НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ
КАФЕДРА ПРОМИСЛОВОЇ ЕЛЕКТРОНІКИ

«На правах рукопису»
УДК 004.942:519.876

«До захисту допущено»
Завідувач кафедри
_____ Ю.С. Ямненко
(підпис) (ініціали, прізвище)
“_____” _____2018 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 171 Електроніка
(код і назва)

спеціалізації Електронні компоненти і системи

на тему: Виявлення аномалій у MicroGrid методами машинного навчання

Виконав: студент II курсу, групи ДС-71мп
(шифр групи)

Комаревич Олександр Миколайович _____ _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник д.т.н., проф. Ямненко Ю.С. _____ _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент ст. викл. Порєва Г.С. _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2018 року

Національний технічний університет України
“Київський політехнічний інститут
імені Ігоря Сікорського”

Факультет електроніки
(повна назва)

Кафедра промислової електроніки
(повна назва)

Рівень вищої освіти – другий (магістерський) за освітньо - професійною програмою

Спеціальність 171 Електроніка
(шифр і назва)

Спеціалізація Електронні компоненти і системи

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Ю.С. Ямненко
(підпис) (прізвище ініціали)
«07» листопада 2018 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ

Комаревичу Олександру Миколайовичу
(прізвище, ім’я, по батькові)

1. Тема проекту «Виявлення аномалій у MicroGrid методами машинного навчання» науковий керівник дисертації Ямненко Юлія Сергіївна, д.т.н., проф.
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від « 07 » листопада 2018 року № 4114-с
2. Строк подання студентом проекту 06 грудня 2018 року
3. Об’єкт дослідження процеси, пов’язані з активністю людини у MicroGrid, що можуть бути зафіксованими відповідними датчиками .
4. Предмет дослідження (вихідні дані для магістерської дисертації за освітньо-професійною програмою) є виявлення аномалій поведінки людини за допомогою методів Machine Learning .
5.Перелік завдань, які потрібно розробити 1. Проаналізувати основні методи Anomaly Detection в Machine Learning та обрати один із методів Machine Learning. 2. Розглянути квартиру з 7 приміщеннями, 32 датчиками та 3 типи аномальної поведінки. 3. Розробити алгоритм пошуку аномальної поведінки людини в MicroGrid. 4. Розробити програмне забезпечення для пошуку аномальної активності людини в MicroGrid
6. Перелік графічного (ілюстративного) матеріалу Програмне забезпечення для пошуку аномальної активності людини в MicroGrid, слайди презентації.

7. Перелік публікацій

1) Ямненко Ю. С., Моргун А. В., Комаревич О. М., Програмне забезпечення для макромоделювання системи керування MicroGrid / Electronics and communications. - 2016. - Т. 21, № 6. - С. 61 - 66. - Режим доступу: http://nbuv.gov.ua/UJRN/eisv_2016_21_6_10 .

2) Комаревич О. М., Хохлов Ю.В., Ямненко Ю.С. Виявлення аномальної поведінки людини у MicroGrid на базі машинного навчання / Мікросистеми, Електроніка та Акустика. – 2018. – Т. 23, N 4. - С. 36-41. DOI: 10.20535/2523-4455.2018.23.4.143310.

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання «03» вересня 2018 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строки виконання етапів магістерської дисертації	Примітка
1	Огляд літератури	03.09.2018-20.09.2018	виконано
2	Аналіз методів Anomaly Detection в Machine Learning	21.09.2018-10.10.2018	виконано
3	Розробка алгоритму виявлення аномалій на базі методу SVM	11.10.2018-24.10.2018	виконано
4	Розробка програмного забезпечення для пошуку аномальної активності людини в MicroGrid	28.10.2018-28.11.2018	виконано
5	Розробка стартап-проекту	29.11.2018 – 04.12.2018	виконано

Студент _____ Комаревич О.М.
(підпис) (ініціали, прізвище)

Науковий керівник дисертації _____ Ямненко Ю.С.
(підпис) (ініціали, прізвище)

АНОТАЦІЯ

В магістерській дисертації розглядається задача застосування Anomaly Detection як один із засобів Machine Learning для визначення нетипових моделей поведінки людини (аномальної поведінки) у MicroGrid. Наукова новизна отриманих результатів полягає у отриманні подальшого розвитку теорії застосування методів машинного навчання для обробки та аналізу великих даних (Big Data) в частині виявлення аномалій поведінки людини у MicroGrid. Вперше запропоновано розв'язання задачі пошуку аномалій різних поведінкових моделей на базі застосування методу опорних векторів (SVM). За допомогою розробленого програмного забезпечення користувач може переглядати інтервали часу обраного дня та аналізувати випадки фіксування аномалій.

Ключові слова: MicroGrid, Big Data, Machine Learning, Anomaly Detection.

АНОТАЦИЯ

В магистерской диссертации рассматривается задача применения Anomaly Detection как одного из средств Machine Learning для определения нетипичных моделей поведения человека (аномального поведения) в MicroGrid. Научная новизна полученных результатов заключается в дальнейшем развитии теории применения методов машинного обучения для обработки и анализа больших данных (Big Data) в части выявления аномалий поведения человека в MicroGrid. Впервые предложено решение задачи поиска аномалий различных поведенческих моделей на базе применения метода опорных векторов (SVM). С помощью разработанного программного обеспечения пользователь может просматривать интервалы времени избранного дня и анализировать случаи фиксирования аномалий.

Ключевые слова: MicroGrid, Big Data, Machine Learning, Anomaly Detection.

ANNOTATION

In the master's thesis the problem of using Anomaly Detection as one of the means of Machine Learning for the determination of non-typical behavior patterns (abnormal behavior) in MicroGrid is considered. The scientific novelty of the results is further development of machine learning methods for the processing and analysis of Big Data in terms of identifying human behavior anormalies in MicroGrid. For the first time the solution of the problem of finding anomalies of different behavioral models based on the application of the method of support vector machine (SVM) is proposed. With the help of the software developed, the user can view the time intervals of the selected day and analyze the cases of fixing anomalies.

Keywords: MicroGrid, Big Data, Machine Learning, Anomaly Detection.

ЗМІСТ

СТ.

ВСТУП

Актуальність теми. Розвиток електронних систем, інтелектуальних засобів керування та обміну інформацію призвів до формування нової інформаційно-енергетичної концепції SmartGrid, в рамках якої розглядаються локальні електротехнічні об’єкти – MicroGrid, що містять певний набір джерел та навантажень, які, як правило, підключені до централізованої електромережі. В рамках MicroGrid постає цілий ряд задач керування та узгодженого функціонування пристроїв, що призводить до необхідності оперування великими інформаційними потоками.

У галузі електроенергетики відбувається активне впровадження «інтелектуальних мереж» SmartGrid та MicroGrid. Значний вклад в розвиток інтелектуальних електричних мереж внесли академіки НАН України Б.С. Стогній, О.В. Кириленко, професори Денисюк С.П., Жуйков В.Я, Клен К.С. У їх працях відображено стан та задачі розвитку інтелектуальних електричних мереж в Україні. Проте питання аналізу та обробки великих інформаційних потоків (Big Data) в цих працях не розглядалося. Тому тематика досліджень, присвячена обробці великих даних у MicroGrid та виявленню аномалій за допомогою Machine Learning, є актуальною.

Зв’язок роботи з науковими програмами, планами, темами. Дослідження за темою магістерської дисертації проводилася у відповідності до наукових напрямків кафедри промислової електроніки КПІ ім. І. Сікорського.

Мета і завдання досліджень. Метою даної роботи є подальший розвиток теорії та методів машинного навчання у застосуванні до задачі знаходження аномальної поведінки людини у MicroGrid. Для досягнення мети в роботі вирішуються такі задачі:

1. Аналіз сучасних методів виявлення аномалій за допомогою Machine Learning, вибір та обгрунтування методу дослідження.

2. Дослідження варіантів аномальних поведінкових моделей у MicroGrid.

3. Побудова алгоритму пошуку аномалій у Microgrid, вибір основних параметрів.

4. Розробка програмного забезпечення для пошуку аномалій у MicroGrid.

5. Тестування, аналіз результатів, оцінка точності пошуку аномалій, модифікація програми до прототипу

Об'єктом дослідження є процеси, пов'язані з активністю людини у MicroGrid, що можуть бути зафіксованими відповідними датчиками

Предметом дослідження є виявлення аномалій поведінки людини за допомогою методів Machine Learning.

Методи дослідження. При розв'язанні поставлених в роботі завдань був використаний класифікатор Support Vector Machines з емпіричним правилом (правило 3х сигм), за допомогою яких було навчено систему для пошуку аномалій; розробка інтерфейсу програми та функціоналу була виконана за допомогою таких мов програмування як JavaScript, використанням фреймворку React.js та верстки за допомогою HTML, CSS.

Наукова новизна отриманих результатів полягає в наступному:

1. Отримала подальший розвиток теорія застосування методів машинного навчання для обробки та аналізу великих даних (Big Data) в частині виявлення аномалій поведінки людини у MicroGrid.

2. Вперше запропоновано розв'язання задачі пошуку аномалій різних поведінкових моделей на базі застосування методу опорних векторів (SVM).

Практичне значення отриманих результатів:

1. Розроблений метод дозволяє виявити аномалії поведінки людини у MicroGrid, що важливо для попередження аварійних чи небезпечних ситуацій.

2. Розроблений алгоритм детектування аномалій оперує з великою кількістю даних у тренувальних вибірках та дає можливість врахувати суб'єктивні фактори, пов'язані з наявністю людини у MicroGrid при розробці алгоритмів та систем керування.

3. За допомогою розробленого програмного забезпечення користувач може переглядати інтервали часу обраного дня та аналізувати випадки фіксування аномалій.

Особистий внесок здобувача. Магістерська робота є узагальненням результатів теоретичних та експерементальних досліджень, проведених автором з консультаціями наукового керівника. У роботах опублікованих із співавторами, автору у [1] належить пошук та аналіз інформації щодо існуючих програмних продуктів для моделювання систем електроживлення MicroGrid, редагування, оформлення; У [2] пошук та аналіз інформації щодо методів машинного навчання для детектування аномальної активності систем електроживлення MicroGrid, редагування, оформлення, розробка Back-End та Front-End частини системи машинного навчання.

Публікації. Основний зміст дисертаційної роботи відображено у 2 наукових працях: 1. Ямненко Ю. С., Моргун А. В., Комаревич О. М. Програмне забезпечення для макромоделювання системи керування MicroGrid / Electronics and communications. - 2016. - Т. 21, № 6. - С. 61-66. - Режим доступу: http://nbuv.gov.ua/UJRN/eisv_2016_21_6_10. 2. Комаревич О. М., Хохлов Ю.В., Ямненко Ю.С. Виявлення аномальної поведінки людини у MicroGrid на базі машинного навчання / Мікросистеми, Електроніка та Акустика. – 2018. – Т. 23, N 4. - С. 36-41. DOI: 10.20535/2523-4455.2018.23.4.143310.

Структура та обсяг роботи Магістерська робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел зі 29 найменувань. Загальний обсяг магістерської роботи становить 93 сторінок, у тому числі 87 сторінок основного тексту, 27 рисунків та 4 таблиці.

РОЗДІЛ 1. АНОМАЛЬНА ПОВЕДІНКА ЛЮДИНИ У СЕРЕДОВИЩІ MICROGRID

1.1. Існуючі проекти MicroGrid з урахованням людського фактору

Одним із відомих проектів розробки розумних будинків є "ACHE MicroGrid" в Боулдері, штат Колорадо, який використовує адаптивний контроль домашнього середовища та нейронних мереж [3].

ACHE MicroGrid контролює кілька функцій будинку, включаючи температуру, освітлення та опалення, без потреби користувача їх налаштовувати. Метою ACHE MicroGrid є побудова програми, заснованої на моделі нейронної мережі для спостереженням за способом життя та бажаннями людини, а потім для навчання – для прогнозування та адаптації режимів роботи пристроїв будинку.

Проект Техаського університету MavHome [4], є розробкою будинку, який поводить себе як "раціональний агент". MavHome прагне підвищити комфортність своїх користувачів, зменшуючи витрати на експлуатацію. Раціональний агент намагається вгадувати моделі мобільності та звичаї жителів. Система MavHome базується на алгоритмі LeZi-update для відстеження користувачів та їх поведінкової активності.

У Флориді був розроблений проект "GatorTech MicroGrid". Він складається з декількох окремих інтелектуальних пристроїв, таких як ліжко, поштова скринька, підлога, вхідні двері та інші. Ці індивідуальні інтелектуальні пристрої включають датчики та виконавчі пристрої, які підключені до "операційної платформи, призначеної для оптимізації комфорту та безпеки людей похилого віку".

Розроблений у Великобританії CarerNet [5] може бути описаний як MicroGrid для покращення та підвищення якості життя людей похилого віку

та тих, хто перебуває в несприятливому становищі, використовуючи технології для підтримки їх здатності самостійно функціонувати в межах свого місцевого середовища.

CarerNet збирає дані про фізіологічний стан людини, визначає спосіб життя пацієнта та екологічну обізнаність через використання кількох пристроїв, таких як термометр, інфрачервоні (IR) датчики, електрокардіографія (ЕКГ) та ін.

У Фінляндії система особистого оздоровчого спостереження під назвою TERVA [6] дозволяє здійснювати нагляд за «змінами, пов'язаними з оздоровчою діяльністю вдома» від короткого до довгострокового періоду часу. TERVA працює на комп'ютері та підтримує зв'язок з різними вимірювальними приладами. Дані вимірюються для таких параметрів як артеріальний тиск, температура користувача, вага, інтервали серцебиття, частота дихання, рухів тощо. Крім того, поведінковий щоденник зберігається для контролю добового добробуту.

Ще один MicroGrid був розроблений у Фінляндії [7], з використанням ємнісного позиціонування в приміщенні та контактного відстеження. Мета полягала в тому, щоб визнати діяльність людини, яка живе будинку Intelligent House. Система використовувала електроди, вбудовані в підлогу, щоб виявити людину на рівні підлоги, а потім визначала взаємодію особи з предметами домашнього облаштування (стіл, ліжко, холодильник та диван).

У Ірландії був розроблений “MicroGrid Великого Північного Неба” [8]. Шістнадцять "SmartHouse" були побудовані для виявлення поведінкових моделей для людей похилого віку.

Система використовувала методи машинного навчання та розпізнавання образів для тестування та прогнозування благополуччя жителів. Система розпізнавання Aurama [9], розроблена в Нідерландах, підтримує старіння на місці та використовує кілька пристроїв, таких як фоторамка, для виявлення змін у моделях поведінки.

Іншими технологіями, що використовуються, є використання радіочастотного сигналу для виявлення присутності мешканців, а також ліжкові датчики для виявлення кількості разів, коли літня людина виходить з ліжка. Було повідомлено про три покоління Aurama, і кожне покоління було вдосконалено за результатами тестування польових робіт.

Проект SPHERE [10] стосується відстеження, мереж та навчання машин для медичного обслуговування в житлових приміщеннях. Ідея полягає в тому, щоб об’єднати дані датчиків і створити багатий набір даних (BigData) для виявлення та управління різними станами здоров'я. SPHERE використовує фізіологічний (носійний) моніторинг сигналів, моніторинг домашнього середовища та моніторинг на основі зору. Цей підхід до системи зондування мультимодальності є повністю інтегрованим з інтелектуальними алгоритмами обробки даних, що керують збором даних.

У Нідерландах система автоматизованого автономного спостереження (UAS) [11] була розроблена для підтримки старіння.

Система використовує мережу ZigBee і бездротові датчики, розташовані навколо будинку, а також використовує камери, які активуються в разі виникнення надзвичайних ситуацій. Ця система тестується дорослими людьми старшого віку в містах Баран та Суст у Нідерландах.

1.2. Джерела великих даних та їх реєстрація за допомогою датчиків

В порівнянні з повністю автоматичними системами, MicroGrid характеризується присутністю людини, що призводить до необхідності врахування суб’єктивних факторів – втручання в процес функціонування електротехнічного обладнання та вплив дій людини на робочі режими всієї системи. Наявність людського фактору призводить до необхідності оперування великими обсягами даних. В свою чергу це змушує дослідників звертатися до спеціалізованих методів роботи з великими даними (Big Data).

Велика кількість та різноманітність цих методів включає підходи, інструменти та методи обробки, ефективні в умовах безперервного зростання обсягів даних, та отримання результатів, придатних для сприйняття людиною. Важливим та ефективним інструментом обробки великих обсягів даних є методи машинного навчання (Machine Learning). Як галузь інформатики, Machine Learning використовує статистичні методи для забезпечення навчання комп'ютерних систем із поступовим покращенням продуктивності вирішення конкретних задач без явного програмування.

До аномальних режимів у MicroGrid можна віднести:

1. аварійні випадки;.

2. вихід технічних параметрів за допустимі межі;

3. незвична активність, нетипова поведінка людини – «людський фактор».

Оскільки перші два випадки відносяться до суто технічних, то їх відпрацювання здійснюється за рахунок конструкторських особливостей інженерної розробки електротехнічного комплексу і здійснюється відомими методами. Що стосується третього випадку, то саме він є характерним для MicroGrid і являє собою найбільший інтерес з точки зору застосування методів машинного навчання, оскільки наявність «людського фактору» веде до великої кількості можливих сценаріїв, які підлягають обробці та подальшому використанню в якості навчальної вибірки. При цьому традиційні методи обробки можуть виявитися нездатними обробити настільки великі обсяги даних, що це дозволяє віднести їх до категорії Big Data.

Серед сукупності методів машинного навчання найбільш придатним є метод детектування аномалій (Anomaly Detection). В даній магістерській роботі розглядається задача застосування Anomaly Detection як один із засобів Machine Learning для визначення нетипових моделей поведінки людини (аномальної поведінки) у MicroGrid.

Застосування технологій обробки великих даних, зокрема методів машинного навчання, за останні роки суттєво прискорило і поліпшило процес обробки даних практично в будь-якій предметній області (фізика, економіка, медицина і т.д.). Однак в галузі керування електротехнічними об‘єктами та інтелектуальними енергетичними мережами, до яких відносяться, зокрема, системи MicroGrid з відновлювальними та альтернативними джерелами енергії, застосування таких методів до теперішнього часу було відсутнім, хоча формування керуючих алгоритмів саме з позицій урахування та обробки великих даних (параметрів клімату, екологічних параметрів, економічних показників) в таких системах має велику перспективу. Адже це дає можливості всебічно оцінювати, аналізувати та прогнозувати режими функціонування, обирати та проводити порівняльний аналіз найкращої стратегії керування, здійснювати неперервну діагностику станів та робочих режимів з метою своєчасного виявлення аномальних, передаварійних та аварійних станів, що є надзвичайно важливим для подальшого розвитку методів енергозбереження, енергоефективності, широкого впровадження та покращення якості функціонування MicroGrid. З іншого боку, врахування великих обсягів даних, необхідність їх подальшої обробки, зберігання і передавання веде до необхідності залучення принципово нових методів та інформаційних технологій – хмарних та розподілених обчислень, штучного інтелекту, машинного навчання та залучення концепції Інтернету речей (Internet of Things) [12] для формування та диспетчеризації єдиного інформаційного середовища.

Для сучасних скадних комплексів MicroGrid актуальною є проблема створення гетерогенної мережі збору, обробки та передавання даних, що характеризують внутрішній стан та параметри оточуючого середовища електротехнічних комплексів з джерелами розосередженої генерації MicroGrid: параметри клімату та екологічні параметри, технічні та технологічні параметри MicroGrid (фізичні та електричні параметри,

показники ефективності режимів функціонування), економічні показники ефективності роботи (оптимізаційна цільова функція, динамічні тарифи на електричну енергію у MicroGrid). Враховуючи великий обсяг цих даних, що обґрунтовує віднесення їх до категорії BigData, можна стверджувати, що питання високоефективної та швидкодіючої обробки цих даних з використанням сучасних методів машинного навчання, розподілених обчислень, штучного інтелекту є надзвичайно актуальними. Такий підхід надає можливість забезпечити функціонування MicroGrid, зокрема, об’єктів військового призначення, критичними вимогами для яких є мінімальна кількість фізичних інформаційних ліній, надійне електропостачання та можливість роботи в автономному режимі.

Сучасні електротехнічні комплекси із джерелами розосередженої генерації MicroGrid характеризуються високим ступенем насиченості різноманітними пристроями генерації, споживання та накопичення енергії, а також передавання великих масивів різнотипних (гетерогенних) даних, що значно ускладнює задачу побудови адекватних методів оперативного керування та інформаційного обміну у інформаційно-керуючому середовищі таких комплексів.

Технології обробки великих даних – методи машинного навчання, класифікації, аномальної детекції, регресійного аналізу – дозволяють оперувати з великими обсягами даних, накопичених в процесі тривалого функціонування MicroGrid в різних режимах, та є основою для вибору ефективної стратегії керування.

Для контролю за діяльністю мешканця в будинку використовується низка сенсорних пристроїв. Інформація, зібрана з пристроїв, обробляється та зберігається для аналізу та використання в поточному та майбутньому стані. Оскільки обсяг цих даних надзвичайно великим, це пояснює доцільність використання методів машинного навчання для обробки Big Data.

MicroGrid з вбудованим модулем інтелектуального машиннгоо навчання реалізує концепцію, в якій мережа датчиків, інтегрована в мережу пристроїв обробки створює великий потік даних".

Сучасні MicroGrid використовують системи моніторингу здоров'я та використання (Health and usage monitoring systems - HUMS).

Деякими типовими прикладами датчиків, наявних у MicroGrid, є датчики температури, руху, відстані, вологості, звуку, потоку води, газу, дверей. Екологічні датчики використовуються для виявлення взаємодії між користувачем та об'єктами, що допомагає визначити показники щоденної діяльності людини. До них відносяться датчики, вбудовані в ліжко, стільці, кухонні прилади тощо. Датчики руху виявляють переміщення людини використовуючи оптичні, мікрохвильові, інфрачервоні, акустичні спостереження. Датчики руху зазвичай використовують у складі систем безпеки та освітлення. Також використовується мережа бездротових датчиків руху, де деякі з них об'єднуються з контактними датчиками на дверях.

Інфрачервоні датчики близькості від Motionwireless використовуються системою In-HomeMonitoring (IMS) [13]. Ці датчики руху можуть використовуватися в Smart Houses для виявлення безпеки та падіння для людей похилого віку, відстеження користувачів та аналізу поведінкових моделей.

В Університеті охорони здоров'я і науки штату Орегон [14], інфрачервоні датчики руху були використані при розробці системи фіксування рухів мешканців. Ці інфрачервоні датчики були поміщені в кожний будинок MicroGrid. Магнітні контактні датчики також були розміщені на дверях, щоб "відстежувати потік відвідувачів" або відстежувати, чи є хтось у домі.

Проте недолік цього полягає в тому, що мешканці повинні були носити ідентифікатори радіочастотної ідентифікації (RFID) для підключення до приймача для ідентифікації. Цей недолік ускладнювався, якщо у будь-який

момент часу у домі було більше одного мешканця, оскільки кожній людині потрібно було позначати RFID, можна виявити, де знаходиться ця особа. Кілька об'єктів можна розпізнавати одночасно, аналізуючи вагу об'єкта.

Ультразвукові датчики також використовуються для виявлення руху. Система Gator MicroGrid [15] використовує ультразвукові датчики для виявлення руху, орієнтації та інформації про місце розташування дорослих людей у MicroGrid. Кілька приймачів дозволяють встановлювати різні відстані для кожного приймача, визначаючи таким чином точне місцезнаходження. Датчики для визначення камери та руху можуть працювати разом у кількох сценаріях. При реалізації розподіленої мережі інтелектуальних камер, функції якої полягали в локалізації та виявленні падінь користувачів. У системі використовуються малопотужні камери з датчиками класу Mote (вузли сенсорів), створюючи інфраструктуру бездротової мережі [16].

Крім того, камери використовують децентралізовану процедуру виявлення падінь. Інший приклад використання камер - це ті, які випадково приймають зображення навколишнього середовища.

Зроблені знімки використовуються для виявлення "контекстної інформації": наприклад, якщо людина спить, це випливає з відсутності руху. Маленькі камери низької роздільної здатності (наприклад, 352-288 пікселів) є переважними в MicroGrid, оскільки їх простіше розгортати, вони вимагають меншої роздільної здатності та меншої потужності, і їх простіше під'єднувати до джерела живлення для обробки. Крім того, камери використовували децентралізовану процедуру виявлення падінь.

Датчики для виявлення лихоманки були розроблені для вимірювання температури на основі аналізу термічних зображень. Детектори лихоманки використовують термічну камеру, яку можна розмістити в будь-якому місці MicroGrid (над ліжком, у ванній кімнаті).

Інший проект використовує датчики, поміщені в кімнату для локалізації людини за допомогою Impulse-Radio Ultra-WideBand (IR-UWB) [17].

Базова станція використовувалася для отримання датчиків даних та координат позначки. Використовуючи оцінку, увімкнуту IR-UWB, для оцінки відстані до позначки, використовуючи алгоритми об'ємного часу (RTT).

Бездротові сенсорні вузли (Sensor Nodes - SN) також використовуються в Smart Houses. SN - це невеликі пристрої з обчислювальними процесорами, блоками бездротового радіо зв'язку та датчиками.

Логічний механізм планування сну на основі кореляції (LCSSM) був впроваджений для зменшення енергоспоживання бездротових SN. В іншому дослідженні був реалізований малопотужний низькочастотний смуговий інтегрований КМОП-фільтр для пасивних інфрачервоних (Passive Infra-red - PIR) датчиків в бездротових SN, що також зменшує споживання електроенергії. PIR також використовувались разом з sensorsflexifore в MicroGrid для виявлення зайнятості людини в об'єктах (диван, туалет, ліжко, стільці тощо). Датчики Flexiforce були реалізовані з динамічним порогом в режимі реального часу, щоб забезпечити більш точне читання вихідного сигналу датчика.

Інші проекти також використовують смартфони, для визначення активності людини за допомогою інтегрованого акселерометра, гіроскопа, GPS та камери.

Акселерометри в наручних годинках використовуються для моніторингу рухової активності. Головна ідея полягає в тому, щоб визначити основні рухи людини (лежати, сидіти, стояти, ходити, бігати, підніматися чи спуститися вниз, працювати на комп'ютері).

Зібраний сигнал надсилається на персональний сервер через радіочастотний зв’язок.

Korel and Koo [18] э розробниками систем про контекстне осмислення з використанням сенсорних мереж тіла (Body Sensor Network - BSN) для постійного моніторингу пацієнтів, щоб виявити аномалії, що загрожують життю. Пацієнт носить або має імплантований пристрій для контролю будь-якого фізіологічного параметру (наприклад, артеріального тиску, пульсу). Дослідження зосереджено на контекстно-осмисленому зондування та порівнянні байєсівських мереж, штучних нейронних мереж та прихованих марківських моделей.

Схожий метод контексного осмислення і у цій роботі для виявлення аномальних поведінкових моделей людини у MicroGrid.

1.3. Активність людини у MicroGrid

Люди, як правило, дотримуються конкретних моделей у своєму повсякденному житті. У контексті MicroGrid щоденна діяльність користувача генерує шаблони, які відіграють важливу роль у прогнозуванні майбутніх подій. Мета інтелектуального інтерфейсу користувача MicroGrid полягає у формуанні шаблонів повсякденної життєвої діяльності; таким чином, MicroGrid повинен знаходити повторювані закономірності в діяльності користувача та передбачити поведінку користувача для отримання додаткової допомоги.

Моніторинг активності користувачів використовується для спостереження та реєстрації дій особи, з метою досягнення цілей комфорту та ефективності, які MicroGrid може запропонувати. Тому необхідно забезпечити здатність до вивчення та застосовування отриманих знань, для адаптації системи керування Microgrid до поведінки користувача. Оскільки користувач створює шаблон, ненормальна поведінка користувачів може бути виявлена шляхом побудови звичайного поведінкового малюнка користувача. Як правило, датчики та фотоапарати в MicroGrid використовуються для

відстеження або ідентифікації діяльності користувача та аналізу поведінки людей.

Спостереження за поведінкою користувача використовується для прогнозування його подальшої активності. Дослідження в напрямку виявлення нетипових (аномальних) поведінкових моделей людини є актуальними для систем психофізіологічного моніторингу літніх людей, осіб, що потребують постійного нагляду та піклування, перебувають у постстресових станах, знаходяться під впливом надзвичайних або екстремальних ситуацій.

Таким чином, розробка методів розпізнавання активності людини у MicroGrid грунтується на алгоритмах інтелектуальному керуванні до яких, зокрема відносяться і алгоритми машинного навчання. Алгоритми штучного інтелекту, алгоритми машинного навчання та методи виведення даних використовуються для моделювання та прогнозування поведінки користувача.

Враховуючи суттєві відмінності поведінкової активності різних людей та навіть тієї чи самоє людини в різні дні, надзвичайно важливим є формуваня тренувальних вибірок, на базі яких здійснюється навчання системи. Повноційність та адекватність тренувальних вибірок є запорукою достовірних результатів.

Інформаційна інфраструктура сучасних комплексів MicroGrid оснащена величезною кількістю датчиків, що фіксують різні типи подій і утворюють інформаційну картину функціонування технічних пристроїв та підсистем, а також переміщення та дії людини. Кількість спрацьовувань n двопозиційних датчиків (що фіксують бінарні події типу «вкл/викл») можна розрахувати

$$N = 2^n \sum_{m=1}^{n} C_n^m + 1, \qquad (1.1)$$

де множник 2 визначає спрацювання у двох режимах.

Кількість сполучень з n елементів по i

$$C_n^m = \frac{n(n-1)...[n-(m-1)]}{m!} . \quad (1.2)$$

Наприклад, при кількості датчиків n = 24 кількість різних інформаційних станів у базі данних може досягати десятків тисяч. При аномальній поведінці людини спрацьовувань датчиків може бути ще більше. Тому задача обробки та аналізу даних з усіх датчиків ускладнюється, а її розв'язок традиційними методами стає неможливим через необхідність оперувати з Big Data. Саме це обумовлює залучення методів машинного навчання для вирішення поставленої задачі.

Висновки до першого розділу

Наявність людського фактору у MicroGrid призводить до необхідності оперування великими обсягами даних, що змушує дослідників звертатися до спеціалізованих методів роботи з великими даними (Big Data).

Серед сукупності методів машинного навчання найбільш придатним є метод детектування аномалій (Anomaly Detection), який було обрано для визначення нетипових моделей поведінки людини (аномальної поведінки) у MicroGrid.

РОЗДІЛ 2. ОГЛЯД ІСНУЮЧИХ АЛГОРИТМІВ ANOMALY DETECTION У MACHINE LEARNING

2.1 Методи Machine Learning

Інформаційно-керуюча система MicroGrid потребує потужного інструментарію інформаційної підтримки для забезпечення збору обробки та прогнозування великих обсягів різнотипних даних.

Важливою частиною таких рішень є модуль аналізу даних, в якому реалізовано агоритм обробки аналізу та встановлення взаємозв'язків між подіями. За допомогою цих моделей даних система MicroGrid має спрогнозувати поведінку користувача на основі історичних даних та розвивати так звану ситуаційну обізнаність, тобто зрозуміти наміри користувача в певний момент і відповідно змінювати параметри робочих режимів пристроїв та підсистем.

На рис 2.1 зображено класифікаційну схему схема методів Machine Learning.

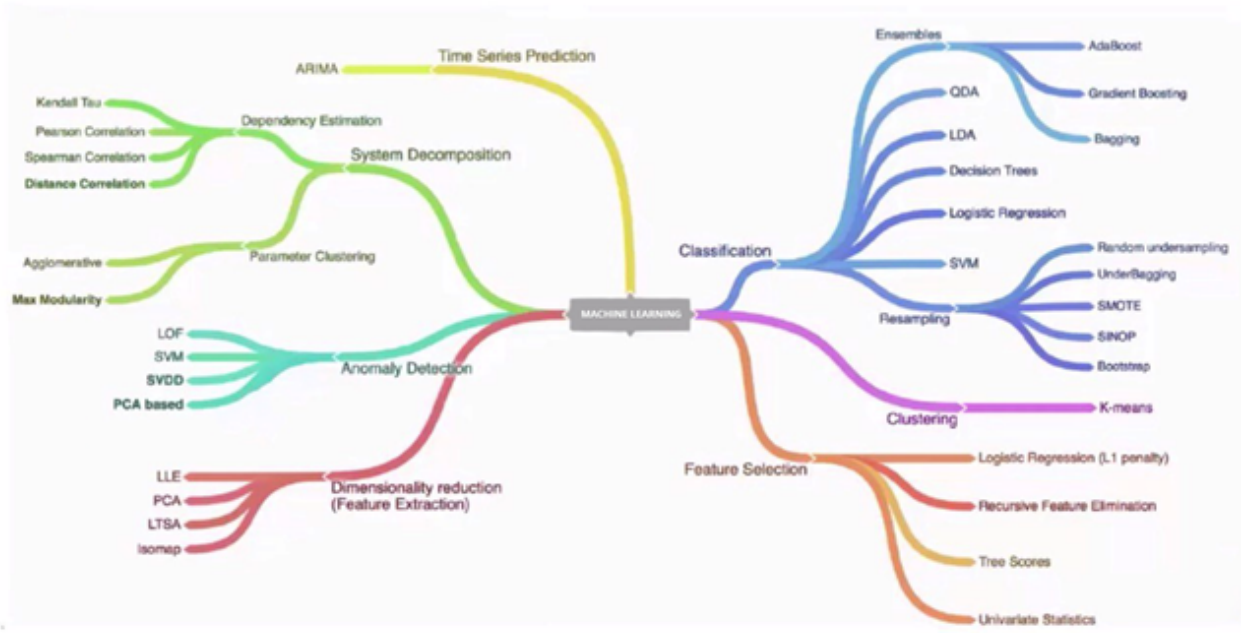

Рис. 2.1 Методи Machine Learning

В даній роботі для формування поставленої мети було обрано гілку Anomaly Detection рис 2.1.

Серед методів Anomaly Detection розрізняють наступні:

1. LOF (Local Outlier Factor);
2. SVM (Support Vector Machine);
3. SVDD (Support Vector Domain Description);
4. PCA (Principal component analysis).

Розглянемо ці методи більш докладно з точки зору їх недоліків та переваг при розв’язанні задачі виявлення аномальної поведінки людини з MicroGrid.

2.2 LOF (Local Outlier Factor)

Локальний рівень викиду є алгоритмом у виявленні аномалій, який запропонували Маркус М. Бройніг, Ганс-Петер Кригель, Реймонд Т. Нг і Сандер Йорг у 2000 році [19] для знаходження аномальних точок даних шляхом вимірювання локального відхилення даної точки даних з урахуванням її сусідів.

Даний метод заснований на оцінці щільності розташування об'єктів, що перевіряються на викиди. Об'єкти, що лежать в областях найбільш низької щільності, вважаються викидами. Перевага методу LOF перед іншими методами, які працюють з високою щільністю об'єктів, полягає в тому, що в LOF розглядається так звана «локальна щільність». Таким чином, LOF успішно розпізнає аномалії в ситуаціях, коли у вибірці присутні об'єкти різних класів, які не є аномаліями.

На рис 2.2 показаний приклад, коли об'єкти навчальної вибірки належать двом класам C_1 і C_2. Об'єкти в двох класах мають різну щільність. Точки o_1 і o_2. є аномаліями. Завдяки обчисленню локальної щільності класів LOF успішно розпізнає обидві аномалії. Методи, засновані на обчисленні

середньої щільності всіх об'єктів, в більшості випадків виявляють викид o_1, але пропускають викид o_2, що підтверджується в [20, 21].

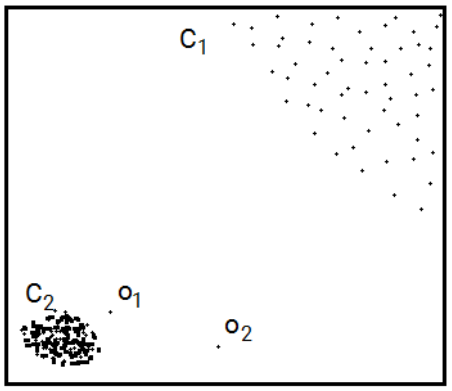

Рис 2.2. Механізм LOF. Випадок областей з різною щільністю

Локальна щільність оцінюється типовою відстанню від точки до сусідніх точок. Визначення «відстані досяжності», що використовується в алгоритмі, є додатковим заходом для отримання більш стійких результатів всередині кластерів.

Формальний опис Local Outlier Factor. Нехай $D_k(y)$ є відстанню від точки y до k найближчого сусіда. Досяжністю (reachability distance) точки x щодо точки y називається величина

$$R_k(x,y) = \max(\rho(x,y), D_k(y)). \qquad (2.1)$$

Нехай $AR_k(x)$ - середня досяжність точки x щодо k своїх найближчих сусідів, $N_k(x)$ - множина k найближчих сусідів x. Тоді:

$$LOF_k(x) = \underset{y \in N_k(x)}{mean} \frac{AR_k(x)}{AR_k(y)} . \qquad (2.2)$$

Сенс (2.2) полягає в тому, щоб порівняти середню досяжність точки та її найближчих сусідів. Для “нормальних” даних вірно не тільки, що оцінка (2.1) локальної щільності мала, а й те, що вона незначно відрізняється від такої ж оцінки для найближчих сусідів. приклад роботи алгоритму наведено на рис. 2.2.

Переваги Local Outlier Factor. Внаслідок локальності підходу алгоритм LOF здатний виявити в наборі даних викиди, які могли б не бути викидами в інших областях набору даних. Наприклад, точка на «малій» відстані до будь-якого щільного кластера є викидом, в той час як точка всередині рідкісного кластера може мати схожі відстані з її сусідами.

У той час як геометричний сенс алгоритму може бути застосований тільки до векторних просторів низької розмірності, алгоритм може бути застосований в будь-якому контексті, де функція несхожості може бути визначена.

Недоліки Local Outlier Factor. Для коефіцієнтів локальної щільності та досяжності не існує встановлених заданих нем допустимих значень, тому, інакше кажучи, немає ніякого чіткого правила, за яким точка буде аномалією. В одному наборі даних значення 1,1 може означати викид, в іншому наборі даних і параметризації (з сильними локальними флуктуаціями) значення 2 може ще означати, що точка входить в зону нормальності. Таким чином, LOF має підвищену чутливість до специфіки досліджуваних наборів даних.

2.3 SVM (Support Vector Machine)

Завдання класифікації полягає у визначенні, до якого як мінімум двох класів спочатку відомих належить цей об'єкт. Зазвичай таким об'єктом є вектор в n-вимірному матеріальному просторі. Координати вектора описують окремі атрибути об'єкта. Наприклад, колір «c», заданий в моделі RGB, є вектором в тривимірному просторі: c = (red, green, blue).

Якщо класів всього два (ON/OFF), то завдання називається бінарною класифікацією. Якщо класів кілька – це випадок мультикласової класифікації. Також можуть бути зразки кожного класу - об'єкти, про які заздалегідь відомо, до якого класу вони належать. Такі завдання називають

навчанням з учителем, а відомі дані називаються навчальною вибіркою. Якщо класи з самого початку не задані, то це завдання кластеризації.

Отже, математичне формулювання задачі класифікації наступне: нехай X - простір об'єктів (наприклад, $\sim^n$ - n-вимірний простір зі значень з плаваючою комою), Y - класи (наприклад, $Y = \{-1;1\}$). Дана навчальна вибірка: $(x_1, y_1),...,(x_m, y_m)$. Потрібно побудувати функцію $F : X \longrightarrow Y$ (класифікатор), яка пов'язує клас Y довільного об'єкту X.

Метод Support Vector Machine спочатку відноситься до бінарних класифікаторів, хоча існують способи змусити його працювати і для задач мультикласифікації.

Ідею методу зручно проілюструвати на наступному простому прикладі: нехай є точки на площині, розбиті на два класи (рис. 2.3) і проводиться лінія, що розділяє ці два класи (червона лінія).

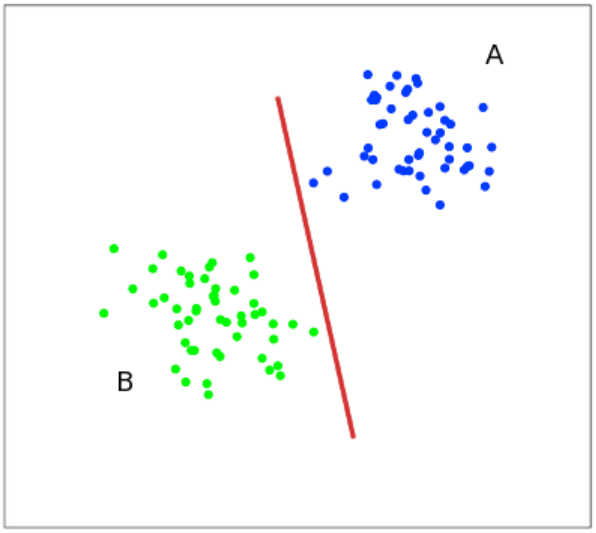

Рис 2.3. SVM - розбиття на класи

Далі всі нові точки (не з навчальної вибірки) автоматично класифікуються наступним чином:

- точка вище прямої потрапляє в клас A,
- точка нижче прямої - в клас B.

Така пряма називається роздільною прямою. Однак, в просторах високих розмірностей пряма вже не буде розділяти класи, так як поняття «нижче прямої» або «вище прямої» втрачає сенс. Тому замість прямих необхідно розглядати гіперплощини - простори, розмірність яких на одиницю менше, ніж розмірність початкового простору. в $\sim^3$, наприклад, гіперплощина - це звичайна двовимірна площина. У прикладі вище існує кілька прямих, які поділяють два класи (рис. 2.4).

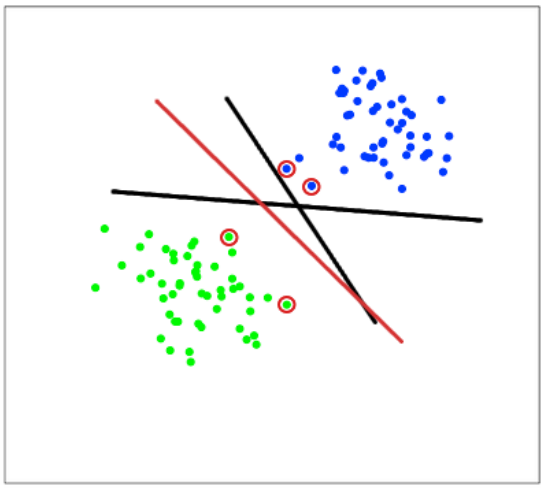

Рис 2.4. SVM - прямі, що розділяють класи

З точки зору точності класифікації найкраще вибрати пряму, відстань від якої до кожного класу є максимальною. Іншими словами, обирається та пряма, яка розділяє класи найкращим чином (червона пряма на рис. 2.4). Така пряма, а в загальному випадку - гіперплощина, називається оптимальною розділяючою гиперплощиною.

Вектори, що лежать ближче всіх до розділяючої гиперплощини, називаються опорними векторами (support vectors). На рис. 2.4 вони позначені червоним.

Нехай є навчальна вибірка: $(x_1, y_1),...,(x_m, y_m), \quad x_i \in \sim^n, y_i \in \{-1,1\}$.

Метод опорних векторів будує классифіковану функцію F у вигляді $F(x) = sign(\langle w, x \rangle + b)$, $\langle w, x \rangle$ - скалярний вираз де w - нормальний вектор до розділяючої гіперплощини, b - допоміжний параметр. Ті об'єкти, для яких

$F(x)=1$, потрапляють в один клас, а об'єкти з $F(x)=-1$ - в інший. Вибір саме такої функції не випадковий: будь-яка гіперплощина може бути задана у вигляді $\langle w,x\rangle+b=0$ для деяких w і b. Далі необхідно вибрати такі w і b, які максимізують відстань до кожного класу. Можна підрахувати, що дана відстань дорівнює $\frac{1}{\|w\|}$ (рис.2.5).

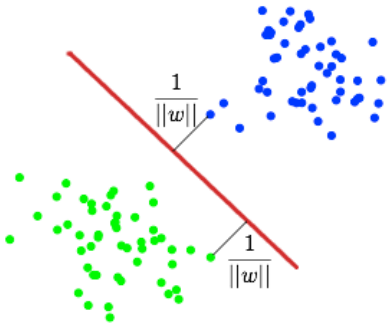

Рис. 2.5. Знаходження максимума $\frac{1}{\|w\|}$ до розділяючої прямої

Проблема знаходження максимуму $\frac{1}{\|w\|}$ еквівалентна проблемі знаходження мінімуму $\|w\|^2$. Необхідно записати все це у вигляді завдання оптимізації:

$$\begin{cases} \arg\min\limits_{w,b}\|w\|^2, \\ y_i\left(\langle w,x_i\rangle+b\right)\geq 1 \end{cases} \quad i=1,\ldots,m. \qquad (2.3)$$

яка є стандартною задачею квадратичного програмування і вирішується за допомогою множників Лагранжа [23].

Поняття оптимальної розділяючої гіперплощини. Припустимо, що вибірка є лінійно роздільною, тобто існують такі значення параметрів w, w_0, при яких функціонал числа помилок приймає нульове значення.

$$Q(w,w_0)=\sum_{i=1}^{\ell}\left[y_i\left(\langle w,x_i\rangle-w_0\right)<0\right] \qquad (2.4)$$

Але тоді розділяюча гіперплощина не єдина, оскільки існують і інші положення розділяючої гиперплощини, що реалізують те ж саме розбиття вибірки. Ідея методу полягає в тому, щоб розумним чином розпорядитися цією свободою вибору. Наприклад, розділяюча гіперплощина розташована максимально далеко від найближчих до неї точок обох класів. Спочатку даний принцип класифікації виник з евристичних міркувань: цілком природно вважати, що максимізація *зазору* (margin) між класами повинна сприяти більш достовірній класифікації.

Слід зауважити, що параметри лінійного порогового класифікатора визначені з точністю до нормування: алгоритм $a(x)$ не зміниться, якщо w та w_0 одночасно помножити на одну і ту ж позитивну константу. Зручно вибрати цю константу таким чином, щоб для всіх межуючих (тобто найближчих до розділяючої гіперплощини) об'єктів x_i з X^ℓ виконувалася умова:

$$\langle w, x_i \rangle - w_0 = y_i . \tag{2.5}$$

Зробити це можливо, оскільки при оптимальному положенні розділяючої гіперплощини всі межуючі об'єкти знаходяться від неї на однаковій відстані. Решта об'єктів знаходяться далі. Таким чином, для всіх $x_i \in X^\ell$

$$\langle w, x_i \rangle - w_0 \begin{cases} \le -1, & \textit{якщо } y_i = -1; \\ \ge 1, & \textit{якщо } y_i = +1; \end{cases} \tag{2.6}$$

Умова $-1 < \langle w, x_i \rangle - w_0 < 1$ задає смугу, що розділяє класи. Жодна з точок навчальної вибірки не може лежати всередині цієї смуги. Межами смуги служать дві паралельні гіперплощини з напрямляючим вектором w. Точки, найближчі до розділяючої гіперплощини, лежать в точності на межах смуги. При цьому сама розділяюча гіперплощина проходить рівно по середині смуги.

Ширина розділяючої смуги. Щоб розділяюча гіперплощина якнайдалі відстояла від точок вибірки, ширина смуги повинна бути максимальною. Нехай x_- і x_+ - дві довільні точки класів -1 і $+1$ відповідно, що лежать на межі смуги. Тоді ширина смуги є:)

$$\left\langle (x_+ - x_-), \frac{w}{\|w\|} \right\rangle = \frac{\langle w, x_+ \rangle - \langle w, x_- \rangle}{\|w\|} = \frac{\langle w_0 + 1 \rangle - \langle w_0 - 1 \rangle}{\|w\|} = \frac{2}{\|w\|} \qquad (2.7)$$

Ширина смуги максимальна, коли норма вектора w мінімальна. Отже, в разі, коли вибірка лінійно роздільна, досить прості геометричні міркування приводять до наступної задачі: потрібно знайти такі значення параметрів w і w_0, при яких норма вектора w мінімальна за умови (2.6). Це задача квадратичного програмування. Вона буде детально розглянута в наступному розділі. Потім буде зроблено узагальнення на той випадок, коли лінійної роздільності немає.

Лінійно роздільна вибірка. Побудова оптимальної розділяючої гіперплощини зводиться до мінімізації квадратичної форми при обмеженнях-нерівностей виду див (2.6) щодо $n+1$ змінних w, w_0 :

$$\begin{cases} \langle w, w \rangle \longrightarrow \min \\ y_i \left(\langle w, x_i \rangle - w_0 \right) \geq 1 \end{cases} i = 1, \ldots, \ell \qquad (2.8)$$

За теоремою Куна-Таккера ця задача еквівалентна двоїстій задачі пошуку сідлової точки функції Лагранжа [24]:

$$\begin{cases} \zeta(w, w_0 \lambda) = \frac{1}{2} \langle w, w \rangle - \sum_{i=1}^{\ell} \lambda_i \left(y_i \left(\langle w, x_i \rangle - w_0 \right) - 1 \right) \longrightarrow \min_{w, w_0} \max_{\lambda}; \\ \lambda_i \geq 0, i = 1, \ldots, \ell; \\ \lambda_i = 0, або \langle w, x_i \rangle - w_0 = y_i, i = 1, \ldots, \ell; \end{cases} \qquad (2.9)$$

де $\lambda = (\lambda_1, \ldots, \lambda_\ell)$- вектор двоїстих змінних. Остання з трьох умов називається умовою доповнюючої нежорсткості.

Необхідною умовою сідлової точки є рівність нулю похідних Лагранджа. Звідси випливають два корисних співвідношення:

$$\frac{\partial \zeta}{\partial w} = w - \sum_{i=1}^{\ell} \lambda_i y_i x_i = 0 \longrightarrow w = \sum_{i=1}^{\ell} \lambda_i y_i x_i; \qquad (2.10)$$

$$\frac{\partial \zeta}{\partial w_0} = -\sum_{i=1}^{\ell} \lambda_i y_i = 0 \longrightarrow \sum_{i=1}^{\ell} \lambda_i y_i = 0; \qquad (2.11)$$

З (2.10) випливає, що шуканий вектор вагів w є лінійною комбінацією векторів навчальної вибірки, причому тільки тих, для яких $\lambda_i \neq 0$. Згідно з умовою, що доповнює нежорсткость на цих векторах x_i обмеження-нерівності звертаються в рівності:

$$\langle w, x_i \rangle - w_0 = y_i,$$

отже, ці вектори знаходяться на межі розділяючої смуги. Всі інші вектори відстоять далі від межі, для них $\lambda_i = 0$, і вони не беруть участі в сумі (2.10). Алгоритм (2.4) не змінився б, якби цих векторів взагалі не було в навчальній вибірці.

Якщо $\lambda_i > 0$ та $\langle w, x_i \rangle - w_0 = y_i$, то об'єкт навчальної вибірки x_i називається опорним вектором (support vector).

Після математичних перетворень формується еквівалентна задача квадратичного програмування, що містить тільки двоїсті змінні:

$$\begin{cases} -\zeta(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \left(\langle x_i, x_j \rangle \right) \longrightarrow \min_{\lambda}; \\ \lambda_i \geq 0, i = 1, \ldots, \ell; \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0; \end{cases} \qquad (2.12)$$

Тут мінімізується квадратичний функціонал, що має невід'ємно певну квадратичну форму, тобто описує опуклу форму. Область, яка визначається обмеженнями нерівностями і одною рівністю, також є опуклою.

Отже, дана задача має єдиний розв'язок. Припустимо, що цей розв'язок знайдено. Тоді вектор w обчислюється за формулою (2.10). Для визначення порогу w_0 досить взяти довільний опорний вектор x_i і виразити w_0 з рівності

$w_0 = \langle w, x_i \rangle - y_i$. На практиці для підвищення чисельної стійкості рекомендується брати в якості w_0 середнє по всім опорним векторах, а ще краще - медіану, яка обчислюється за формулою:

$$w_0 = med\left\{\langle w, x_i \rangle - y_i : \lambda_i > 0, i = 1,..., \ell\right\}. \quad (2.13)$$

В результаті алгоритм класифікації може бути записаний в наступному вигляді:

$$a(x) = sign\left(\sum_{i=1}^{\ell} \lambda_i y_i \langle x_i, x \rangle - w_0\right) \quad (2.14)$$

Звернемо увагу, що реально підсумовування йде не по всій вибірці, а тільки по опорним векторах, для яких $\lambda_i \neq 0$. Саме це властивість розрідженості (sparsity) позитивно відрізняє SVM від інших лінійних роздільників - дискримінанту Фішера, логістичної регресії і одношарового персептрона [25], оскільки дозволяє оперувати зі значно меншим набором даним, отже, спростити реалізацію та підвищити швидкодію алгоритму.

Лінійна неподільність. На практиці випадки, коли дані можна розділити гіперплощиною або лінійно, досить рідкісні. Приклад лінійної нероздільності можна бачити на рис. 2.6.

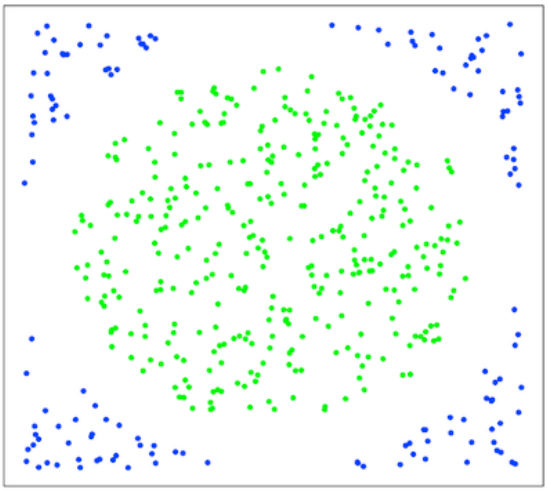

Рис. 2.6. Приклад лінійної нероздільності даних

В цьому випадку всі елементи навчальної вибірки вкладаються в простір X більш високої розмірності за допомогою спеціального відображення $\varphi : \tilde{}^{n} \longrightarrow X$. При цьому відображення φ обирається так, щоб в новому просторі X вибірка була *лінійно* роздільною.

Класифікована функція F приймає вигляд $F(x) = sign(\langle w, \varphi(x) \rangle + b)$. Вираз $k(x, x') = \langle \varphi(x), \varphi(x') \rangle$ називається *ядром* класифікатора. З математичної точки зору ядром може служити будь-яка позитивно певна симетрична функція двох змінних. Позитивна визначеність необхідна для того, щоб відповідна функція Лагранжа в задачі оптимізації була обмежена знизу, тобто задача оптимізації була б коректно визначена. Точність класифікатора залежить, зокрема, від вибору ядра.

Існує кілька «стандартних» ядер, які при найближчому розгляді призводять до вже відомих алгоритмів: поліноміальних розділяючих поверхонь, двошарових нейронних мереж, потенційних функцій (RBF-мереж), та інших.

Таким чином, ядра претендують на роль універсальної мови для опису широкого класу алгоритмів навчання по прецедентах. Спостерігається парадоксальна ситуація. З одного боку, ядра - одне з найкрасивіших і плідних винаходів в машинному навчанні. З іншого боку, до цих пір не знайдено ефективного загального підходу до їх підбору в конкретних завданнях.

2.4 SVDD (Support Vector Domain Description)

Межа сферичної форми навколо набору об'єктів базується на наборі опорних векторів. Вона має можливість перетворити дані в новому просторі об'єктів без набагато більшої обчислювальної вартості. Використовуючи перетворені дані, SVDD може отримати більш зрозумілі та точніші описи даних. Помилку першого роду, тобто частку навчальних об'єктів, які будуть

відхилені, можна оцінити відразу без використання незалежного тестового набору, що робить дані цього методу важливими.

З набору даних, що містить N об'єктів даних $\{x_i, i = 1,...,N\}$, необхідно вивести сферу з мінімальним обсягом, що містить всі (або більшість) об'єктів даних. Це дуже чутливо до найбільш віддаленого об'єкта в цільовому наборі даних. Коли один або декілька дуже віддалених об'єктів знаходяться в тренувальному наборі, виходить дуже велика сфера, яка не відображатиме дані дуже добре. Тому дозволяються деякі точки даних поза сферою і вводяться так звані «слабкі змінні» ξ_i. Зі сфер, описаних центром *a* і радіусом *R*, мінімізується радіус

$$F(R,a,\xi_i) = R^2 + C\sum_i \xi_i, \tag{2.15}$$

де змінна C дає компроміс між простотою (або обсягом сфери) та кількістю помилок (кількість цільових відхилених об'єктів).

Це потрібно звести до мінімуму за обмеженнями:

$$(x_i - a)^T (x_i - a) \le R^2 + \xi_i \quad \forall_i, \xi_i \ge 0. \tag{2.16}$$

Враховуючи ці обмеження в (2.15), побудуємо рівняння методу Лагранжа [23]:

$$L(R,a,\alpha_i,\xi_i) = R^2 + C\sum_i \xi_i - \sum_i \alpha_i \left\{R^2 + \xi_i - (x_i^2 - 2ax_i + a^2)\right\} - \sum_i \gamma_i \xi_i, \tag{2.17}$$

з множниками Лагранжа $\alpha_i \ge 0$ і $\gamma_i \ge 0$. Нові обмеження визначаються рівністю нулю часткових похідних:

$$\sum_i \alpha_i = 1, \;\; a = \frac{\sum_i \alpha_i x_i}{\sum_i \alpha_i} = \sum_i \alpha_i x_i,$$
$$C - \alpha_i - \gamma_i = 0 \quad \forall_i. \tag{2.18}$$

Оскільки $\alpha_i \ge 0$ та $\gamma_i \ge 0$, можна видалити змінні γ_i з третього рівняння у (2.18) і використовувати обмеження $0 \le \alpha_i \le C. \;\; \forall_i$.

З рівнянь (2.17) і (2.18) максимізується щодо α_i:

$$L = \sum_i \alpha_i (x_i \cdot x_i) - \sum_{ij} \alpha_i \alpha_j (x_i \cdot x_j) \tag{2.19}$$

з обмеженнями $0 \le \alpha_i \le C$, $\sum_i \alpha_i = 1$.

З другого рівняння у (2.18) слідує, що центр сфери є лінійною комбінацією об'єктів даних з ваговими коефіцієнтами α_i, які отримуються шляхом оптимізації рівняння (2.19). Тільки для невеликого набору об'єктів рівність у рівнянні (2.16) відповідає об'єктам, які знаходяться на межі самої сфери. Для цих об'єктів коефіцієнти α_i будуть ненульовими і називаються опорними об'єктами. У описі сфери потрібні лише ці об'єкти. Радіус R сфери можна отримати, розраховуючи відстань від центру сфери до опорного вектора з вагою менше, ніж C. Об'єкти, для яких $\alpha_i = C$, потрапляють у верхню межу в (2.18) і виходять за межі сфери. Ці вектори підтримки вважаються перевершеними.

Щоб визначити, чи є випробувальна точка z всередині сфери, слід розрахувати відстань до центру сфери. Тестовий об'єкт z приймається, коли ця відстань менша, ніж радіус, тобто, коли $(z-a)^T(z-a) \le R^2$. Тестовий об’єкт вважається таким, що належить до зони нормальності, якщо для нього виконується умова:

$$(z \cdot z) - 2\sum_i \alpha_i (z \cdot x_i) + \sum_{ij} \alpha_i \alpha_j (x_i \cdot x_j) \le R^2. \tag{2.20}$$

Узагальнення на інші ядра. Представлений спосіб лише обчислює сферу навколо даних у вхідному просторі. Зазвичай дані не розподіляються сферично, навіть якщо найближчі об'єкти ігноруються. Отже, у загальному випадку неможливо розраховувати на отримання дуже жорсткого опису. Оскільки завдання повністю виражається з точки зору добутків опорних векторів (рівняння (2.19) та (2.20)), метод можна зробити більш гнучким. Добуток $(x_i \cdot x_j)$ можна замінити функцією ядра $K(x_i \cdot x_j)$, коли це ядро $K(x_i \cdot x_j)$ задовольняє умовам теореми Мерсера [26]. Це неявно відображає

об'єкти x_i в деякий простір функцій і коли вибирається відповідний простір ознак, можна отримати кращий, більш щільний опис. Не потрібно явне відображення, проблема виражається повністю з точки зору $K(x_i \cdot x_j)$.

Тому всі добутки $(x_i \cdot x_j)$ заміняються на відповідні значення $K(x_i \cdot x_j)$ і рівняння (2.19) набуває вигляду:

$$L = \sum_i \alpha_i K(x_i \cdot x_i) - \sum_{ij} \alpha_i \alpha_j K(x_i \cdot x_j) \qquad (2.21)$$

з обмеженнями $0 \le \alpha_i \le C$, $\sum_i \alpha_i = 1$. Тестовий об'єкт z приймається, коли (див. (2.20):

$$K(z,z) - 2\sum_i \alpha_i K(z,x_i) + \sum_{ij} \alpha_i \alpha_j K(x_i,x_j) \le R^2 \qquad (2.22)$$

Різні функції ядра K приводять до різних меж опису в початковому вхідному просторі. Проблема полягає в пошуку відповідної функції ядра $K(x_i \cdot x_j)$. Розглядаються два варіанти: поліноміальне ядро і гауссове ядро.

Перший варіант для ядра $K(x_i \cdot x_j)$ - це розширений добуток: $K(x_i,x_j) = (x_i \cdot x_j + 1)^d$, де вільним параметром d є ступінь поліноміального ядра. Як стверджує Вапнік В. (1995) [22], це ядро переносить об'єкти у простір високої розмірності, додаючи добутки оригінальних функцій до ступеня d. Наприклад, 2D-вектор (x_1, x_2) пов'язаний з $(x_1, x_2, x_1x_2, x_1^2, x_2^2)$, коли використовується поліноміальне ядро з $d = 2$. Це ядро не призводить до задовільного опису набору даних. Для вищих ступенів d збільшуються всі інші добутки для об'єктів, найбільш віддалених від початку системи координат. Це показано на рис.2.7 за допомогою двовимірного набору даних, що містить 10 об'єктів. Для різних значень ступеня $(d = 1, 5, 25)$ обчислюється сферичний опис.

Відстань до центру сфери побудована на початковому вхідному просторі. Штрихова біла лінія, що перетинає опорні вектори (позначена невеликими колами), є межами опису (див. рис. 2.7).

Чим темніший колір, тим менше відстань. Об'єкти у верхній частині найбільш віддалені від початку, і хоча ці об'єкти не є самими віддаленими об'єктами даних у двох вимірах, вони стають опорними векторами, коли використовуються вищі ступені.

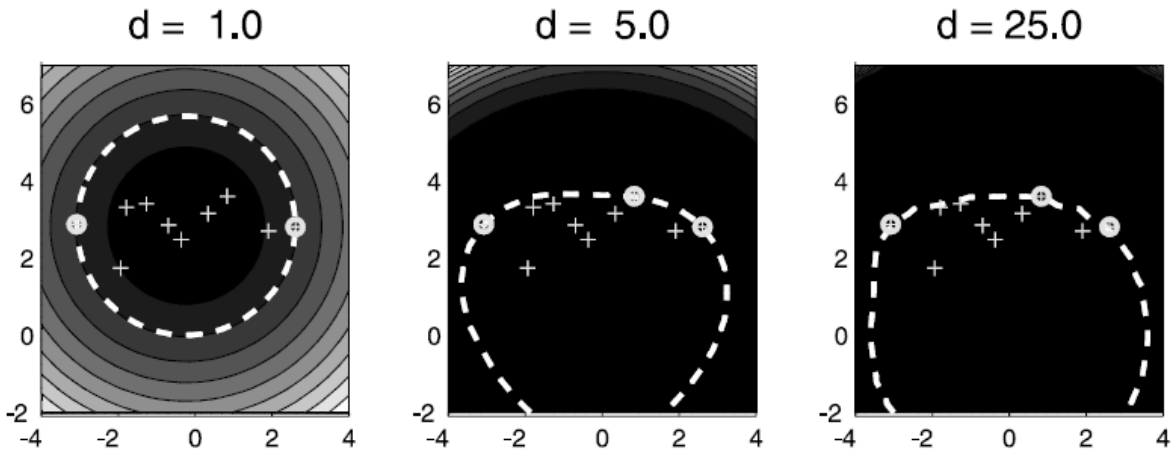

Рис 2.7. Сфери при різних значеннях ступеня

Більша частина простору вводу стає прийнятною. Це описує дуже велику і розріджену сферу в оригінальному двовимірному вхідному просторі. Для зменшення зростаючих відстаней для більших просторових ознак, гауссовське ядро $K_G(x_i, x_j) = \exp(-(x_i - x_j)^2 / s^2)$ є більш доцільним. Рівняння (2.21) набуває вигляду:

$$L = 1 - \sum_i \alpha_i^2 - \sum_{i \neq j} \alpha_i \alpha_j K_G(x_i, x_j) \qquad (2.23)$$

а рівняння (2.22) -

$$-2 \sum_i \alpha_i K_G(z, x_i) \leq R^2 - C_X - 1 \qquad (2.24)$$

де C_X залежить тільки від опорних векторів та α_i, а не від тестового об'єкта z.

На рис 2.8 знову відображається 2D штучний набір даних, що містить 10 об'єктів. Тепер використовується опис опорного вектора з гауссовим ядром для різних значень *S*. Параметр ширини *S* коливається від дуже дрібних (*S* = 1.0 у лівій частині фігури) до великих (*S* = 25.0 у правій частині

фігури). Кількість опорних векторів зменшується, і опис стає більше сферичним.

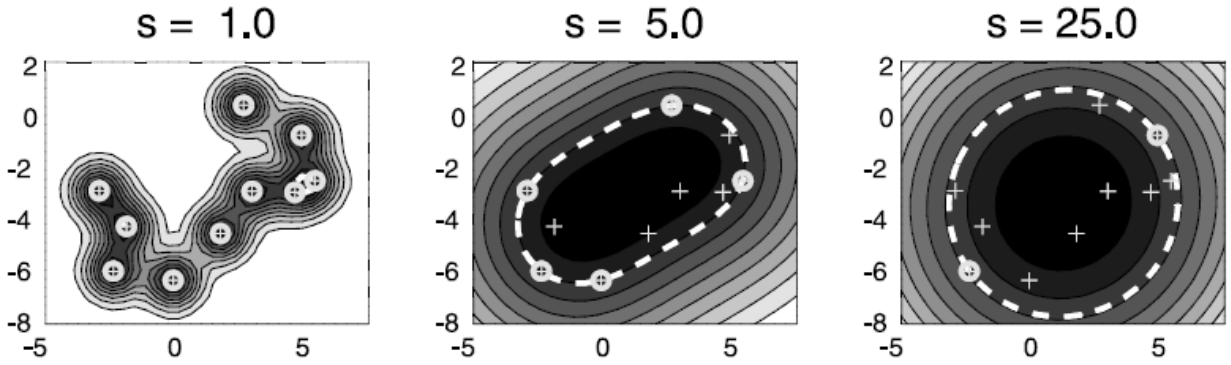

Рис 2.8. Сфери при різних значеннях ширини

Можна одержувати явні рішення для рівняння (2.21) для двох різних екстремальних ситуацій - одне для дуже малих значень і одне для дуже великих значень S (див. рис. 2.8). Для дуже малих S, $K_G(x_i, x_j) \simeq 0, \quad i \neq j$ та $L = 1 - \sum_i \alpha_i^2$. Це максимізовано, коли $\alpha_i = 1/N$ і L становить $1 - 1/N$. Це аналогічно оцінці щільності Парцена [31], де кожен об'єкт підтримує ядро (див. (2.24)) [29]. Всі відстані до центру сфери стають рівними $1 - 1/N$.

Для дуже великих S, $K_G(x_i, x_j) = 1,$ і $L = 1 - \sum_i \alpha_i^2 - \sum_{i \neq j} \alpha_i \alpha_j$. Це максимізовано, коли всі $\alpha_i = 0$ крім одного $\alpha_j = 1,$ а всі відстані до сфери центру дорівнюють нулю. Ця практично велика сфера не буде отримана на практиці, і вона не буде достатньо великою, щоб дати рівне $K_G(x_i, x_j)$ для всіх пар i, j. У правій частині підграфіку рис. 2.8 побудована реалістична гранична ситуація. Опис даних знову є найменшою сферою, яка охоплює повний набір даних, без викидів. Розширення Тейлора рівняння (2.23) показує, що коли вищі замовлення ігноруються, то отримується рівняння (2.19) (вище до масштабного та залежного множника).

У випадку середніх значень S (середній підграфік на рис. 2.8) лише частина об'єктів стає опорними об'єктами. Вираз (2.24) показує, що в цьому випадку отримана відредагована та зважена оцінка щільності Парцена [31]. Параметр C дає верхню межу для параметрів α_i, таким чином, обмежує опорні вектори (2.24). Коли об'єкт x_1 отримує $\alpha_i = C$, опис більше не буде адаптовано до цього об'єкта, і він залишиться поза сферою. Через обмеження $\sum_i \alpha_i = 1$ і $\alpha_i \geq 0$ - лише ті варіанти, для яких C може мати будь-яке входження до рішення рівняння. (2.23), коли $1/N \leq C \leq 1$. Для $C \leq 1/N$ неможливо знайти рішення, тому тоді обмеження $\sum_i \alpha_i = 1$ ніколи не може бути виконано, тоді як для $C > 1$ завжди можна знайти вирішення (α_i завжди менше або дорівнює 1). Коли C обмежується малими значеннями, перевага виходу за межі сфери не є дуже великою, а більша частина об'єктів може бути поза сферою. На практиці перевага C не дуже критична.

Узагальнення. Щоб отримати вказівки на узагальнення або переоцінку характеристик SVDD, необхідно отримати інформацію про (2.15) кількість цільових шаблонів, які будуть відхилені (помилки першого випадку) за цим описом і (2.16) числа віддалених моделей, які будуть прийняті (помилки другого випадку).

Можливо оцінити похибку першого випадку, застосувавши метод leave-one-out на тренувальному наборі, що містить цільовий клас [22]. Коли із навчального набору вилучається об'єкт, який не є опорним об'єктом, межі сфери залишаться незмінними. Якщо вилучається опорний вектор, сфера може стати меншою, оскільки цей опорний вектор знаходиться на межі сфери. Цей об'єкт, що залишився, буде відхилений, тоді як решту навчальних об'єктів буде прийнято (оскільки цей метод навчений цими даними). Таким чином, помилка може бути оцінена як:

$$E\left[P(error)\right] = \frac{\#SV}{N} \qquad (2.25)$$

де $\#SV$ - це кількість опорних векторів.

Коли використовується ядро Гауса, можливо регулювати кількість опорних векторів, змінюючи параметр ширини S. Тому також можливо встановити помилку в виді першого випадку. Коли кількість опорних векторів надто велика, доведеться збільшувати S, а коли число занадто низьке, доведеться зменшити S. Щоб перевірити, наскільки добра оцінка рівняння (2.25), на рис.2.9 побудована оцінка помилок першого випадку як функції параметра ширини S. Метод застосовано до двовимірного набору даних, що містить 10 об'єктів. Також відображається помилка, оцінена за допомогою незалежного тестового набору з 100 об'єктів. Можна зробити висновок, що ця оцінка добре працює. Тому, коли потрібен опис набору даних, можливо заздалегідь встановити оцінку очікуваної швидкості відхилення цільових даних. Лагранжа з рівняння. (2.23), і очікувана помилка для цього рішення отримається за допомогою рівняння. (2.25). Коли ця помилка занадто велика, параметри ширини S збільшуються, або коли ця помилка може все ще збільшуватися, параметри ширини S зменшуються. Це гарантує, що параметр ширини у SVDD адаптований для вирішення проблеми, з огляду на помилку.

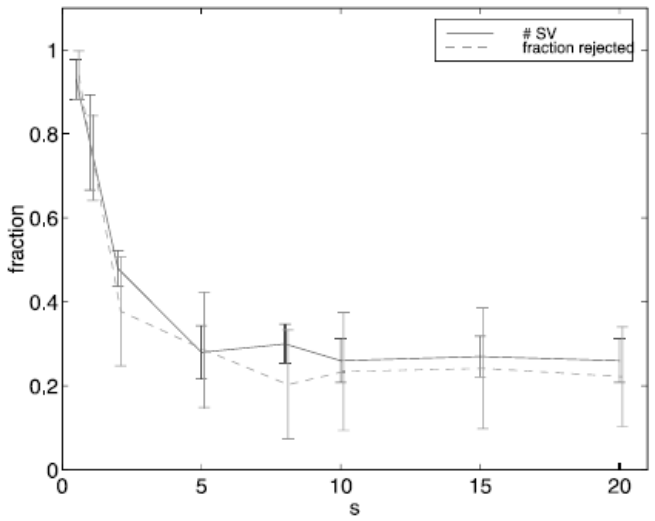

Рис. 2.9. Порівняння частки об'єктів та оцінка помилок

Випадок, коли віддалені об’єкти будуть прийняті за описом сфери, визначає помилку другого роду, яку не можна оцінити за допомогою цієї міри. Загалом, доступний лише добре описаний цільовий класу у формі тренувального набору. Всі інші моделі вважаються недосяжними. Щоб отримати оцінку помилки другого роду, дані навколо оригінального набору слід штучно створити та протестувати. Цей метод вимагає способу отримання або створення даних навколо тренувальних даних, але не в навчальному наборі. Кількість тестових шаблонів повинна бути досить великою для адекватної оцінки, що може стати проблемою в просторах високої розмірності. У задачі класифікації один клас приймається як тестовий, а всі інші класи використовуються як навчальні. Таким чином можуть бути побудовані штучні викиди. Це означає, що вводиться похибка продуктивності. Задачі класифікації часто містять перекриваючі класи і, використовуючи таким чином класичні задачі, продуктивність методів виходу буде нижчою, ніж у звичайних методах класифікації для завдання класифікації. Тим не менш, це дає можливість порівняти різні методи.

2.5 Метод головних компонент PCA

Метод головних компонент (Principal component analysis) використовується для перетворення даних в стек з вхідного багатовимірного атрибутивного простору в новий багатовимірнийц атрибутивний простір, осі якого повернуті по відношенню до осей вихідного простору. Осі (атрибути) в новому просторі некорельовані.

Головна мета перетворення даних при виконанні аналізу за методом головних компонент - стиснення даних шляхом виключення існуючої в них надмірності за рахунок пошуку взаємопов’язаних компонент, виділенню головних компонент та оцінки дисперсії кожної з них.

Перша головна компонента буде характеризуватися найбільшою дисперсією, друга буде відповідати другому за величиною значенню дисперсії і так далі. У більшості випадків перші три або чотири компоненти, отримані в результаті Principal component analysis, будуть описувати більше 95% дисперсії. Решта компонент можуть бути відкинуті. Оскільки новий набір містить меншу кількість компонент, і більше 95% дисперсії вихідного набору залишилося незмінним, обчислення будуть виконуватися швидше, а їх точність при цьому зберігається.

Для інструменту PCA потрібно, щоб були визначені вхідні канали, число головних компонент, в які будуть трансформуватися дані, ім'я вихідного файлу статистики і ім'я вихідного набору.

Основні принципи аналізу за методом головних компонент. При використанні двоканального набору переміщення і обертання осей і трансформація даних здійснюється наступним чином:

- дані наводяться на діаграмі розсіювання.
- для зв'язку точок на діаграмі розсіювання обчислюється еліпс (рис. 2.10).

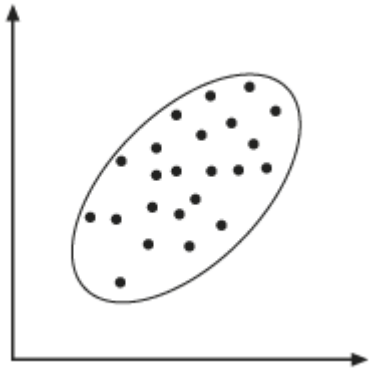

Рис 2.10. Побудована межа еліпса

Визначається головна вісь еліпса (рис. 2.11). Головна вісь стане новою віссю x - першою головнною компонентою (PC1). PC1 має найбільшу дисперсію, тому що це найбільший розріз, який можна зробити через еліпс.

Напрямком PC1 є власний вектор, а його величиною - власне значення. Кут осі x до PC1 - це кут повороту, який використовується в трансформації.

Далі обчислюється перпендикуляр до PC1. Ця лінія є другою головною компонентою (PC2) і новою віссю для вихідної осі y (рис. 2.12).

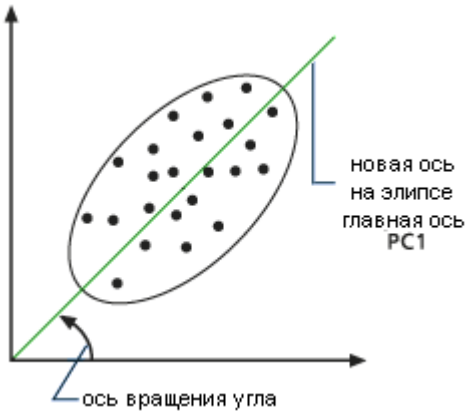

Рис 2.11. Перша головна компонента

Нова вісь відображає найбільшу дисперсію після PC1.

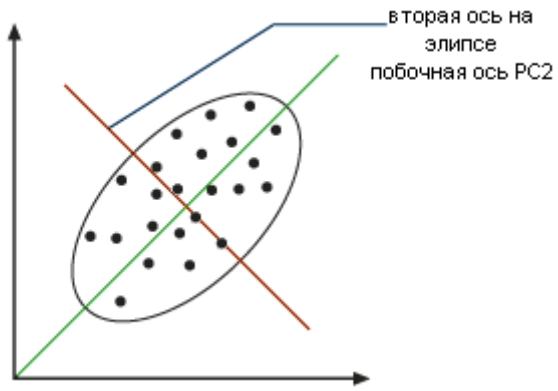

Рис. 2.12 Друга головна компонента

При використанні власних векторів, власних значень і обчисленої коваріаційної матриці вхідних даних багатоканального набору створюється лінійна формула, яка визначає зрушення і поворот. Ця формула застосовується для трансформації кожного значення відносно нової осі.

Завдання аналізу головних компонент має, як мінімум, чотири базових версії:

- апроксимувати дані лінійними залежностями меншої розмірності;

- знайти підпростір меншої розмірності, в ортогональній проекції на який розкид даних (тобто середньоквадратичне відхилення від середнього значення) максимальний;
- знайти підпростір меншої розмірності, в ортогональноій проекції на який середньоквадратична відстань між точками максимальна;
- для досліджуваної багатовимірної випадкової величини побудувати таке ортогональное перетворення координат, в результаті якого кореляції між окремими координатами будуть дорівнювати нулю. Перші три версії оперують кінцевими множинами даних. Вони еквівалентні і не використовують жодної гіпотези про статистичний породженні даних. Четверта версія оперує випадковими величинами. Кінцеві множини розглядаються як вибірки з даного розподілу, а рішення трьох перших завдань - як наближення до розкладання по теоремі Кархунена - Лоєва («істинного перетворення Кархунена - Лоєва») [28].

Висновки до другого розділу

Огляд та порівняльний аналіз чотирьох методів машинного навчання показав, що найбільш доцільним серед них для вирішення поставленої задачі є метод Support Vector Machine (SVM). В методі SVM наявні тренувальні дані (класу) для порівняння з тестовими даними, що надходять у систему. На відміну від методу Local Outlier Factor, де використовується поняття щільності, метод SVM використовує гіперплощини для розподілення та порівняння даних, що є більш наближеним до задачі детектування аномальної поведінки. Метод PCA дозволяє зменшити дані для пошуку аномалій, але може призводити до значних похибок. На відміну від PCA, метод SVM дозволяє чітко розділити гіперплощини та знайти. Метод SVDD є більш складною похідною від SVM і використовує чотиривимірну модель, що утворює додаткові ускладення, які не є доцільними в рамках розв’язання

задачі дисертаційної магістерської роботи, де використовується двовимірна модель, тому для подальших досліджень було обрано метод SVM.

РОЗДІЛ 3. ВИЗНАЧЕННЯ АНОМАЛЬНОЇ ПОВЕДІНКИ ЗА ДОПОМОГОЮ ANOMALY DETECTION У MICROGRID

3.1 Розробка сервісу детектування аномалій

В аналізі даних методами машинного навчання є два напрямки, які займаються пошуком аномалій: детектування аномалій (Outlier Detection) і детектування новизни (Novelty Detection). В першому випадку детектується вихід значень досліджуваних параметрів за межі області, яка в результаті аналізу була встановлена як «нормальна». В другому випадку реєструється поява «нового об’єкту» - відсутнього у наявній базі даних, але не аномального.

Як «викид», так і «новий об’єкт» відрізняється від об’єктів наявної навчальної вибірки. Проте на відміну від викиду, у майбутньому цей об’єкт повинен бути включеним до навчальної вибірки. Таким чином, межі «області нормальності» динамічно змінюються, вміщуючи виявлені нові об’єкти.

Наприклад, при аналізі виміряних значень температури навколишнього середовища відкидаються аномально великі або маленькі значення, що відноситься до детектування аномалій. Якщо ж алгоритм аналізу передбачає оцінку кожного нового значення у термінах «схожості» на наявні значення у навчальній вибірці – це детектування новизни.

В термінах задачі виявлення аномальної поведінки людини важливим є детектування як аномалій значень на координатній площині параметрів поведінки, так і детектування новизни – поява нових поведінкових шаблонів, що є відмінними від наявних у навчальній вибірці та підлягають обробці як новий варіант «норми».

Викиди виникають внаслідок:

- помилок у даних (неточності вимірювання, округлення, невірного запису);

- наявності завад, що спричиняють невірну класифікацію об'єктів, тобто їх помилкове віднесення до певних груп;

- присутності об'єктів «сторонніх» вибірок (наприклад, даних з датчика, що вийшов з ладу).

Суб’єктивні фактори, в свою чергу, здатні обумовити появу аномалій внаслідок помилкової інтерпретації або несанкціонованого втручання людини у роботу технічного обладнання.

При аналізі поведінкових характеристик постає задача вибору з усієї сукупності параметрів, що описують поведінку та переміщення людини, саме тих вирішальних параметрів, на підставі яких можна детектувати викиди або новизну поведінки.

В найпростішому випадку одним з таких вирішальних параметрів може виступати кількість спрацьовувань датчика руху в одному і тому самому приміщенні протягом певного невеликого інтервалу часу τ (наприклад, τ=5 хв), а другим – усереднене значення енергії споживання за цей самий інтервал. Такий підхід дозволить виявити випадки швидкого «безсистемного» переміщення людини, що може бути наслідком знаходження у стані афекту, наляканості чи психологічного розладу, адже однакова кількість спрацювань одного датчику протягом 5 хвилин розглядається як «аномалія», а протягом години – як «норма».

Незвичну активність людини можна детектувати за допомогою аналізу даних з датчиків (рис. 3.1), які підключаються до центрального контроллера (ЦК), що надсилає інформацію на віддалений сервер (ВС). В свою чергу, сервер передає дані у сервіс обробки та аналізу, а результат обробки відображується користувачу.

Сервіс складається з блоку «Back-end (ML)», в якому відбувається власне машинне навчання - обробка даних, формування навчальної вибірки та подальший аналіз, результат якого надходить до блоку «Front-end». Блок

«Front-end» відповідає за відображення результатів машинного навчання користувачеві.

Слід зазначити, що весь процес проходить без втручання користувача, якому надається можливість лише переглядати виявлені викиди – аномалії.

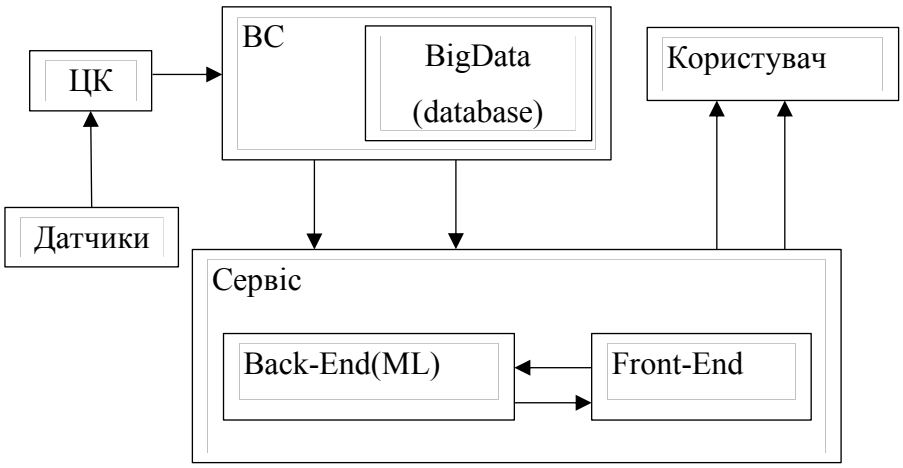

Рис. 3.1 Робота сервісу детектування аномалій

Послідовність дій при детектуванні аномалій поведінки людини у MicroGrid містить три основних етапи:

1. вимірювання і накопичення даних (Big Data).
2. аналіз даних – виявлення «областей нормальності».
3. застосування результатів навчання для аналізу поточної активності.

На першому етапі відбувається збір даних для моделювання та адаптації інформаційного середовища, що досліджується. Спрацювання відповідних датчиків розцінюються як події, фіксуються у часі та заносяться у базу даних. Історія спостережуваних подій датчиків відображає дії, які відбуваються в навколишньому середовищі і можуть бути використані для виявлення часто повторюваних моделей поведінкової активності. При цьому визначаються різні варіанти активності повсякденного життя з урахуванням зовнішніх факторів (час доби, сезонність, вихідні/святкові дні, тощо), а також виявляються аномальні стани.

Для ілюстрації роботи алгоритму детектування аномальної поведінки було розглянуто типову 2-кімнатну квартиру, план якої представлено на рис.3.2. У роботі алгоритму беруть участь 32 встановлених датчики, з них 14 датчиків відкриття дверей або вікон, 10 датчиків руху і 8 датчиків світла.

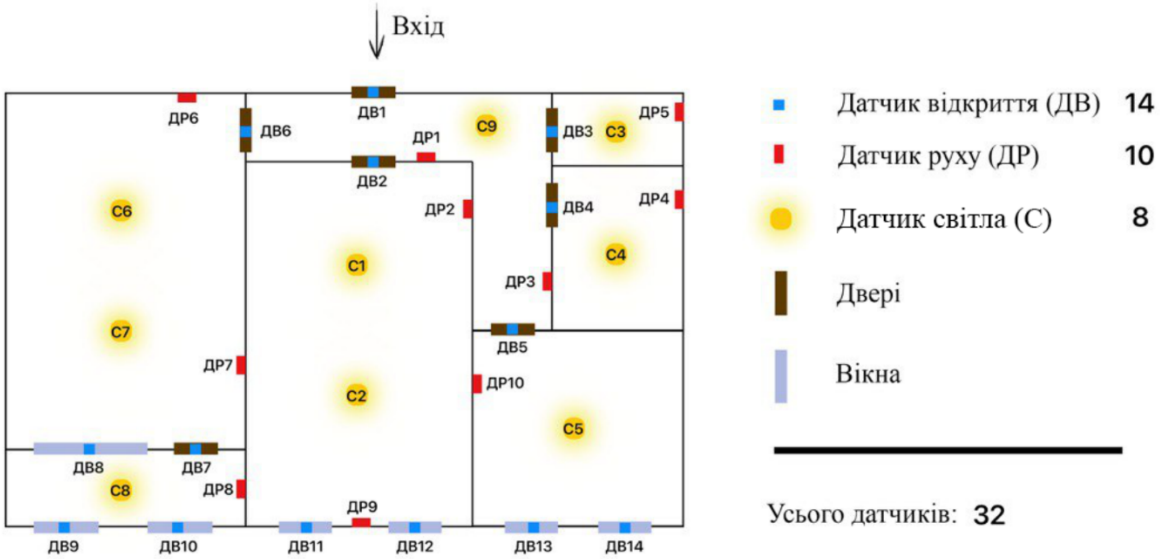

Рис. 3.2. План встановлення датчиків у 2-кімнатній квартирі

Дані сприйняття перетворюються на символічне подання, щоб полегшити подібність зібраних даних із поточним нормальним шаблоном у послідовності подій. Дані сприйняття, зібрані за часовий інтервал τ, визначаються як часові ряди: {S1, S2 ... St}. За допомогою SAX дані можна визначити як рядок алфавіту: {C1, C2 ... Cn}, де n позначає довжину символізованого рядка, має бути набагато менше τ. Дані перетвореного датчика складають дату та час, коли дані були зібрані, ідентифікація датчика (ID) та стан відповідних датчиків (ON / OFF) або (OPEN / CLOSE). Цей крок визначається як попередня обробка сприйманих даних. Дані датчика можуть бути помічені, наприклад, шляхом розгляду руху датчиків, представлених «M», температурою датчиків на «T» та дверцята датчиків за допомогою «D». Вони включають різні поведінки користувачів та пов'язані з ними дії користувачів, які імітують тренувальну операцію для методу AD.

Вказані датчики реєструють настання певного виду подій (відкриття двері чи вікна), поява людини у приміщенні, вмикання людиною світла у кімнаті. Дані від датчиків передаються у центральний контролер, який обробляє отримані дані і надсилає їх на віддалений сервер (див. рис.3.1). На сервері формується база даних у вигляді таблиці історії подій.

Формат даних, занесених у таблицю історії подій, містить 4 поля: ітерація, дата, час, назва датчика. Приклад фрагменту такої історії подій наведено у табл. 3.1, де відображено події спрацювання датчиків, що мають умовні позначення згідно рис. 3.2.

Таблиця 3.1.

№	Дата	Час	Сенсор
1	02.10.2018	14:00	ДВ1
2	02.10.2018	14:00	ДР1
3	02.10.2018	14:02	С9
4	02.10.2018	14:03	ДВ4
5	02.10.2018	14:03	ДР3
6	02.10.2018	14:04	ДВ4
7	02.10.2018	14:04	ДР4
8	02.10.2018	14:04	С4
9	02.10.2018	14:08	ДВ4
10	02.10.2018	14:08	ДР3
11	02.10.2018	14:09	ДВ5
12	02.10.2018	14:09	ДР10
13	02.10.2018	14:10	С5
14	02.10.2018	14:13	ДВ13
15	02.10.2018	14:14	ДВ14
16	02.10.2018	14:30	ДВ5
17	02.10.2018	14:30	ДР3
18	02.10.2018	14:30	С9

На рис.3.3 зображено логічну схему послідовності дій машинного навчання на другому (аналіз та навчання) та третьому (тестування поточного режиму) етапах.

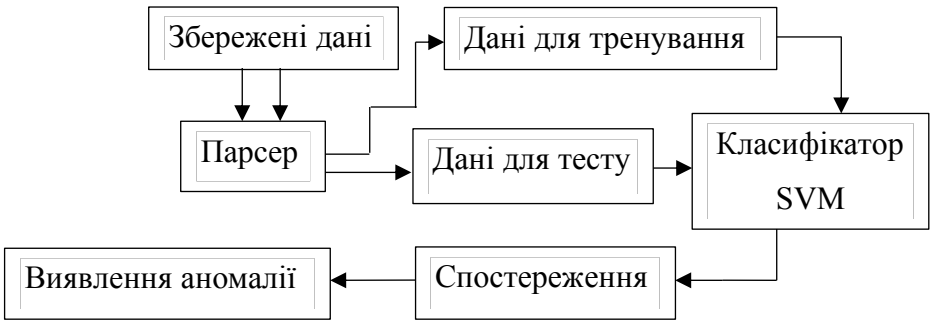

Рис. 3.3 Машинне навчання

1) Блок “Збережені дані” – накопичені дані після вимірювання (Big Data).

2) Блок «Парсер» як інструмент синтаксичного аналізу перетворює вхідні дані в структурований формат з введенням додаткових атрибутів для підвищення ефективності обробки набору даних.

3) Блок “Дані для тренування” – дані що накопичились за певний проміжок часу t.

4) Блок “Дані для тесту” – дані що надходять до системи для виявлення: чи є аномалія у даному наборі даних.

5) Блок “Класифікатор SVM” (Support Vector Machine) контролюється заданими моделями та алгоритмами навчання, які аналізують тренувальні та тестові дані, що використовуються для класифікації та регресійного аналізу. Враховуючи набір навчальних прикладів, кожен з яких позначається як такий, що належить до однієї з двох категорій, SVM-модель являє собою представлення тестового прикладу як точки в просторі, мапованому таким чином, що приклади окремих категорій діляться явним розривом. Нові приклади потім вносяться у той самий простір і беруть участь у подальшому аналізі вже як елементи навчальної вибірки.

6) Блок “Спостереження” – для якісної оцінки, необхідно спостерігати за системою щоб виділити що є нормою, а що є аномалією. Необхідний експерт для оцінки.

7) Блок “Виявлення аномалії” – виявлення аномалії після спостереження і оцінки даних.

За допомогою алгоритму Anomaly Detection сервіс виявляє аномалії і попереджає про них користувача. Оскільки мікроконтролер передає дані через інтерфейс Rest API, доцільно використовувати Web-інтерфейс в якості Front-end частини. Для моделювання найпростішого варіанту аналізу поведінкових моделей активності людини за допомогою методу Anomaly Detection в якості вирішальних параметрів було обрано: **Ознака 1** - кількість N спрацьовувань датчика руху протягом кожного з послідовних періодів тривалістю 5 хвилин, **Ознака 2 –** усереднене електроспоживання W за цей період. Значення цих ознак за період спостереження утворюють зразкову матрицю – множину двоелементних векторів (K=2 – кількість ознак, що беруться до розгляду). Приклад накопичених даних значень цих ознак наведено у табл. 3.2.

Таблиця 3.2

№	Час	W, Вт	Кількість, N
1	00:00-00:05	20	**50**
2	00:05-00:10	30	40
3	00:10-00:15	25	35
4	00:15-00:20	27	40
…	…	…	…
127	10:30-10:35	20	**10**
128	10:35-10:40	25	30
…	…	…	…

Отримані результати наносяться на координатну площину параметрів N та W. В подальшому графічно-аналітичними методами Anomaly Detection визначається «зона нормальності» (рис.3.4).

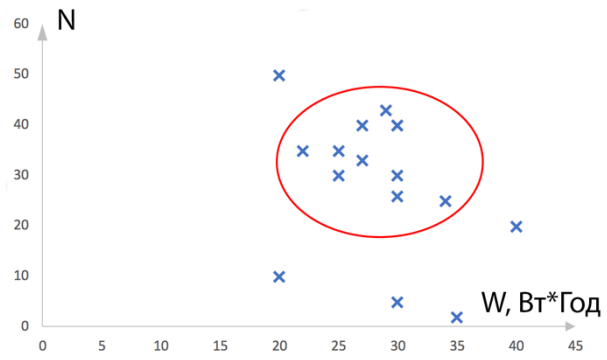

Рис. 3.4 Приклад зони нормальності

Для формування цієї зони використовуються різні підходи, зокрема, з урахуванням експертних оцінок та даних попередніх періодів спостережень. Наявні на графіку відхилення вважаються аномаліями і потребують втручання інших сервісів інформаційно-програмного забезпечення MicroGrid для подальшого відпрацювання – активації тривоги, сповіщення користувача, тощо.

Виявлення меж «зони нормальності» може бути полегшено за рахунок попередньої кластеризації. Тоді кластери суттєво меншого розміру, швидше за все, відносяться до аномалій.

Для навчання алгоритмів машинного навчання величезну кількість даних збирається з багатьох датчиків в IoT [12]. Однак через використання цього великої кількості даних кращий алгоритм навчання контролюється неконтрольованим алгоритмом навчання. Дійсно, алгоритм кластеризації має можливість ефективно обчислювати дані і групувати подібні шаблони активності користувачів в кластери.

Використання алгоритму кластеризації в повинно призвести до виявлення тимчасових відносин шляхом обробки сприйманих даних. Дійсно, обробка сприйманих даних являє собою перший крок у визначенні активності користувачів в розумному будинку. Таким чином, алгоритм кластеризації моделі відповідає методології, яка складається з етапу перетворення сприйманих даних датчиків, найбільш частому спостереженні за схемою і етап видобутку та аналогічний етап групування.

У реальній системі MicroGrid кількість точок спостереження, що складають навчальну вибірку, і відповідно, кількість рядків у табл.3 буде збільшуватися. Постійне накопичення даних дозволяє збільшувати обсяг навчальної вибірки та відповідно динамічно змінювати «зону нормальності».

3.2 Поняття історії подій

Таблиця історії подій, яка постійно доповнюється новими даними, що надходять від датчиків у реальному масштабі часу, використовується для пошуку заданих послідовностей подій, які можуть виступати індикаторами аномальної поведінки. Послідовності, які потрібно відшукати, задаються експертом на базі попереднього досвіду роботи з таблицями історії подій та з урахуванням результатів першого етапу алгоритму – спостереження для даного MicroGrid. В роботі розглянуто три типи аномальної поведінки з відповідними індикаторами, наведеними у табл.3.3. Умови аномалії визначаються за обраний невеликий проміжок часу τ (для прикладу розглянуто τ=5 хвилин).

Таблиця 3.3

№	Подія	Умова аномалії	Типи датчиків	Індикатор аномалії	Тип аномальної поведінки
1	Спрацювання датчиків руху (ДР)	Повторюваність комбінацій однієї групи більше N разів за час τ	ДР, ДВ однієї групи	Комбінація послідовних спрацьовувань датчиків	Безсистемне, хаотичне переміщення людини у приміщенні
2	Спрацювання датчиків світла (С)	Повторюваність події ON > N разів за час τ	ДС	Значна кількість перемикань за короткий проміжок часу	Багаторазове часте вмикання/ вимикання світла
3	Переміщення людини у приміщенні більше зазначенного значення	Зміна Group > 10 разів за час τ	ДР, ДВ різних груп	Значна кількість переміщення людини за проміжок часу	Безсистемне, хаотичне переміщення людини між кімнатами

Розбиття датчиків на групи. Індикатори подій можуть бути як простими, які базуються на спрацьовуванні одного датчику (як, наприклад, багаторазове вмикання/вимикання світла або відкриття дверей протягом заданого проміжку часу), так і складними, коли індикатором аномалії виступає певна послідовність подій, що фіксуються різними датчиками. Прикладом аномальної поведінки зі складним індикатором є багаторазове переміщення людини всередині квартири за одним і тим самим маршрутом протягом заданого невеликого інтервалу часу без явно вираженої мети.

На відміну від простого випадку (табл. 3.1), де в якості параметрів було взято кількість спрацьовувань одного датчику за інтервал часу в залежності від електроспоживання, для виявлення переміщення людини між кімнатами був розроблений більш складний алгоритм, що дозволяє виявити швидке пересування між приміщеннями (кожному приміщенню відповідає своя група (Group) датчиків, що позначається відповідною літерою A, B, C,…, див. рис.3.5).

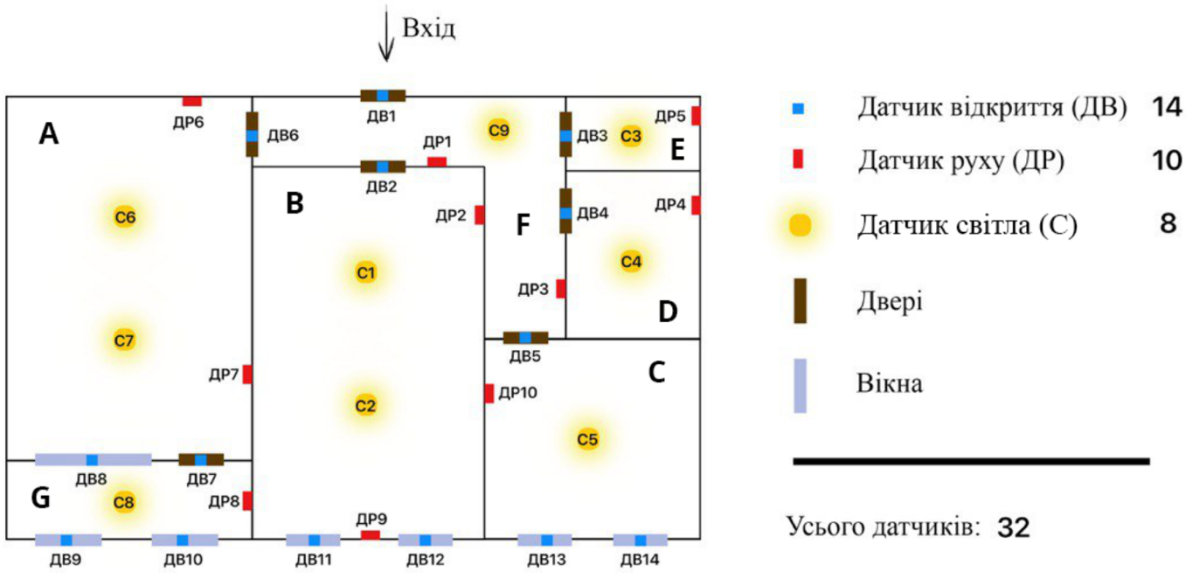

Рис. 3.5. Розподілення приміщень за групами (Group)

Зафіксувати момент переходу людини з одного приміщення в інше можна шляхом перевірки умови, чи змінилося значення величини Group (з A

на В, тощо). На рис. 3.6 наведено алгоритм детектування аномалій поведінки, пов’язаних з пересуванням людини між приміщеннями (спрацьовування датчиків з різних груп).

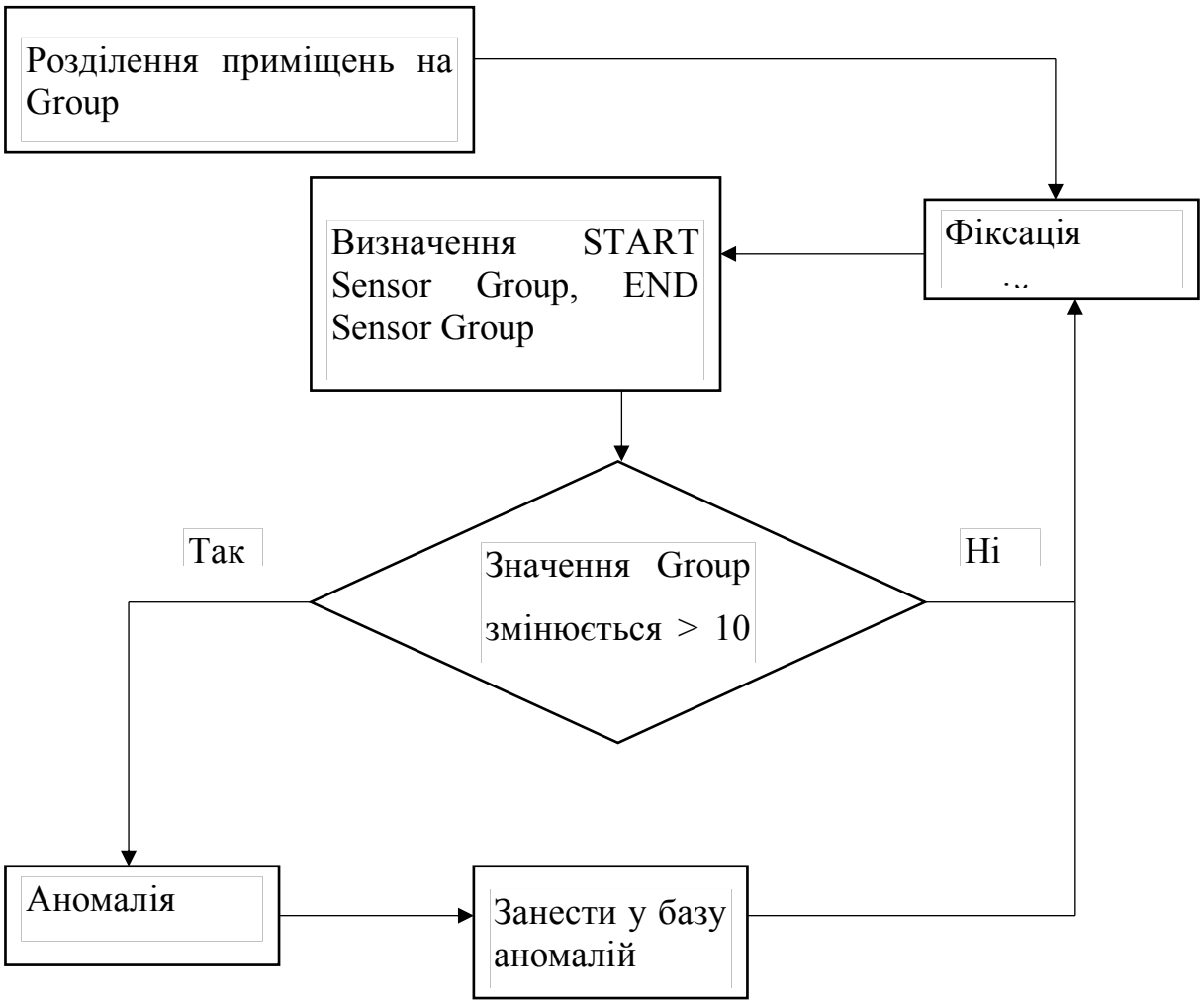

Рис. 3.6 Алгоритм детектування аномалій поведінки

1) Блок “Розділення приміщення на Group” – визначає поділ всіх приміщеннь у будинку на умовні групи (надалі Group)

2) Блок “Фіксація подій” – у реальному часі фіксуються події спрацювання датчиків які додаються у таблицю історії подій.

3) Надалі визначається перше спрацювання будь-якого датчика (START Sensor Group) у відповідній групі (Group) приміщення та останнє спрацювання будь-якого датчика (END Sensor Group) у відповідній групі (Group) приміщення.

4) Перевіряється значення Group. Якщо значення Group змінюється за проміжок часу τ (наприклад, 5хв) більше, ніж 10 разів, це свідчить про аномалію, що заноситься до бази аномалій і система надалі фіксує події. Далі робота алгоритму продовжується циклічно.

Для прикладу аномалії розглянутого типу таблиця історії подій набуває наступного вигляду (табл.3.4).

Таблиця 3.4

№	Дата	Час	Датчик	Подія	Group
1	02.10.2018	14:00	ДВ1	ON	A
2	02.10.2018	14:00	ДР1	ON	A
3	02.10.2018	14:02	С9	ON	A
4	02.10.2018	14:03	ДР3	ON	A
5	02.10.2018	14:03	ДВ4	ON	B
6	02.10.2018	14:04	С4	ON	B
7	02.10.2018	14:04	ДВ4	ON	B
8	02.10.2018	14:04	ДР3	ON	A
9	02.10.2018	14:08	ДВ5	ON	C
10	02.10.2018	14:08	ДР10	ON	C
11	02.10.2018	14:09	С5	ON	C
12	02.10.2018	14:09	ДВ13	ON	C
13	02.10.2018	14:10	ДВ14	ON	C
14	02.10.2018	14:13	ДВ5	ON	C
15	02.10.2018	14:14	ДР3	ON	A
16	02.10.2018	14:30	С9	ON	A

Як видно з табл.3.4, рядки №№4-9 містять дані, що відповідають аномальній поведінці. Задачею сервісу Anomaly Detection є відшукання заданих простих та складних індикаторів у таблиці історії подій.

3.3 Приклад сервісу детектування аномалій

В якості дослідження обирається будинок, що містить кімнати з предустановленими датчиками руху, електрочутливості, датчиків відкривання/закривання дверей (див. рис 3.5).

Функціонування сервісів машинного навчання та їх взаємодія з блоками MicroGrid і користувачем наведено на рис. 3.7

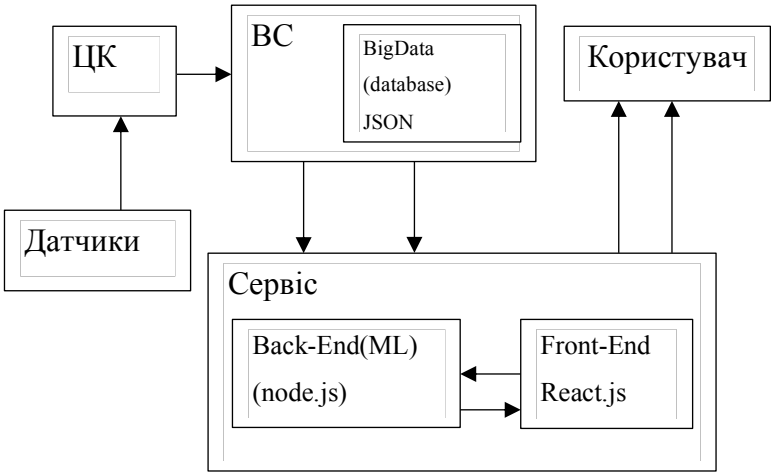

Рис. 3.7 Робота сервісів машинного навчання у MicroGrid

Датчики надсилають дані у центральний контроллер у форматі даних JavaScript Object Notation (JSON) з використанням архітектурного стилю взаємодії компонентів розподіленого додатка в мережі REST API (Representational State Transfer - Application Programming Interface).

Для передачі даних у центальний контроллер необхідно зв'язати датчики та ЦК за допомогою бездротової передачі даних локальної мережі (Wi-Fi). Для цього до кожного датчику необхідно під'єднати модуль, що вміє відправляти і отримувати інформацію в локальну мережу або в інтернет за допомогою Wi-Fi. В якості прикладу даного модулю можна обрати NodeMCU - це платформа на основі модуля ESP8266, що може керувати різними схемами на відстані за допомогою передачі сигналу в локальну мережу або інтернет через Wi-Fi. На базі Node MCU можна створити «розумний будинок», налаштувавши управління світлом або вентиляцією через телефон, реєстрацію показань датчиків і багато іншого. Розмір плати NodeMCU - 6 х 3 см. Плата досить компактна (рис. 3.8), це дозволяє використовувати її для різних застосувань. Виводи плати NodeMCU розташовані так, що її можливо легко під'єднати до стандартих макетних

плат (breadboard). На лицьовій частині плати є роз'єм Micro USB, за допомогою якого в контролер завантажують програмний код або подається живлення від powerbank або комп'ютера. Найбільше місця на платі займає чіп ESP8266, на якому розташований мікропроцесор з тактовою частотою 80 МГц (можна розігнати до 160 МГц). Плата має 4 мегабайта Flash-пам'яті. Для живлення на плату можна подавати напругу від 5 до 12 В, але рекомендується від 10 В. Живлення можна здійснювати від Micro USB, так і від контакту Vin. Існують додаткові плати розширення для зручного живлення модулів.

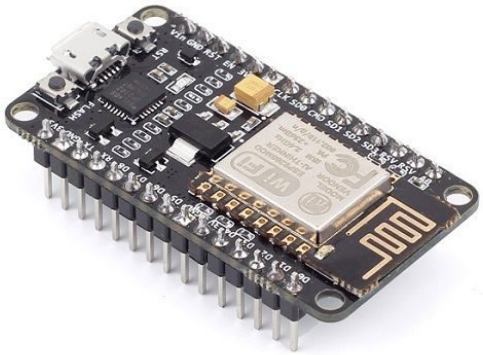

Рис 3.8. Плата ESP8266 (NodeMCU)

Вартість NodeMCU на відміну від Arduino та Raspberry Pi досягає ~ 3$, що дозволяє економити на “розумних датчиках”. В якості датчика руху можливо використовувати, наприклад, HC-SR501 (рис. 3.9). Датчик може виявляти людину на відстані до 6 метрів.

Рис 3.9. HC-SR501 піроелектричний інфрачервоний датчик руху PIR

Підключення PIR-датчиків до NodeMCU дійсно просте. PIR виступає як цифровий вихід. Більшість PIR-модулів мають 3-контактні з'єднання збоку або внизу. Живлення зазвичай становить 3-5 В. З'єднання схем зроблені, як показано на рис 3.10.

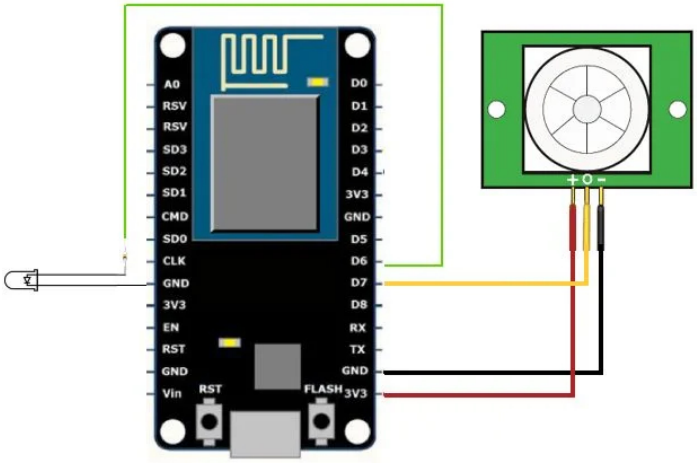

Рис 3.10. Підключення PIR до NodeMCU

Вивід Vcc HC-SR501 підключено до + 3В NodeMCU. Вихідний контакт HC-SR501 підключений до цифрового контакту D7 NodeMCU. Вивід GND HC-SR501 підключено до заземлення (GND) NodeMCU. У цьому прикладі необхідно підключити анодний контактний індикатор світлодіода до цифрового дроту D6 і катодний штифт до контакту GND NodeMCU. Наступним кроком є написання програми для посилання JSON об’єктів до ЦК. У свою чергу, центральний контроллер, підключений до мережі інтернет, збирає дані з датчика і надсилає їх на віддалений сервер (ВС). Усі дані, що фіксуються з центрального контроллера, надходять на ВС у реальному часі. ВС через спеціальний ключ доступу додає дані у нереляційну базу даних, що з часом стає дуже великою через великий обсяг даних і відноситься до категорії BigData.

Сервіс, зображений на рис. 3.7, складається з двох частин, що взаємодіють між собою: клієнтського користувацького інтерфейсу (Front-End) та програмно-апаратної частини (Back-End). Сервіс кможе знаходитись на віддаленому сервері, іншому сервері, або на персональному комп’ютері

користувача. Спочатку дані надходять у Back-End частину, де оброблюються для відображення на користувацькому інтерфейсі (Front-End) для перегляду користувачем.

Алгоритм роботи програми. За алгоритмом, зображеним на рис. 3.11, програма працює в наступному порядку. Спочатку йде ініціалізація програми, завантаження середовища та ініціалізація робочого оточення програми. Після цього здійснюється генерація часових інтервалів за кожен день, де система оброблює дані отримані з бази даних віддаленого сервера і записує, скільки переміщень було зроблено людиною за n-інтервал часу, при цьому фіксується дата та день тижня. Наступним кроком є створення векторів інтервалів часу різних днів, тобто створюється окремий вектор для кожного проміжку часу на основі обробленої бази з попереднього кроку, в які заносяться кількість переміщень людини за даний проміжок, за всі дні. Далі система детектує аномалії за певні проміжки часу: оброблює кожен проміжок часу за певний день тижня, з цього вектору формує тренувальний вектор . Дані тренувального векторупорівнюються з даними щодо кількості переміщень людини за певний проміжок часу поточної ітерації перебору об’єктів даних дня. Якщо аномалію зафіксовано, поточний вектор заноситься у базу аномалій. Після цього кроку система виводить дані користувачеві, відображає графіки активності користувача, аномальну межу, позначає аномалії на графіку та генерує відповідні таблиці аномалій.

Висновки до третього розділу

В результаті розглядання видів аномалій поведінкової активності людини у MicroGrid розроблено структуру окремих блоків сервіс машинного навчання, досліджено особливості взаємодії з датчиками, створено алгоритм пошуку аномалій на базі методу опорних векторів SVM.

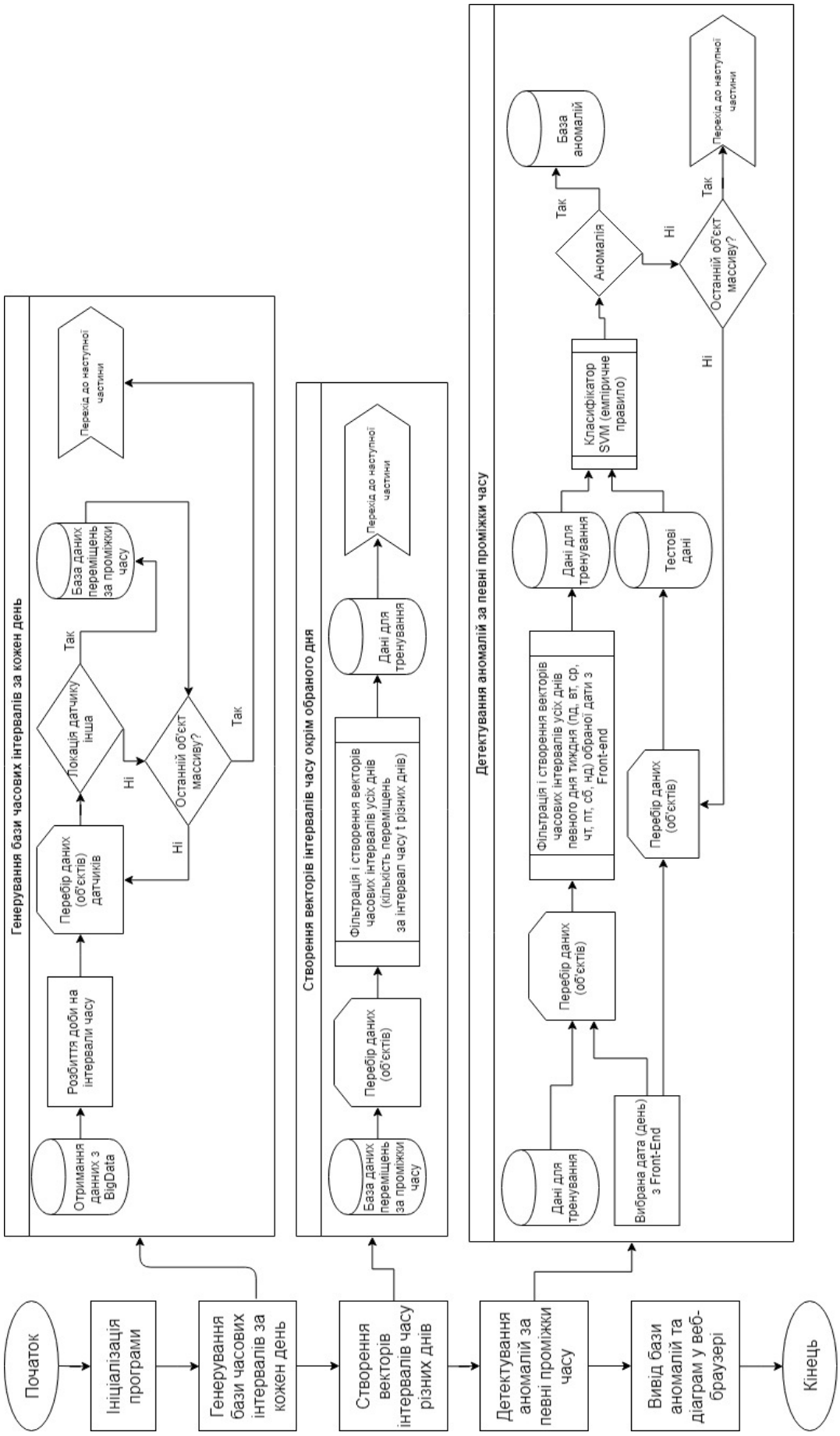

Рис 3.11. Алгоритм роботи програми детектування аномалій поведінки

РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОЇ ЧАСТИНИ

4.1 Опис Back-End частини

В якості **Back-End** частини сервісу детектування аномалій було розроблено компонент пошуку аномальних даних у масиві данних.

Дані, що надходять до масиву - це кількість змін величини Group (визначеної групи приміщень, в яких зафіксовано переміщення людини) за проміжок часу τ. В якості **бази даних** було обрано таблицю історії подій, що була попередньо згенерована за допомогою моделювання поведінки людини у приміщеннях за 3 місяці (див. рис 3.5).

Для розробки програми було обране веб-середовище як універсальний метод доступу до програми з будь-якого пристрою, що підтримує веб-браузер.

Метод детектування аномалій у масиві даних базується на емпіричному правилі «трьох сигм», або «68-95-99,7» [27].

У статистиці правило «68-95-99,7» - це скорочення, яке використовується для вказання імовірності подій, які характеризують знаходження аналізованих значеньв межах смуги навколо середнього значення при нормальному законі розподілу випадкової величини з шириною двох, чотирьох та шести стандартних відхилень, відповідно. Точніше, ці імовірності складають 68,27%, 95,45% та 99,73%, тобто відповідні частки всіх значень випадкової величини лежать в межах одного, двох та трьох стандартних відхилень від середнього значення відповідно.

При проведенні обчислень за одиницю виміру відхилення випадкової величини, що підпорядковується нормальному закону розподілу, від центру величини розсіювання (математичного очікування) приймається середньоквадратичне відхилення σ. Тоді на підставі виразу:

$$P(-l < \bar{x} < l) = \widehat{\Phi}(\frac{l}{E}) \tag{4.1}$$

випливають корисні при різних обчисленнях рівності:

$$P(-\sigma < \bar{x} < \sigma) = \Phi(\frac{1}{\sqrt{2}}) = 0.683,$$

$$P(-2\sigma < \bar{x} < 2\sigma) = \Phi(\sqrt{2}) = 0.954,$$

$$P(-3\sigma < \bar{x} < 3\sigma) = \Phi(\frac{3}{\sqrt{2}}) = 0.997.$$

Ці результати геометрично зображені на рис 4.1.

Майже достовірно, що випадкова величина (помилка) не відхиляється від математичного очікування по абсолютній величині більше ніж на 3σ. Це припущення називається **“правилом 3х сигм”** [27].

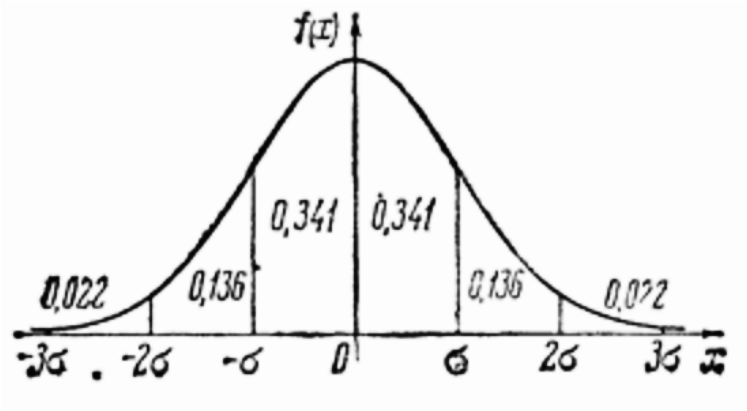

Рис. 4.1 Нормальний розподіл випадкової величини

Для приблизної нормальної області даних значення в межах одного стандартного відхилення середнього значення складають близько 68% встановленого; в межах двох стандартних відхилень - близько 95%; в межах трьох стандартних відхилень - близько 99,7%. Наведені значення являють собою теоретичні ймовірності, призначені для наближеної оцінки емпіричних даних, отриманих від нормальної вибірки.

Пошук аномалій на базі правила трьох сигм. Пошук аномалій базується на правилі 3-х сигм. Спочатку дані тренування (normalValues) і тестові дані (value) заносять в якості аргументів до функції find(), що повертає значення, чи є тестові дані аномалією. Програмний код цієї частини виглядає наступним чином:

```
  find(normalValues, value) {
    const ex = this.expectedValue(normalValues);
    const stdv = this.standardDeviation(normalValues, ex);
    return {
      thisAnomaly: !(value <= (3 * stdv) + ex),
      anomalyZone: (3 * stdv) + Math.abs(ex)
    };
  }
```

У програмному коді математичне очікування Mx позначається через «ex», що розраховується за допомогою функції expectedValue(). Математичне очікування розраховується за формулою:

$$Mx = \xi = M_1 = \sum_x xp(x) = \frac{1}{m}\sum_{i=1}^{n} x_i;$$

```
  expectedValue(values, pow = 1) {
    const accuracy = 0.1;
    let sum = 0;
    if (!values.length) return 0;
    for (let i = 0; i < values.length; i++) {
      sum += Math.pow(values[i], pow) / accuracy;
    }
    return sum / (values.length / accuracy);
  }
```

accuracy – точність

stdv – середньоквадратичне відхилення, що розраховується за функцією standardDeviation().

```
standardDeviation(values, ex) {
  const ex2 = this.expectedValue(values, 2);
  ex = ex || this.expectedValue(values);
  return Math.sqrt(ex2 - Math.pow(ex, 2));
}
```

ex2 – математичне очікування в квадраті - Mx^2

$$Mx^2 = M_2 = \frac{1}{m}\sum_{i=1}^{n} x_i^2;\ M_2\text{- математичне очікування в квадраті}$$

σ - середньоквадратичне відхилення, Dx - дисперсія:

$$Dx = \sigma^2 = Mx^2 - \xi^2 = M_2 - M_1^2;$$
$$\sigma = \sqrt{M_2 - M_1^2};$$

Якщо умова $value >= 3\sigma + M_1$ виконується, тобто тестове значення більше, ніж $3\sigma + M_1$, це свідчить про наявність аномалії.

4.2 Інструменти та реалізація Back-End частини

Оскільки, як зазначалося раніше, для розробки програми було обрано веб-середовище, для реалізації алгоритму використовано мову програмування JavaScript. В якості Back-End складової було обрано середовище Node.js. Оскільки у JavaScript доцільно використовувати об’єкти, було вирішено використовувати нереляційну базу даних (NoSQL).

Основними перевагами NoSQL над реляційними базами даних (SQL) є:

1) базова доступність - запити гарантовано завершується (успішно чи безуспішно).

2) гнучкий стан - стан системи може змінюватися з часом, навіть без введення нових даних, з метою забезпечення узгодженність даних.

3) узгодженість в кінцевому рахунку - дані можуть бути деякий час неузгоджені, але приходять до узгодження через деякий час.

NoSQL володіє властивістю «ключ-значення», що є найпростішим сховищем даних, яке використовує ключ для доступу до значення. Такі сховища використовуються для зберігання в системах, спроектованих з прицілом на масштабованість.

Розробка програмного забезпечення, аналіз результатів та відповідні корегування логіки програми дослідження на аномальність проводилося в три ітерації.

На першому кроці (перша ітерація) було вирішено детектувати кількість переміщень людини за проміжок часу, збирати цю кількість за певні інтервали та додавати до спільної бази переміщень людини (рис. 4.2).

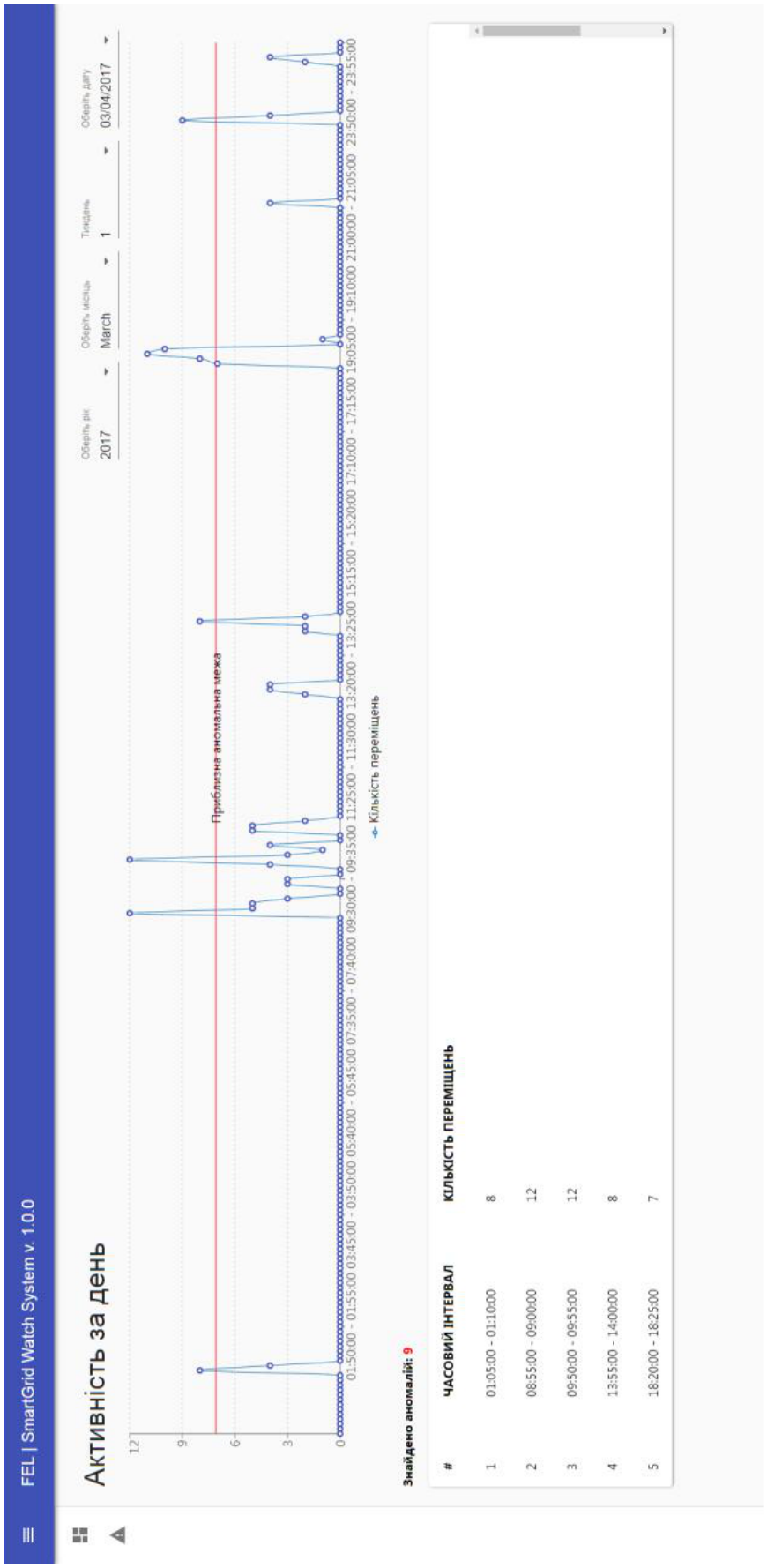

Рис 4.2. Детектування аномальної поведінки за проміжки часу усіх днів

Після цього система отримувала один вектор даних за усі дні і за усі проміжки часу і порівнювала з тестовими даними. Після перевірки програми було зроблено висновок щодо неточності результатів при такому підході, адже дані були різні – за різний проміжок часу і за різний час доби. Наприклад, система вважала нормальною переміщення людини вночі, рівно як і в денний час.

На другому кроці (друга ітерація) на підставі аналізу результатів було вирішено розглядати два вектори даних, а саме: дані щодо переміщень людини за проміжки часу кожен день та додатково вектор відповідності проміжків даних за різні дні. Таким чином було реалізовано прив'язку до часу доби. В результаті межа аномальності, визначена на етапі тренування, стала різною для різних проміжків часу, на відміну від першої ітерації, де межа аномальності була постійною величиною.Таким чином, система "навчилася", що вважається для людини нормою у кожен проміжок часового інтервалу і порівнювала попередні (training data) результати з новими (test data).

Наприклад, після проходження етапу навчання розглядається поточний інтервал часу 01:00:00-01:05:00 кожен день. В результаті формується вектор кількості переміщень протягом даного інтервалу часу, який доповнюється кожного дня і має вигляд:

[0,0,0,3,0,4,1,0 …].

Надалі цей вектор вважається тренувальним для цього інтервалу часу, система порівнює кожне нове значення кількості переміщень зі значеннями у масиві і виявляє для певного проміжку часу, чи аномальним є це нове значення.

Після тестувань точність пошуку аномалій у різний часовий інтервал збільшилася більше ніж в 3 рази порівняно з першою ітерацією. Наприклад, для дати 12.04.2017 р., де була кількість аномалій 26, у другій ітерації кількість знайдених аномалій становила 95.

Аномальна межа, що була раніше для всіх значень рівномірною, після другої ітерації стала східчастою, тобто для кожного часового інтервалу τ в результаті навчання формується своє значення аномальної межі.

Недоліком цієї ітерації виявилося те, що враховувався лише час доби, але не враховувався день тижня. Зважаючи на те, що поведінка людини може суттєво відрізнятися для різних днів тижня (особливо відчутною є різниця для вихідних та робочих днів), було прийнято рішення про врахування цього фактору для підвищення точності виявлення аномалій.

Відповідно, на третьому кроці (третя ітерація) до векторів тренувальних даних було додано ще один (третій) вектор, значення якого відповідають дням тижня. Система перевіряла усі дні за кожен інтервал з урахуванням дня, визначаючи тим самим, що слід вважати нормою за певний день і певний проміжок часу.

Для розширення вибірки тренувальних даних з урахуванням дня тижня було проведено дослідження більшої базою даних.

Приклади векторів даних, які використовуються для детектування аномалій поведінки на третьому кроці, зображено на рис 4.3.

Рис 4.3. Детектування аномалій з урахуванням днів тижня

Після реалізації третьої ітерації було виявлено, що точність детектування аномалій зросла більше ніж в 6.5 разів порівняно з першою ітерацією, та приблизно в 1.8 разів – порівняно з другою ітерацією (кількість знайдених аномалій для тієї ж дати становила 176).

Під час проведення трьох ітерацій розробки було досліджено питання необхідного обсягу даних для виявлення аномалій, адже розмір навчальної вибірки є важливим параметром алгоритму. Програма тестувалась на даних, що описують кількість переміщень людини у будинку у інтервали часу τ=5 хвилин протягом 3 місяців. Загальний обсяг накопичених даних за цей період склав близько 200 000 значень. Незважаючи навіть на цей значний обсяг, що обгрунтовує залучення методів машинного навчання для Big Data, було зроблено висновок, що для більш достовірного виявлення аномалії необхідно ще більше даних (краще за період не менше двох років).

4.3 Опис реалізації Front-End частини - користувацького інтерфейсу

В якості **Front-End** фреймворку було обрано React.js. За допомогою React.js можливо розроблювати сучасний веб-додаток, який працює без перезавантаження сторінки та передбачає легку взаємодію користувача із системою. Після верстки, обробки даних з Back-End було розроблено структуру користувацького інтерфейсу, зображену на рис 4.4. Програма ділиться на 2 сторінки, перехід на які відбувається без перезавантаження, а дані з датчиків отримуються з серверу асинхронно. На головній сторінці користувач може переглядати активність за поточний або обраний день та поточні покази датчиків. Меню, через яке можливо перейти у моніторинг аномалій, знаходиться у лівій частині програми. На сторінці ‘Аномальність’ знаходиться моніторинг аномалій, через який можливо переглядати наявні аномалії у системі для обраної дати. Також на екран виводиться список аномалій та кількість переміщень за певний час (рис. 4.5).

Код розробленого програмного забезпечення розміщено у відкритому доступі на інтернет-ресурсі за посиланням: https://github.com/Aqhefull/kpi-anomaly-detection.

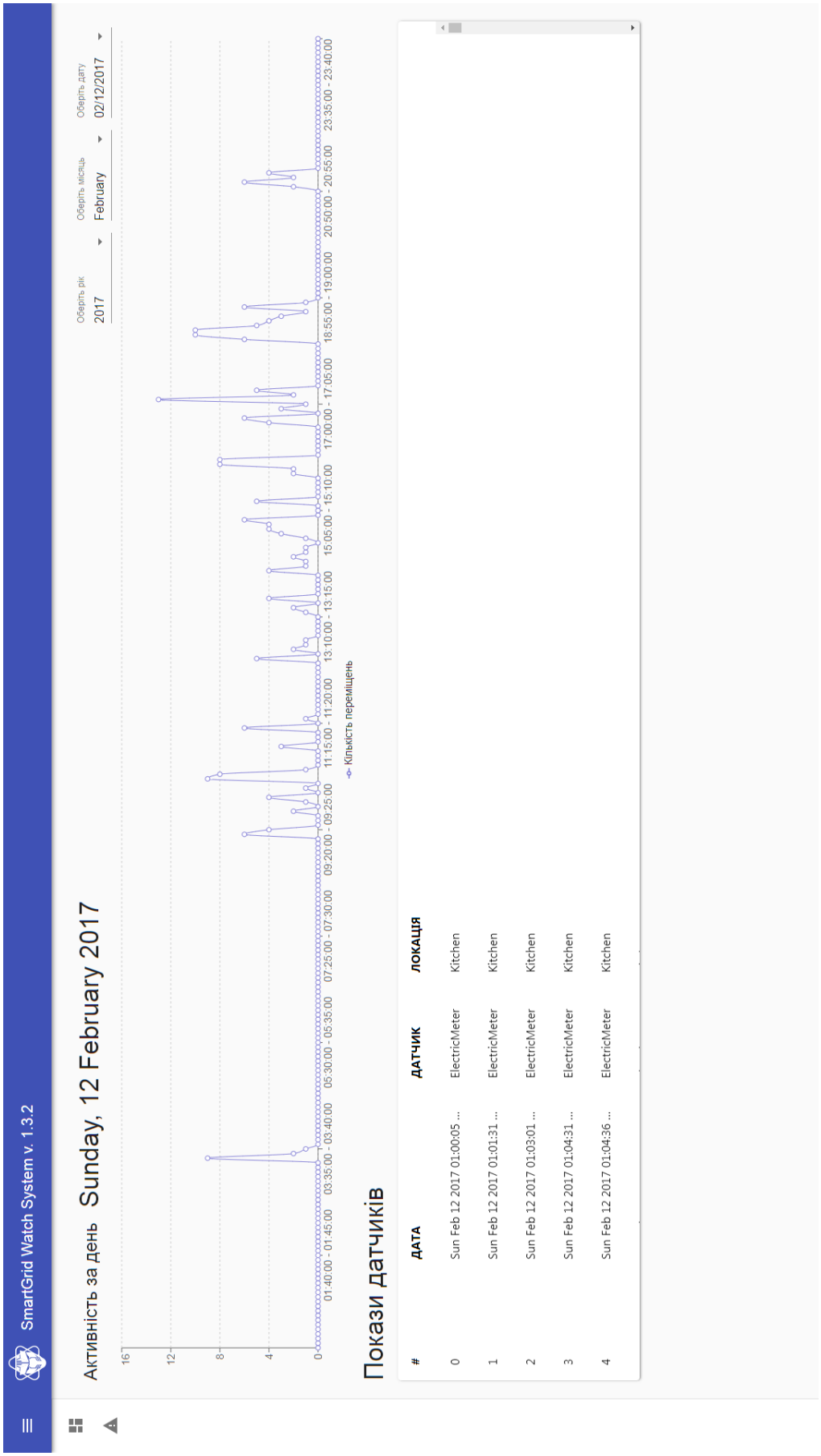

Рис 4.4. Головна сторінка користувацького інтерфейсу Front-End

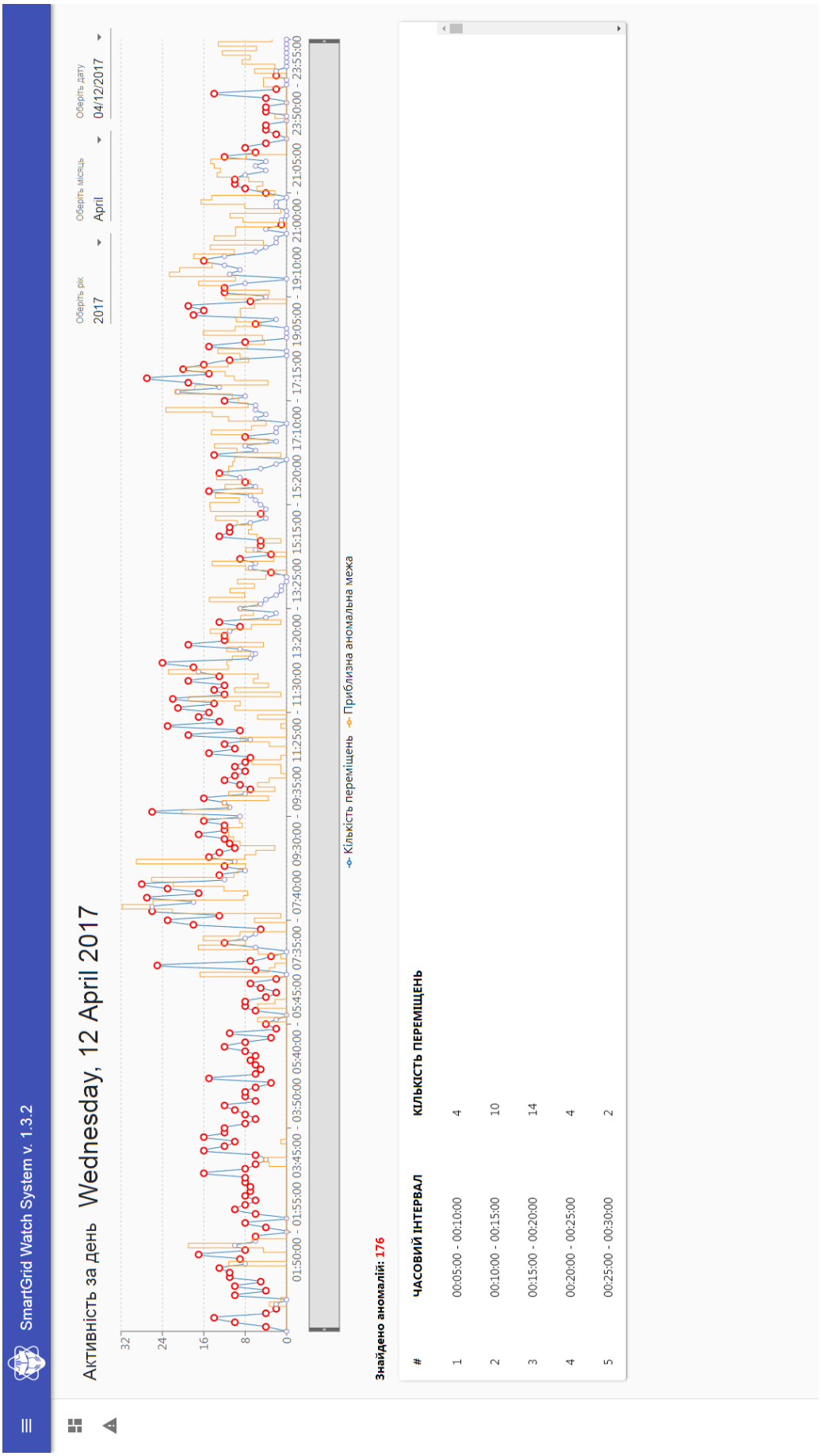

Рис 4.5. Сторінка аномальної активності

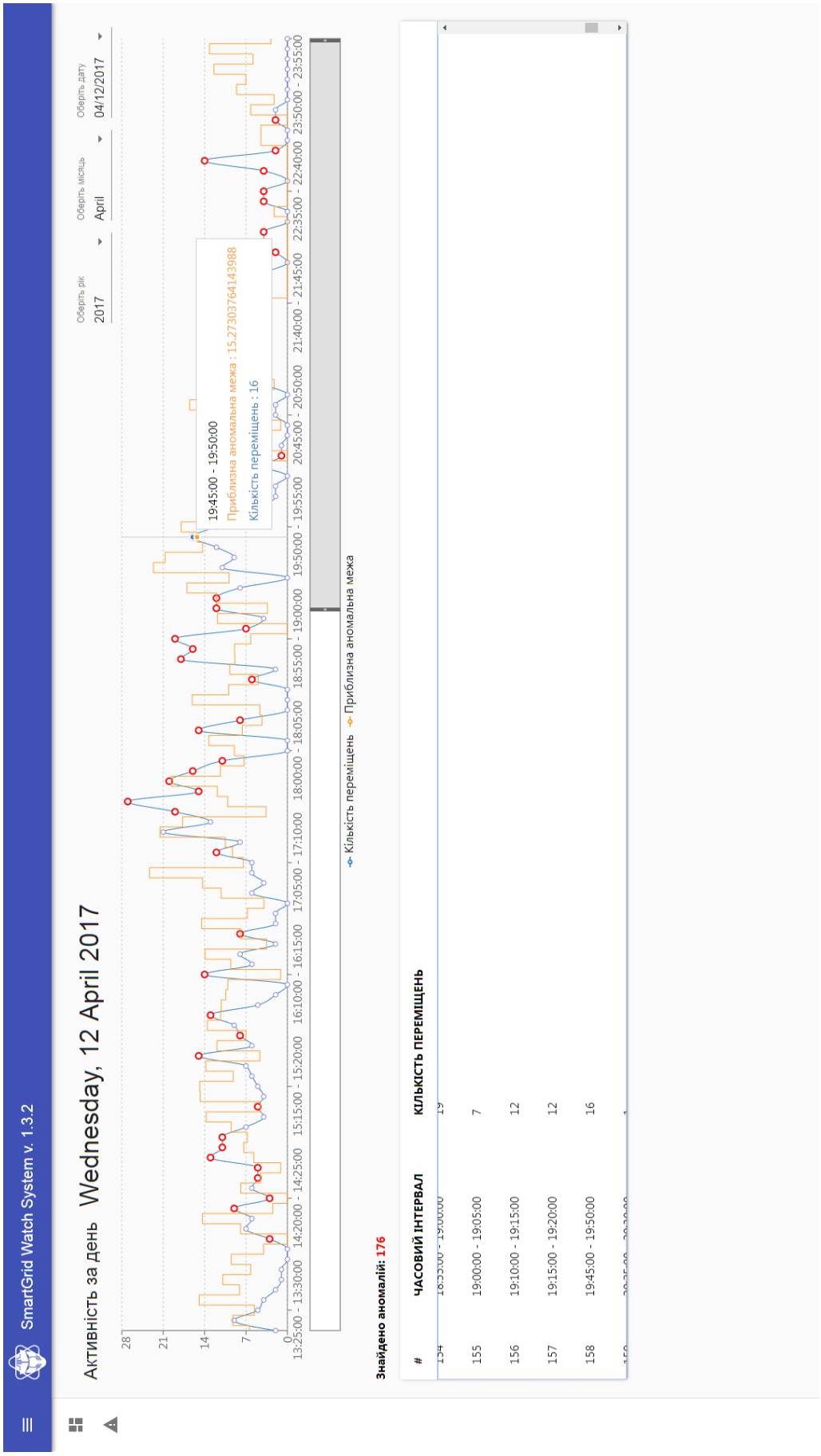

Рис 4.6. Сторінка аномальної активності (збільшений масштаб)

Висновки до четвертого розділу

Платформою для розробки програми обрано фреймворк React.js з використанням мови JavaScript, мови розмітки HTML та каскадних таблиць стилів CSS, розроблено користувацький інтерфейс Front-End та програмно-апаратний модуль Back-End програмного забезпечення, реалізовано взаємодію із віддаленим сервером. Розроблена програма пройшла тестування на точність та виявлення помилок. В ході розробки було реалізовано три ітерації алгоритму виявлення аномальної поведінки

1) постійна межа аномальності – без урахування специфіки часу доби;
2) східчаста межа аномальності – з додатковим урахуванням часу доби;
3) східчаста межа аномальності з додатковим урахуванням часу доби та дня тижня.

Аналіз результатів показав, що з кожною ітерацією очікувано зростає точність виявлення аномалій. У другій ітерації точність пошуку аномалій у різний часовий інтервал збільшилася більше ніж в 3 рази порівняно з першою ітерацією. У третій ітерації точність детектування аномалій зросла більше ніж в 6.5 разів порівняно з першою ітерацією, та приблизно в 1.8 разів – порівняно з другою ітерацією.

РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП – ПРОЕКТУ

Однією з основних причин створення, успішного розвитку та подальшого існування стартапів вважають неповороткість і повільність великих корпорацій, які успішно використовують уже наявні продукти, а розробкою і створенням нових майже не займаються. Тому стартапи, завдяки своїй мобільності в плані втілення нових ідей складають конкуренцію великим корпораціями.

Основним ресурсом для створення нового стартапу служить хороша новаторська ідея. Власне за свіжими і незвичайними ідеями женеться більшість і часто, купуючи їх, не шкодують великі суми грошей. Сама ідея, що не має ніякого матеріального втілення, а існує тільки на папері, або "на словах" (план стартапу), може коштувати дуже багато. Іншим фактором успішності цієї ідеї є її затребуваність (ступінь необхідності для споживача), адже ідея може бути незвичайною і новою, але користі від неї буде мінімум.

Етапи розроблення стартап-проекту:

1. Маркетинговий аналіз стартап-проекту

1. розробляється опис самої ідеї проекту та визначаються загальні напрями використання потенційного товару чи послуги, а також їх відмінність від конкурентів;
2. аналізуються ринкові можливості щодо його реалізації;
3. на базі аналізу ринкового середовища розробляється стратегія ринкового впровадження потенційного товару в межах проекту.

2. Організація стартап-проекту

1. складається календарний план-графік реалізації стартап-проекту;
2. розраховується потреба в основних засобах та нематеріальних активах;

3. визначається плановий обсяг виробництва потенційного товару, на основі чого формулюється потреба у матеріальних ресурсах та персоналі;

4. розраховуються загальні початкові витрати на запуск проекту та планові загальногосподарські витрати, необхідні для реалізації проекту.

3. Фінансово-економічний аналіз та оцінка ризиків проекту

1. визначається обсяг інвестиційних витрат;

2. розраховуються основні фінансово-економічні показники проекту (обсяг виробництва продукції, собівартість виробництва, ціна реалізації, податкове навантаження та чистий прибуток) та визначаються показники інвестиційної привабливості проекту (запас фінансової міцності, рентабельність продажів та інвестицій, період окупності проекту);

3. визначається рівень ризикованості проекту, визначаються основні ризики проекту та шляхи їх запобігання (реагування на ризики).

4. Заходи з комерціалізації проекту

1. визначення цільової групи інвесторів та опису їх ділових інтересів;

2. складання інвест-пропозиції (оферти): стислої характеристики проекту для попереднього ознайомлення інвестора із проектом;

3. планування заходів з просування оферти: визначення комунікаційних каналів та площадок та планування системи заходів з просування в межах обраних каналів;

4. планування ресурсів для реалізації заходів з просування оферти.

Означені етапи, реалізовані послідовно та вчасно – створюють передумови для успішного ринкового старту**.**

В рамках магістерської дисертації буде розглянуто перший етап розробки стартап-проекту, а саме маркетинговий аналіз стартап-проекту системи виявлення аномалій у поведінковій активності людини.

5.1. Опис ідеї проекту (технології)

Спочатку було розроблено зміст ідеї та можливі базові потенційни ринки даного проекту в межах яких слід шукати потенційних клієнтів. В табл. 5.1 наведено напрямки застосування та вигоди для користувача.

Таблиця 5.1

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Зміст ідеї: створення та реалізація алгоритму виявлення аномалій поведінки людини за допомогою методів Machine Learning;	1. Інформаційні технології	Слідкування за аномальною активністю людини
	2. Військові і технології	Експрес-оцінка стану людини; Можливість надання екстреної першої допомоги;
	3. Амбулаторний нагляд	Консультаційна підтримка; виявлення незвичної активності людини

Було визначено 4 напрямки застосування ідеї та вигоди для користувача.

Для того, щоб оцінити бізнес-ідею, необхідно зробити аналіз її потенційних техніко-економічних переваг (відмінність від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів. Для цього було обрано 4 конкуренти.

В табл. 5.2 наведено визначення сильних, слабких та нейтральних характеристик ідеї проекту.

Було визначено, що ідея має спроможність на конкуренцію та має унікальні властивості для ринку.

Таблиця 5.2.

№ п/п	Технико-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	GatorTech MicroGrid	TERVA	MavHome			
1.	Система збору даних	Детектування переміщення людини за часові інтервали використовуючи Machine Learning за допомогою датчиків	Спостереження за аварійними режимами за допомогою датчиків	Спостереження за людиною за допомогою телефону	Спостереження за людиною за допомогою датчиків без Machine Learning			+
2.	Електронна база даних користувача	Ведеться з повною історією. Є можливість її перегляду	Не ведеться	Ведеться без запису (у режими онлайн)	Не ведеться			+
3.	Відображення даних для користувача	У реальному часі, без перезавантаження	Із перезавантаженням	Із перезавантаженням	Без перезавантаження використовуючі застарілі технології			+
4.	Прогнозування аномалії	Прогнозування за допомогою Machine Learning	Не має	Не має	Не має			+

5.2. Технологічний аудит ідеї проекту

Наступним кроком було розроблено технологічний аудит ринку для визначення доступності технологій для розробки програмного забезпечення (табл. 5.3).

Таблиця 5.3

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Детектування аномалій за допомогою Machine Learning використовуючи SVM	Розробка, дослідження, програмування;	Не наявна	Доступна
2	Вивід результатів, оптимізація та розробка веб-додатку для відображення даних	Розробка, дослідження, програмування;	Не наявна	Доступна
Обрана технологія реалізації ідеї проекту: самостійна розробка на основі дослідження				

Було визначено, що технічно реалізацію продукту можливо зробити, але необхідно розробити відповідні технології для досягнення мети.

5.3. Аналіз ринкових можливостей запуску стартап-проекту

Під час аналізу було визначено можливості, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту. Це дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів (табл. 5.4).

Таблиця 5.4

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	300
2	Загальний обсяг продаж, грн/ум.од	13000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Є
6	Середня норма рентабельності в галузі (або по ринку), %	70

Отже, динаміка ринку зростає і загальний обсяг продаж великий, тому розвиток ідеї даного проекту є перспективним та доцільним.

Далі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (табл. 5.5).

Таблиця 5.5

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
Спостереження за станом MicroGrid; передвчасне попередження про проблеми; прогнозування в реальному часі	Простий користувач (для домашніх умов); Військові та рятувальні служби; Медичні клініки; Охоронні системи;	Експлуатація як у динаміці так і у статиці, в різних, як складних, так і у умовах спокою;	- до продукції: Точність; Надійність; Дешевизна; Якість; - до компанії-постачальника: Точність; Брендінг та відомість; Гарантійність

Далі в рамках аналізу було визначено цільову аудиторію та її вимоги. Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів (табл. 5.6, 5.7), що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають

Таблиця 5.6

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Перехват передачі інформації	Складність персоналізації інформації	Налагодження системи захисту інформації; розширення серверів для зберігання інформації
2	Конкуренція		
3	Зберігання інформації		

Таблиця 5.7

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Достовірність і надійність інформації	Переваги при передачі інформації	Маркетинг у цих напрямках для рекомендування себе, як компанії, на ринку;
2	Безпомилковість		

Було визначено, що загрози і можливості можливо фізично подолати. Надалі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку (табл. 5.8).

Таблиця 5.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Тип конкуренції: чиста	В кого краще - в того купують	Покращення товару та сфери обслуговування
2. За рівнем конкурентної боротьби: локальна	Належить до повсякденного ринку збуту;	Розширення функціоналу та орієнтації користувачів
3. За галузевою ознакою: міжгалузева	Притаманна різним галузям застосування;	Розширення функціоналу та галузей застосування

4. Конкуренція за видами товарів: товарно-родова та товарно-видова	Належить до аналізаторів поведінки людини	Розширення фукціоналу пристрою
5. За характером конкурентних переваг: цінова та нецінова	Чим дешевше – тим привабливіше; Чим краще – тим рентабельніше;	Покращення цінової політики та якості товару
6. За інтенсивністю: не марочна	Не жорстка конкуренція	Агресивні та не агресивні форми піару

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (табл. 5.9) - матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities).

Таблиця 5.9

Сильні сторони:	Слабкі сторони:
Навчання системи за допомогою Machine Learning – SVM; Робота програми на усіх пристроях що містять веб-браузер; Обробка великих данних (Big Data)	Не захищені дані; Необхідно багато серверів для зберігання Big Data.
Можливості:	Загрози:
Покращення безпомилковості інформації; Надійність захищеності інформації при її передачі;	Передача інформації; Зберігання інформації; Захист інформації;

5.4. Розроблення маркетингової програми стартап-проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: вибір стратегії конкурентної поведінки та опис цільових груп потенційних споживачів (табл. 5.10).

Таблиця 5.10

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	З погляду аналізатора – ні, з погляду поєднання MicroGrid і Machine Learning – так	Буде шукати нових розширюючи функціонал і потенціал продукту, а також існуючі клієнти у конкурентів самовільно будуть використовувати більш кращій продукт	Ні, не буде, так як це зменшить клієнтську базу	Помірна, місцями агресивна

Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього потрібно підсумувати результати попереднього аналізу конкурентоспроможності товару (табл. 5.11).

Таблиця 5.11

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	«Спостереження у Real-Time-Database»; «Machine Learning»; Швидка обробка даних; Прогнозування аномальних явищ;	Реалізація ідеї виявлення аномальної поведінки людини характеризується невисовою вартістю	Створення надійного бренду; Постійний розвиток та апгрейд системи та компанії у всіх напрямках; Дотримуватися схеми ціна – якість; Розширення планів щодо користування сервісом

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 5.12).

Таблиця 5.12

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	180-200% від ціни нашого продукту	180-210% від ціни нашого продукту	20000 - 400000 грн зі 100 проданих од.	6000/13000 грн

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (табл. 5.13).

Таблиця 5.13

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Задоволення потреб користувача за автоматичним та розумним виявленням аномалій у системі	Збут товару та задоволення запитуваних потреб клієнтів	Усі можливі канали збуту (глибока)	Власна
2		Збут та реклама товару та задоволення запитуваних потреб клієнтів	Усі можливі канали збуту (глибока)	Залучена

Висновки до п’ятого розділу

Згідно проведеного аналізу розроблюваний проект має можливість ринкової комерціалізації. Зростання попиту на аналогічні послуги додає масовості придбання даного програмного забезпечення, але створює жорсткі конкурентні умови виходу на ринок, де динаміка ринку доволі сприятлива до розроблюваного проекту.

Проект має високі перспективи впровадження з огляду на потенційні групи клієнтів, якими виступають медичні та експериментальні клініки, військові служби та прості рядові користувачі. Бар’єрами входження на ринок можуть бути відсутність масового виробника, сильний конкурентний тиск з боку великих фірм аналогічних продуктів. Але якщо вести агресивну боротьбу в конкурентному середовищі, проект має великі шанси та можливість зарекомендувати бренд, де в подальшому здобудеться місце на ринковій економіці. Подальша імплементація проекту є доцільною та рентабельною.

ЗАГАЛЬНІ ВИСНОВКИ

В магістерській дисертації розроблено метод виявлення аномальної поведінки людини у MicroGrid за допомогою методів машинного навчання.

1. Наявність людського фактору у MicroGrid призводить до необхідності оперування великими обсягами даних, що обгрунтовує доцільність застосування спеціалізованих методів роботи з великими даними (Big Data).

2. Серед сукупності методів машинного навчання найбільш придатним є метод детектування аномалій (Anomaly Detection), який було обрано для визначення нетипових моделей поведінки людини (аномальної поведінки) у MicroGrid. Огляд та порівняльний аналіз чотирьох методів машинного навчання показав, що найбільш доцільним серед них для вирішення поставленої задачі є метод Support Vector Machine (SVM).

3. В результаті розглядання видів аномалій поведінкової активності людини у MicroGrid розроблено структуру окремих блоків сервісу машинного навчання, досліджено особливості взаємодії з датчиками, створено алгоритм пошуку аномалій на базі методу опорних векторів SVM. Платформою для розробки програми обрано фреймворк React.js з використанням мови JavaScript, мови розмітки HTML та каскадних таблиць стилів CSS, розроблено користувацький інтерфейс Front-End та програмно-апаратний модуль Back-End програмного забезпечення, реалізовано взаємодію із віддаленим сервером.

4. Розроблена програма пройшла тестування на точність та виявлення помилок. В ході розробки було реалізовано три ітерації алгоритму виявлення аномальної поведінки:

- постійна межа аномальності – без урахування специфіки часу доби;
- східчаста межа аномальності – з додатковим урахуванням часу доби;

- східчаста межа аномальності з додатковим урахуванням часу доби та дня тижня.

5. Аналіз результатів показав, що з кожною ітерацією очікувано зростає точність виявлення аномалій. У другій ітерації точність пошуку аномалій у різний часовий інтервал збільшилася більше ніж в 3 рази порівняно з першою ітерацією. У третій ітерації точність детектування аномалій зросла більше ніж в 6.5 разів порівняно з першою ітерацією, та приблизно в 1.8 разів – порівняно з другою ітерацією.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Програмне забезпечення для макромоделювання системи керування MicroGrid / Ю. С. Ямненко, А. В. Моргун, О. М. Комаревич // Electronics and communications. - 2016. - Т. 21, № 6. - С. 61-66. - Режим доступу: http://nbuv.gov.ua/UJRN/eisv_2016_21_6_10.

2. Komarevych, O. Виявлення аномальної поведінки людини у MicroGrid на базі машинного навчання / Oleksandr Komarevych, Yuriy Khokhlov, Yuliia Yamnenko // Мікросистеми, Електроніка та Акустика. – 2018. – Т. 23, N 4. - С. 36-41. – Режим доступу : DOI : 10.20535/2523-4455.2018.23.4.143310.

3. Mozer, M.C. The neural network house: An environment hat adapts to its inhabitants. Proc. AAAI Spring Symp. Intell. Environ. 1998, 110–114.

4. Das, S.K.; Cook, D.J.; Battacharya, A.; Heierman, E.O.; Lin, T.Y. The role of prediction algorithms in the MavHome smart home architecture. IEEE Wirel. Commun. 2002, 9, 77–84.

5. Williams, G.; Doughty, K.; Bradley, D. A systems approach to achieving CarerNet-an integrated and intelligent telecare system. IEEE Trans. Inf. Technol. Biomed. 1998, 2, 1–9.

6. Korhonen, I.; Lappalainen, R.; Tuomisto, T.; Kööbi, T.; Pentikäinen, V.; Tuomisto, M.; Turjanmaa, V. TERVA: Wellness monitoring system. In Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Hong Kong, China, 1 November 1998; Volume 4, pp. 1988–1991

7. Valtonen, M.; Vuorela, T.; Kaila, L.; Vanhala, J. Capacitive indoor positioning and contact sensing for activity recognition in smart homes. J. Ambient Intell. Smart Environ. 2012, 4, 305–334.

8. Doyle, J.; Kealy, A.; Loane, J.; Walsh, L.; O'Mullane, B.; Flynn, C.; Macfarlane, A.; Bortz, B.; Knapp, R.B.; Bond, R. An integrated home-based self-

management system to support the wellbeing of older adults. J. Ambient Intell. Smart Environ. 2014, 6, 359–383.

9. Dadlani, P.; Markopoulos, P.; Sinitsyn, A.; Aarts, E. Supporting peace of mind and independent living with the Aurama awareness system. J. Ambient Intell. Smart Environ. 2011, 3, 37–50.

10. Zhu, N.; Diethe, T.; Camplani, M.; Tao, L.; Burrows, A.; Twomey, N.; Kaleshi, D.; Mirmehdi, M.; Flach, P.; Craddock, I. Bridging e-health and the internet of things: The sphere project. IEEE Intell. Syst. 2015, 30, 39–46.

11. Van Hoof, J.; Kort, H.; Rutten, P.; Duijnstee, M. Ageing-in-place with the use of ambient intelligence technology: Perspectives of older users. Int. J. Med. Inf. 2011, 80, 310–331.

12. A. Osseiran, O. Elloumi, J. Song and J. F. Monserrat, "Internet of Things," in IEEE Communications Standards Magazine, vol. 1, no. 2, pp. 84-84, 2017.

13. Demiris, G.; Hensel, B.K.; Skubic, M.; Rantz, M. Senior residents' perceived need of and preferences for "smart home" sensor technologies. Int. J. Technol. Assess. Health Care 2008, 24, 120–124.

14. Kaye, J. Home-based technologies: A new paradigm for conducting dementia prevention trials. Alzheimer Dement. 2008, 4, S60–S66.

15. Helal, S.; Mann, W.; El-Zabadani, H.; King, J.; Kaddoura, Y.; Jansen, E. The gator tech smart house: A programmable pervasive space. Computer 2005, 38, 50–60.

16. Williams, A.; Ganesan, D.; Hanson, A. Aging in place: Fall detection and localization in a distributed smart camera network. In Proceedings of the 15th ACM international conference on Multimedia, Augsburg, Germany, 25–29 September 2007; pp. 892–901.

17. Arrue, N.; Losada, M.; Zamora-Cadenas, L.; Jiménez-Irastorza, A.; Velez, I. Design of an IR-UWB indoor localization system based on a novel RTT ranging estimator. In Proceedings of the 2010 First International Conference on

Sensor Device Technologies and Applications, Venice, Italy, 18–25 July 2010; pp. 52–57.

18. Korel, B.T.; Koo, S.G. Addressing context awareness techniques in body sensor networks. In Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW ‘07), Niagara Falls, ON, Canada, 21–23 May 2007; Volume 2, pp. 798–803.

19. Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. (2000). LOF: Identifying Density-based Local Outliers (PDF). Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD. pp. 93–104. doi:10.1145/335191.335388. ISBN 1-58113-217-4.

20. Chandola V., Banerjee A., Kumar V. Anomaly detection: A Survey // ACM Computing Surveys. 2009. Vol. 41, no. 3. Article 15. DOI: 10.1145/1541880.1541882

21. Breunig M., Kriegel H.-P., T. Ng R., Sander J. LOF: Identifying Density-Based Local Outliers // Proceedings of the ACM SIGMOD International Conference on Management of Data. ACM Press. P. 93-104.

22. Vapnik, V. The nature of statistical learning theory [M] // NY. : Springer-Verlag, 1995.

23. A. Dmitruk and N. Osmolovskii, "A general lagrange multipliers theorem," 2017 Constructive Nonsmooth Analysis and Related Topics (dedicated to the memory of V.F. Demyanov) (CNSA), St. Petersburg, 2017, pp. 1-3.

24. Зангвилл У. Нелинейное программирование. Единый подход. 1969 г. Пер. с англ., под ред. Е. Г. Гольштейна, М., «Сов. радио», 1973, 312 с.

25. Gallant, S. I. (1990). Perceptron-based learning algorithms. IEEE Transactions on Neural Networks, vol. 1, no. 2, pp. 179–191.

26. J. Mercer, Functions of positive and negative type and their connection with the theory of integral equations, Philos. Trans. Roy. Soc. London 1909

27. Пискунов Н. С. Дифференциальное и интегральное исчисления для втузов, т. 2: Учебное пособие для втузов.—13-е изд.— М.: Наука, Главная редакция физико-математической литературы, 1985. — 560 с.

28. K. Karhunen, Kari, Uber lineare Methoden in der Wahrscheinlichkeitsrechnung, Ann. Acad. Sci. Fennicae. Ser. A. I. Math.-Phys., 1947, No. 37, 1-79

29. Parzen, E. (1962). "On Estimation of a Probability Density Function and Mode". The Annals of Mathematical Statistics. 33 (3): 1065–1076. doi:10.1214/aoms/1177704472. JSTOR 2237880.

ABSTRACT

Actuality of theme. The development of electronic systems, intelligent controls and information exchange has led to the formation of a new SmartGrid information and energy concept, which addresses the local electrical engineering objects - MicroGrid, which contain a certain set of sources and loads that are usually connected to a centralized grid. As part of MicroGrid, there are a number of management and device-specific tasks that result in the need to handle large information flows.

In the electric power industry, SmartGrid and MicroGrid are actively implementing Smart Grids. A significant contribution to the development of intelligent electrical networks was made by NAS academicians B.S. Stogniy, O.V. Kirilenko, Professors Denisyuk SP, Zhykov V.Ya., Klen K.S. Their works reflect the state and objectives of the development of intelligent electric networks in Ukraine. However, the issue of analysis and processing of large information flows (Big Data) in these works was not considered. Therefore, the subject of research devoted to the processing of large data in MicroGrid and the detection of anomalies through Machine Learning is relevant.

The purpose and tasks of the research. The purpose of this work is to further develop the theory and methods of machine learning in applying to the problem of finding abnormal human behavior in MicroGrid. To achieve the goal, the following tasks are solved:

1. Analysis of modern methods of detecting anomalies using Machine Learning, selection and justification of the research method.
2. Research of variants of abnormal behavioral models in MicroGrid.
3. Building an algorithm for finding anomalies in Microgrid, selecting the main parameters.
4. Development of software for finding anomalies in MicroGrid.

5. Testing, analysis of results, evaluation of the accuracy of the search for anomalies, modification of the program to the prototype

The object of the research is the processes associated with human activity in MicroGrid, which can be fixed by the appropriate sensors.

The subject of the research is to identify abnormalities of human behavior using Machine Learning methods.

Research methods. When solving the tasks set in the job, the Support Vector Machines classifier was used with an empirical rule (rule 3x sigma), through which the system was trained to search for anomalies; The development of the interface of the program and the functional was accomplished using such programming languages as JavaScript, using the React.js framework and layout using HTML, CSS.

The scientific novelty of the results obtained is as follows:

1. The theory of the use of methods of machine learning for the processing and analysis of large data (Big Data) in the part of detecting abnormalities of human behavior in MicroGrid has been further developed.

2. For the first time the solution of the problem of finding anomalies of different behavioral models based on the application of the reference vector method (SVM) is proposed.

The practical value of the results obtained:

1. The developed method allows to identify abnormalities of human behavior in MicroGrid, which is important for the prevention of emergencies or hazardous situations.

2. The developed algorithm for detecting anomalies operates with a large amount of data in the training samples and allows you to consider the subjective factors associated with human presence in MicroGrid in the development of algorithms and control systems.

3. With the help of the developed software, the user can view the time intervals of the chosen day and analyze the cases of fixing anomalies.

Personal applicant's fee. Master's work is a generalization of the results of theoretical and experimental research conducted by the author with the advice of a scientific supervisor. In works published with co-authors, the author in [1] is to search and analyze information on existing software products for modeling power supply systems MicroGrid, editing, design; [2] search and analysis of information on machine learning techniques for detecting the abnormal activity of MicroGrid power systems, editing, designing, developing the Back-End and Front-End parts of the machine learning system.

Publications. The main content of the dissertation is reflected in 2 scientific papers: 1. Ямненко Ю. С., Моргун А. В., Комаревич О. М. Програмне забезпечення для макромоделювання системи керування MicroGrid / Electronics and communications. - 2016. - Т. 21, № 6. - С. 61-66. - Режим доступу: http://nbuv.gov.ua/UJRN/eisv_2016_21_6_10. 2. Комаревич О. М., Хохлов Ю.В., Ямненко Ю.С. Виявлення аномальної поведінки людини у MicroGrid на базі машинного навчання / Мікросистеми, Електроніка та Акустика. – 2018. – Т. 23, N 4. - С. 36-41. DOI: 10.20535/2523-4455.2018.23.4.143310.

Structure and work. The master's thesis consists of an introduction, four chapters, conclusions, a list of sources used from 29 titles. The total volume of the master's thesis is 93 pages, including 87 pages of the main text, 27 figures and 4 tables.