feb. 2017.
- 46 J. Wan et al. Software-Defined Industrial Internet of Things in the Context of Industry 4.0. *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7373-7380, Oct. 15, 2016.
- 47 C. C. Aggarwal, N. Ashish, A. Sheth. The Internet of Things: A Survey from the Data-Centric Perspective. *Managing and Mining Sensor Data*, Springer US, pp. 383-428, 2013.
- 48 D. Miorandi, S. Sicari, F. De Pellegrini, I. Chlamtac. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497-1516, 2012.
- 49 M. Palattela, N. Accettura, X. Vilajosana, T. Watteyne, L. Grieco, G. Boggia, M. Dohler. Standardized Protocol Stack for the Internet of (Important) Things. *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1389-1406, 2013.
- 50 A. Jara, A. Olivieri, Y. Bocchi, M. Jung, W. Kastner, A. Skarmeta. Semantic Web of Things: an analysis of the application semantics for the IoT moving towards the IoT convergence. *International Journal of Web and Grid Services*, vol. 10, no. 23, p. 244, 2014.

ДОДАТКИ

Таблиця А.1 – Види активності людини у розумному будинку

Активність	Назва активності	Сценарій
A_1	Приймання ліків	Піти на кухню – взяти пляшку з водою – дістати ліки з коробки – проковтнути ліки
A_2	Приготування сніданку	Піти на кухню – взяти продукти для сніданку – розмістити їжу на столі. (Може включати: заварювання чаю з застосуванням електричного чайника або приготування гарячого шоколаду з молоком, використовуючи мікрохвильову піч)
A_3	Приготування обіду	Піти на кухню – дістати деякі продукти з холодильника – розмістити їжу на столі. (Може включати: приготування гарячої їжі на плиті або нагрівання їжі швидкого приготування в мікрохвильовій печі)
A_4	Приготування вечері	Піти на кухню – дістати деякі продукти з холодильника – розмістити їжу на столі. (Може включати: приготування гарячої їжі на плиті або нагрівання їжі швидкого приготування в мікрохвильовій печі)
A_5	Сніданок	Піти на кухню зранку – сісти за стіл для прийому їжі – поїсти – прибрати посуд в раковину або посудомийну машину
A_6	Обід	Піти на кухню в обід – сісти за стіл для прийому їжі – поїсти – прибрати посуд в раковину або посудомийну машину
A_7	Вечеря	Піти на кухню ввечері – сісти за стіл для прийому їжі – поїсти – прибрати посуд в раковину або посудомийну машину
A_8	Перекус	Піти на кухню – взяти фрукт або снєк – з'їсти на кухні або у вітальні. (Може включати: прибирання посуду в раковину або посудомийну машину)
A_9	Перегляд телевізора	Піти у вітальню – взяти пульт – сісти на диван – подивитись телевізор – залишити пульт біля телевізора
A_{10}	Вхід до будинку	Зайти до будинку, використовуючи вхідні двері – залишити ключ в спеціальній корзині
A_{11}	Гра у відеоігри	Піти у вітальню – взяти пульт від телевізора та джойстик від ігрової приставки – сісти на дивані – пограти – встати з дивану – покласти пульт і джойстик біля телевізора
A_{12}	Відпочинок на дивані	Піти у вітальню – сісти на дивані – посидіти декілька хвилин – встати з дивану

Активність	Назва активності	Сценарій
A_{13}	Вихід з будинку	Підійти до вхідних дверей – відкрити вхідні двері – вийти з будинку – закрити двері
A_{14}	Відвідування мешканця будинку (гість)	Підійти до вхідних дверей – відкрити двері – поговорити з кимось біля дверей – закрити двері
A_{15}	Викидання сміття	Піти на кухню – взяти сміття з відра для сміття – взяти ключі зі спеціальної корзини при вході – вийти з будинку – повернутися – залишити ключі в корзині
A_{16}	Миття рук	Піти у ванну кімнату – відкрити кран – намилити руки – промити руки – закрити кран – висушити руки
A_{17}	Чистка зубів	Піти у ванну кімнату – відкрити кран – почистити зуби – закрити кран
A_{18}	Використання туалету	Піти у ванну кімнату – підняти кришку унітазу – використати туалет – опустити кришку унітазу – натиснути на зливний бачок
A_{19}	Миття посуду	Піти на кухню – покласти брудний посуд в посудомийну машину – запустити – покласти посуд на місце
A_{20}	Прання	Піти у спальню – взяти корзину для білизни – піти на кухню – покласти одяг в пральну машину – зачекати кінця прання – витягнути одяг з пральної машини – покласти випраний одяг до шафи
A_{21}	Робота за столом	Сісти на робочому місці – зробити роботу – встати після завершення
A_{22}	Переодягання	Піти до спальні – покласти брудний одяг у корзину для білизни – відкрити шафу – одягнути чистий одяг – зачинити шафу
A_{23}	Відхід до сну	Піти у спальню – лягти у ліжко – заснути
A_{24}	Пробудження	Прокинутись – встати з ліжка

БІБЛІОГРАФІЧНА ДОВІДКА

Тема дипломної роботи: "Використання байєсового класифікатора для інтелектуального модуля ідентифікації дій людини в контексті технологій Smart House"

Обсяг пояснювальної записки 59 аркушів.

14 рисунки;

2 таблиці;

1 додатки.

Дата завершення роботи: *10 грудня 2025 р.*

Підпис студента-дипломника _____ *Регнер В. В.*