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

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

"На правах рукопису"
УДК__________

«До захисту допущено»
Завідувач кафедри
________ О.В. Коваль
(підпис) (ініціали, прізвище)
“___”_____________2018р.

Магістерська дисертація

зі спеціальності - 122 Комп’ютерні науки та інформаційні технології
за спеціалізацією - Геометричне моделювання в інформаційних системах
на тему:
«Інтелектуальний аналіз даних в умовах розумного будинку»

Виконав: студент 6 курсу, групи ТР-71МП
Малишев Микола Сергійович ___________
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник
доцент, к.ф.-м.н., Тарнавський Ю.А. ___________
(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Рецензент
доцент кафедри атомних електричних станцій і інженерної теплофізики
к.т.н., доцент Новаківський Є.В. ___________
(посада, 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

Національний технічний університет України
"Київський політехнічний інститут ім. Ігоря Сікорського"

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий, магістерський

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології
за спеціалізацією - Геометричне моделювання в інформаційних системах

ЗАТВЕРДЖУЮ
Завідувач кафедри
Коваль О.В. ________
(прізвище, ініціали) (підпис)

«____» ___________ 2018р.

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

Малишеву Миколі Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема дисертації Інтелектуальний аналіз даних в умовах розумного будинку

Науковий керівник Тарнавський Юрій Адамович, к.ф.-м.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від "___"__________20___року №___

2. Строк подання студентом дисертації ____

3. Об'єкт дослідження: аналіз активності мешканців розумного будинку

4. Предмет дослідження: є розпізнавання активності мешканців будинку на базі бінарних сенсорів руху розумного будинку у реальному часі

5. Перелік питань, які потрібно розробити ____

– проаналізувати існуючі дослідження по темі розпізнавання активностей в розумному будинку;

– на базі існуючих досліджень виділити ті які фокусуються на аналізі даних з сенсорів розумного будинку у реальному часі;

– побудувати модель розпізнавання активності на базі даних потоку сенсорів у реальному часі яка б покращувала результати існуючих досліджень;

– розробити програмну систему, яка використовуючи розроблену модель може здійснювати візуалізації результатів розпізнавання і аналізу активностей на аномалії.

6. Орієнтований перелік ілюстративного матеріалу презентація на тему «Інтелектуальний аналіз даних в умовах розумного будинку»

7. Орієнтований перелік публікацій

1. Малишев М. С. ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ В УМОВАХ "РОЗУМНОГО БУДИНКУ" / М. С. Малишев, Ю. А. Тарнавський // СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ / М. С. Малишев, Ю. А. Тарнавський., 2018. – С. 195.
2. Тарнавський Ю. А. Інтелектуальний аналіз даних в умовах розумного будинку / Ю. А. Тарнавський, М. С. Малишев // СТАЛИЙ РОЗВИТОК — XXI СТОЛІТТЯ: УПРАВЛІННЯ, ТЕХНОЛОГІЇ, МОДЕЛІ / Ю. А. Тарнавський, М. С. Малишев., 2018. – С. 448–450.

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Основна частина	Тарнавський Ю.А., доцент., к.ф.-м.н.		

9. Дата видачі завдання «____»_______________2018 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строки виконання етапів магістерської дисертації	Примітка
1	Отримання завдання	30.09.17р.	
2	Аналіз вимог завдання, вибір методів і засобів розв’язання поставленої задачі	01.10.17р. — 31.01.18р.	
3	Підготовка матеріалів магістерської роботи	02.09.17р. — 11.05.18р.	
4	Підготовка публікацій	05.02.17р. — 31.05.18р.	
5	Доповідь на конференції	27.04.18р.	
6	Проходження переддипломної практики	1.09.18р. —21.10.18р	
6	Захист програмного продукту	25.10.18р	
7	Розроблення стартап-проекту	14.11.18 р. —25.11.18р.	
8	Передзахист	29.11.18р	
9	Захист	17.12.18р	

Студент ____________ Малишев М.С.
(підпис) (прізвище та ініціали)

Науковий керівник ____________ Тарнавський Ю.А.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

на магістерську дисертацію

виконану на тему: «Інтелектуальний аналіз даних в умовах розумного будинку»

студентом: Малишевим Миколою Сергійовичем

Магістерська дисертація складається зі вступу, п’яти розділів, висновку, переліку посилань з 54 найменувань, 3 додатків, містить 38 рисунків і 21 таблицю. Повний обсяг магістерської дисертації складає 98 сторінок, з яких перелік посилань займає 6 сторінок, додатки – 13 сторінок.

Актуальність теми. На сьогоднішній розумні будинки мають певну кількість обмежень серед яких неможливість аналізувати активність мешканців. На даний момент відсутні рішення які б були в змозі вирішити дану проблему. Одним з підходів є аналіз бінарних сенсорів руху, перевагою такого аналізу є дешевизна самих сенсорів і достатня конфіденційність даних мешканців. Для принесення користі система аналізу має функціонувати у реальному часі, однак аналіз існуючих досліджень показав що лише невелика кількість з них фокусується на аналізі у реальному часі, крім того їх точність не є достатньою для використання в реальних умовах. Саме тому необхідно розробити систему розпізнавання активності у реальному часі яка б покращувала результати існуючих досліджень.

Метою дослідження є розробка програмного забезпечення інтелектуального аналізу даних в умовах розумного будинку, яке має можливості щодо розпізнавання активності у реальному часі з можливістю подальшого використання результатів.

Для досягнення мети необхідно виконати наступні завдання:

- проаналізувати існуючі підходи розпізнавання активностей у реальному часі мешканців розумного будинку;
- удосконалити метод розпізнавання активності у реальному часі;
- розробити програмну систему, яка використовуючи розроблену модель може здійснювати візуалізацію результатів розпізнавання і аналізу активностей на аномалії.

Об'єктом дослідження комп'ютерні інформаційні системи і технології розумного будинка.

Предметом дослідження є системи інтелектуального аналізу даних в умовах розумного будинку.

Методи дослідження. При розв'язку даної задачі було використано наступні методи:

- здійснення сегментації потоку даних сенсорів;
- побудова моделей машинного навчання;

Наукова новизна одержаних результатів. Найбільш суттєвими науковими результатами магістерської дисертації є:

- удосконалення методу розпізнавання активності на базі даних бінарних сенсорів руху у реальному часі за рахунок використання моделі випадкового лісу в якості класифікатора і моделі Латентного Розміщення Діріхле для включення в сегмент інформацію про розподіл активностей, що призвело до зростання точності класифікації;
- розробка програмної системи яка використовуючи розроблену модель здійснює візуалізацію результатів розпізнавання і аналіз на аномальні активності;

Практичне значення одержаних результатів полягає в можливості використання програмної системи для автоматизації розумних будинків а також у сфері охорони здоров'я для моніторингу активності людей з вадами здоров'я.

Ключові слова. *РОЗУМНИЙ БУДИНОК, АНАЛІЗ ДАНИХ, РОЗПІЗНАВАННЯ АКТИВНОСТІ, СЕГМЕНТАЦІЯ, МАШИННЕ НАВЧАННЯ, СЕНСОРИ РУХУ*

ABSTRACT

on master's thesis

on topic: « Intelligent data analysis in a smart home»

Student: Mykola Malyshev

Master's thesis consists of an introduction, five sections, a conclusion, a list of references from 54 denominations, 3 appendices, and have 38 figures, 21 tables. The full volume of the master's thesis is 98 pages, 6 of which is list of references, 13 – appendices.

Topicality. Today's smart houses have a number of limitations, including the inability to analyze the activity of residents. At the moment, there are no solutions that would be able to solve this problem. One approach is the analysis of binary motion sensors, the advantage of such an analysis is the cheapness of the sensors themselves and the sufficient confidentiality of the data of the inhabitants. The analysis system needs to be operational in real time, but an analysis of existing research has shown that only a small number of them are focused on real-time analysis, in addition, their accuracy is not sufficient for use in real-world conditions. That is why it is necessary to develop a system of real-time activity recognition that would improve the results of existing research.

The aim of the research is the development of intelligent data analysis software in a smart home that has the ability to recognize real-time activity with the ability to further use the results.

To achieve the goal, following tasks must be completed:

- to analyze approaches of recognition of real-time activities of residents of a smart home;
- Improve the real-time activity recognition method;
- develop a software system, which by using the created model can perform visualization of the results of recognition and analysis of activities on anomalies.

The object of research computer information systems and technologies of smart home.

The subject of research systems of intelligent data analysis in a smart home.

Research Methods. The solving of defined tasks was performed using the following methods:

- Segmentation of sensor data flow;
- Construction of models of machine learning;

Scientific novelty. The most significant scientific results of the master's thesis are:

– Improvement of the method of recognition of activity on the basis of binary real-time motion sensors through the use of the model of random forest as a classifier and the model of Latent Dirichlet Allocation for inclusion in the segment information about distribution of activities, which led to an increase in the accuracy of the classification;

– a software system was developed that, using the developed model, visualizes the recognition results and analyzes the abnormal activity;

The practical value of research is the possibility of using a software system for the automation of smart homes as well as in health care to monitor the activity of people with disabilities.

Keywords. *SMART HOME, DATA ANALYSIS, ACTIVITY RECOGNITION, SEGMENTATION, MACHINE LEARNING, MOTION SENSORS*

ЗМІСТ

ВСТУП

На сьогоднішній день можна спостерігати посилений інтерес до ринку розумних будинків зі сторони великих компаній, таких як Google, Samsung, Amazon та інших. Передбачається що ринок розумних будинків перевищить 40 мільярдів доларів до 2020 року [1]. Однак, незважаючи на ріст ринку, розумні будинки мають певну кількість обмежень серед яких неможливість аналізувати поведінку мешканців будинку.

Важливою характеристикою розумних будинків є те, що наявні в ньому пристрої мають можливість зберігати та обмінюватися інформацією. Саме аналіз цієї інформації надасть змогу проаналізувати активність мешканців будинку [2].

Об’єктом аналізу можуть слугувати дані з різноманітних джерел, таких як бінарні сенсори руху так і відеозаписи з камер. В даній роботі аналізуються дані бінарних сенсорів руху. Це обумовлено невисокою вартістю самих сенсорів і вартістю зберігання даних з цех сенсорів, так як вони зберігаються у текстових файлах. Крім того інформація з цих сенсорів не є настільки конфіденційною для потенційних мешканців на відміну від тих же відеозаписів з камер.

Загалом методики аналізу активності на сьогоднішній день можна поділити на 2 групи: ті, що базуються на знаннях і ті, що базуються на даних.

Підходи, які базуються на знаннях використовують евристики та попередні знання про предметну область для моделювання активностей. Такі моделі не потребують великих наборів даних, однак в свою чергу є дещо статичними і обмеженими до конкретних заданих характеристик того чи іншого розумного будинку [3].

В свою чергу в підходах які базуються на даних використовуються моделі машинного навчання: нейронні мережі, метод опорних векторів і інші. Вони потребують набір даних для тренування однак в свою чергу можуть бути використані у декількох різних за характеристиками розумних будинках.

Ці підходи загалом можна поділити на ті, які використовують навчання з вчителем та навчання без вчителя. У підходах навчання без вчителя використовуються різноманітні методи кластеризації. Як правило виходом цих моделей є шаблони які наявні в даних або існуючі в даних аномалії [4].

Так у підходах які використовують навчання з вчителем використовують анотовані активності. Тобто виходом таких підходів є множина класифікованих активностей. Існують як прості підходи, які використовують лише одну модель класифікації так і різноманітні гібридні підходи, наприклад комбінуючи моделі класифікації і кластеризації, передаючи результат кластеризації на вхід моделям класифікації [5-10].

Для принесення користі система аналізу сенсорів має мати можливість функціонування у реальному часі, але при результаті аналізу існуючих досліджень, було з'ясовано, що серед існуючих запропонованих підходах, лише невелика кількість з них фокусується на аналізі даних в реальному часі. Крім того їх точність не є достатньою для можливості використання в реальних умовах [11-15]. Також наразі не представлено програмних систем, які надавали б можливості користувачу щодо здійснення класифікації даних і подальшого використання результатів класифікації.

Розроблена програмна система має декілька можливостей щодо використання. Система може бути використана для моніторингу активностей людей з певними вадами і людей похилого віку. В такому випадку медичні або соціальні працівники мають можливість щодо спостереження за активностями мешканців будинку і своєчасного реагування при необхідності. Для таких користувачів є важливою можливість автоматичного виділення аномальних активностей [16].

Також розроблена система покращить можливості щодо автоматизації. Враховуючи, що система може функціонувати у реальному часі, потенційні розробники систем автоматизації розумного будинку зможуть розробити більш персоналізовані середовища існування завдяки класифікованим активностям [17]. Так, наприклад, стане можливим створення правила у вигляді: виключити світло у будинку у випадку активності сну.

Саме тому метою даної роботи є розробка програмного забезпечення інтелектуального аналізу даних в умовах розумного будинку, яке має можливості щодо розпізнавання активності у реальному часі з можливістю подальшого використання результатів.

Для досягнення мети необхідно виконати наступні завдання:

- проаналізувати існуючі підходи розпізнавання активностей у реальному часі мешканців розумного будинку;
- удосконалити метод розпізнавання активності у реальному часі;
- розробити програмну систему, яка використовуючи розроблену модель може здійснювати візуалізацію результатів розпізнавання і аналізу активностей на аномалії.

1. ЗАДАЧА ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ В УМОВАХ РОЗУМНОГО БУДИНКУ

Для можливості розробки моделі класифікації активності у реальному часі необхідно розглянути існуючі дослідження і представлені в них результати. Важливим аспектом є також вибір набору даних для здійснення тренування моделі, так як її валідація буде здійснена на тому ж наборі даних. Дані мають бути максимально схожими на ті, які будуть поступати в умовах застосування розробленої програмної системи. Так був вибраний проект Вашингтонського університету центру досліджень адаптивних систем по причині наявності великої кількості наборів даних, які найбільше всього підходять для тренування та валідації моделей класифікації.

1.1 Проект WSU CASAS

Проект Вашингтонського університету центру досліджень адаптивних систем CASAS (Center for Advanced Studies in Adaptive Systems at Washington State University) був створений для вивчення проблеми аналізу активності людини, виділення шаблонів поведінки в рамках розумного будинку [18].

Дані були зібрані протягом певного часу під час життя мешканців будинку. Певна частина представлених наборів даних розмічена активностями, тобто у відповідність активаціям сенсорів у наборі поставлена та чи інша активність. Активності розмічені вручну організаторами дослідження. Так набори даних відрізняються між собою: наявністю розмітки, часом проведення дослідження, планом будинків і розташуванням сенсорів в них, кількістю сенсорів і анотованих активностей а також кількістю мешканців. Усі набори даних доступні у відкритому доступі [19].

На фізичному рівні CASAS містить апаратні компоненти, включаючи сенсори та актуатори. Архітектура використовує мережу на базі Zigbee, яка безпосередньо зв'язується з встановленими пристроями. Рівень проміжного програмного середовища регулюється менеджером. Менеджер надає канали, які дозволяють компонентним мостам відправляти та отримувати повідомлення. Крім того, менеджер додає часові мітки до виникаючих подій і присвоює UUID. Кожен компонент архітектури CASAS обмінюється повідомленнями з цим менеджером через спеціальний міст XMPP [20].

Як правило інформація у наборах даних являє собою послідовно зібрані показники сенсорів. Кожен рядок містить інформацію про час активації сенсору, його унікальний ідентифікатор і значення а також відповідну анотовану активність (Рисунок 1.1).

2010-11-04	05:40:51.303739	M004	ON	Bed_to_Toilet	begin
2010-11-04	05:40:52.342105	M005	OFF		
2010-11-04	05:40:57.176409	M007	OFF		
2010-11-04	05:40:57.941486	M004	OFF		
2010-11-04	05:43:24.021475	M004	ON		
2010-11-04	05:43:26.273181	M004	OFF		
2010-11-04	05:43:26.345503	M007	ON		
2010-11-04	05:43:26.793102	M004	ON		
2010-11-04	05:43:27.195347	M007	OFF		
2010-11-04	05:43:27.787437	M007	ON		
2010-11-04	05:43:29.711796	M005	ON		
2010-11-04	05:43:30.279021	M004	OFF	Bed_to_Toilet	end
2010-11-04	05:43:45.7324	M003	ON	Sleeping	begin
2010-11-04	05:43:52.044085	M003	OFF		
2010-11-04	05:43:53.185335	M002	ON		
2010-11-04	05:43:53.253809	M003	ON		
2010-11-04	05:43:59.493281	M002	OFF		
2010-11-04	05:44:04.048766	M003	OFF		
2010-11-04	05:44:06.14204	M003	ON		
2010-11-04	05:44:11.229146	M003	OFF		

Рисунок 1.1. Приклад вхідних даних

Кожен набір даних також містить план розташування цих сенсорів у будинку. На рисунку 1.2 можна бачити план для будинку який містить сенсори руху(починаються з літери M), сенсори розміщені на двері(починаються з літери D) а також датчики температури (містять літеру T).

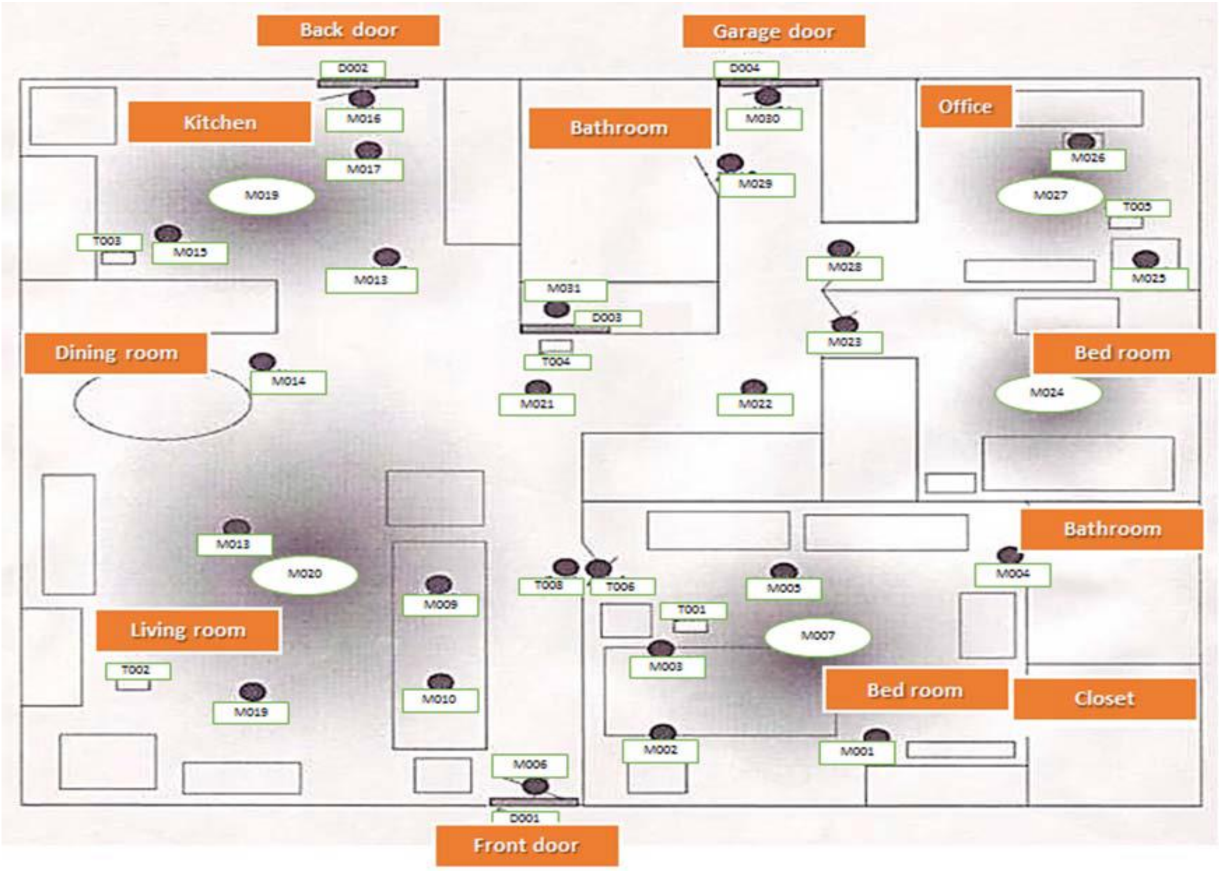


Рисунок 1.2. Розташування сенсорів в розумному будинку

Інформація з датчиків температур не несе корисної інформація для задачі розпізнавання активностей саме тому, як правило, видаляється з набору даних при здійсненні моделювання.

1.2 Сегментація даних і її роль для аналізу в реальному часі

Однією з вимог даної роботи є можливість працювати у реальному часі. Для можливості здійснення такої класифікації необхідно поділити потік вхідних даних з сенсорів на сегменти, так як надалі розроблена модель буде аналізувати сенсори саме в їх рамках. Тобто навчальна вибірка буде сформована з сегментів у відповідність кожному з яких буде поставлена певна активність.

Можна виділити такі види сегментації: за активностями, за часом і за сенсорами [21]. Види сегментації представлені на рисунку 1.3.

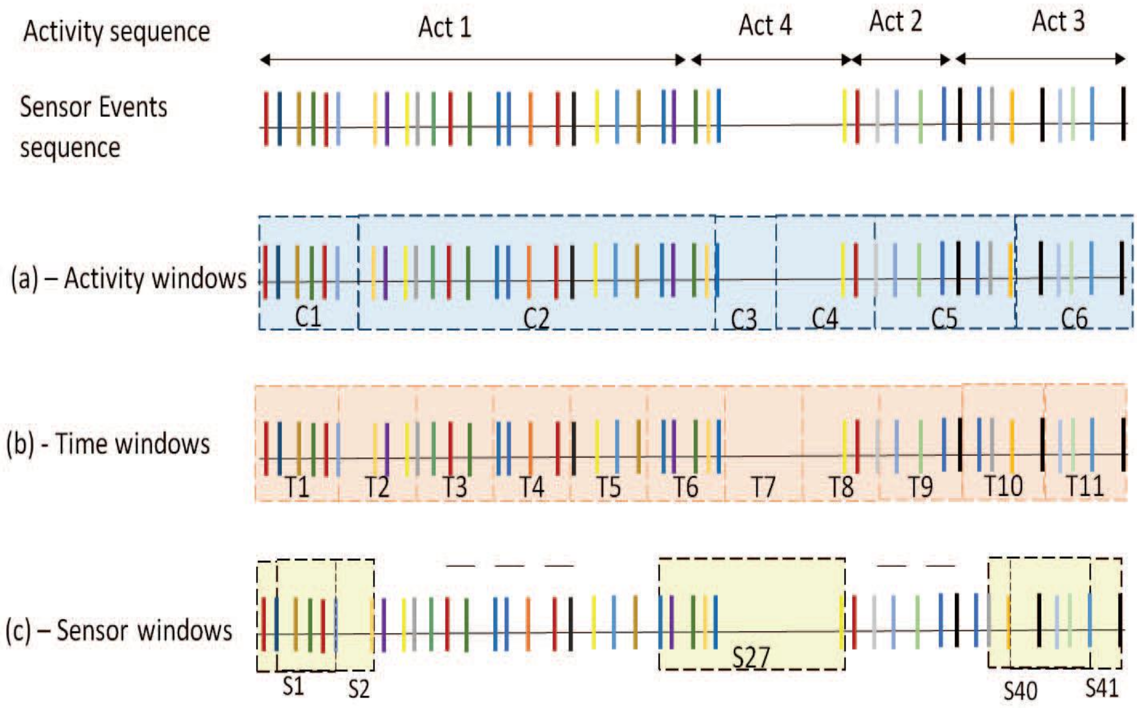


Рисунок 1.3. Види сегментації даних потоку сенсорів

При сегментації за активностями вхідні сенсори спочатку діляться на вікна різної довжини і передбачається що кожне вікно містить певну активність. Надалі таке вікно подається на вхід до прогнозуючої моделі класифікації. Проблемами такого підходу є те, що кожне вікно може містити декілька активностей і при функціонуванні у реальному часі такий спосіб має очікувати певну кількість активацій сенсорів, для того, щоб здійснити класифікацію.

При сегментації за часом потік сенсорів ділиться на вікна однакової часової довжини. Проблемою такого підходу є те, що в залежності від вибору довжини вікна, кожне вікно може як не мати в собі достатньої інформації для класифікації активності, так і мати в собі інформацію що відповідає декільком активностям.

При сегментації за сенсорами, потік сенсорів ділиться на вікна однакової кількості активацій сенсорів. В даному випадку передбачається класифікація саме останнього сенсору в вікні, минулі сенсори в даному випадку визначають контекст цього вікна. Даний метод теж має певні недоліки: вікно може містити в собі активації сенсорів між якими великі проміжки часу. Крім цього при наявності в будинку декількох мешканців, кожне вікно може містити інформацію про декілька паралельних активностей. Цей метод був вибраний по причині того, що для здійснення класифікації непотрібно очікувати певну кількість сенсорів, тобто класифікація здійснюється в момент надходження кожного сенсору. Кількість активацій сенсорів в сегменті в даному випадку залежить від обраної ширини вікна.

Так як система має класифікувати кожну активацію сенсору на конкретну активність у реальному часі – сегментація дозволяє використати певну кількість минулої інформації для здійснення класифікації останнього сенсору. Більш детально цей процес зображений на рисунку 1.4.

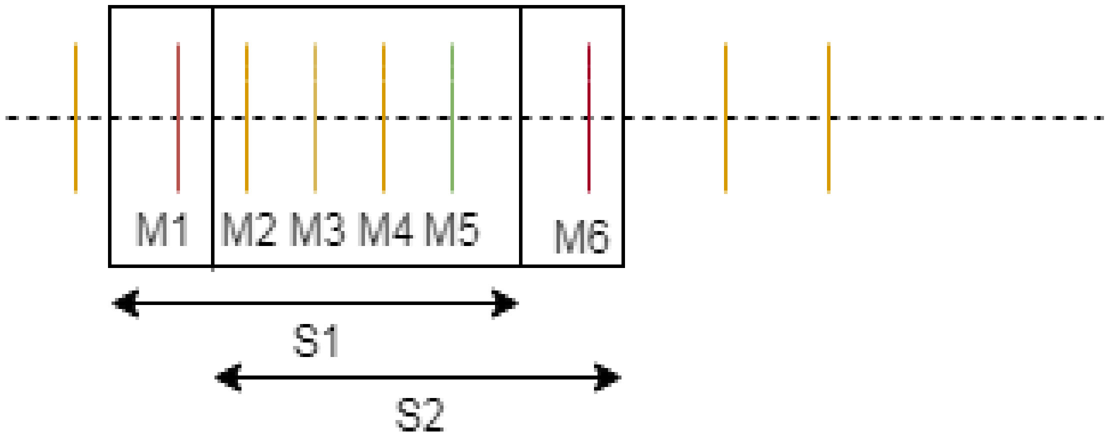


Рисунок 1.4. Сегментація за сенсорами

Нехай ми маємо на вході активацію сенсора M5 і маємо довжину вікна 5. На базі цього формується сегмент S1 до якого входять активації M1, M2, M3, M4. На наступному кроці коли на вхід потрапляє сенсор M6 – відповідно формується сегмент S2 до якого входять активації M2, M3, M4, M5. Надалі з сформованих сегментів формується вектор ознак.

1.3 Основні підходи до аналізу даних в реальному часі

В рамках роботи було відібрано ряд публікацій, які фокусуються саме на аналізі у реальному часі. Надалі точність розробленої моделі буде порівнюватися саме з тими результатами які представлені у цих дослідженнях.

В одному з досліджень було проаналізовано ряд підходів щодо формування вектору ознак [11]. Базовий підхід включає в кожний сегмент інформацію про час активацій першого та останнього сенсору у вікні, тривалість та кількість активацій кожного з сенсорів. Інші підходи є модифікаціями базового. Так одним з підходів є зважування частот активацій сенсорів в залежності від часової відстані до останньої активації сенсору. Значення ваг визначається наступним чином:

$$C(i,j) = \exp(-\chi\,(t_i - t_k)) \tag{1.1}$$

де значення χ визначає швидкість занепаду впливу а t_i та t_k відповідно час останнього та к-ї активації сенсору вікна.

Інший підхід пропонує включити в сегмент інформацію про минуле вікно. Так тренуються дві моделі. Перша модель тренується на сегментах які не включають інформацію про минуле вікно, друга модель використовує результат класифікації першої моделі як нову ознаку. Принцип функціонування такого підходу зображений на рисунку 1.5.

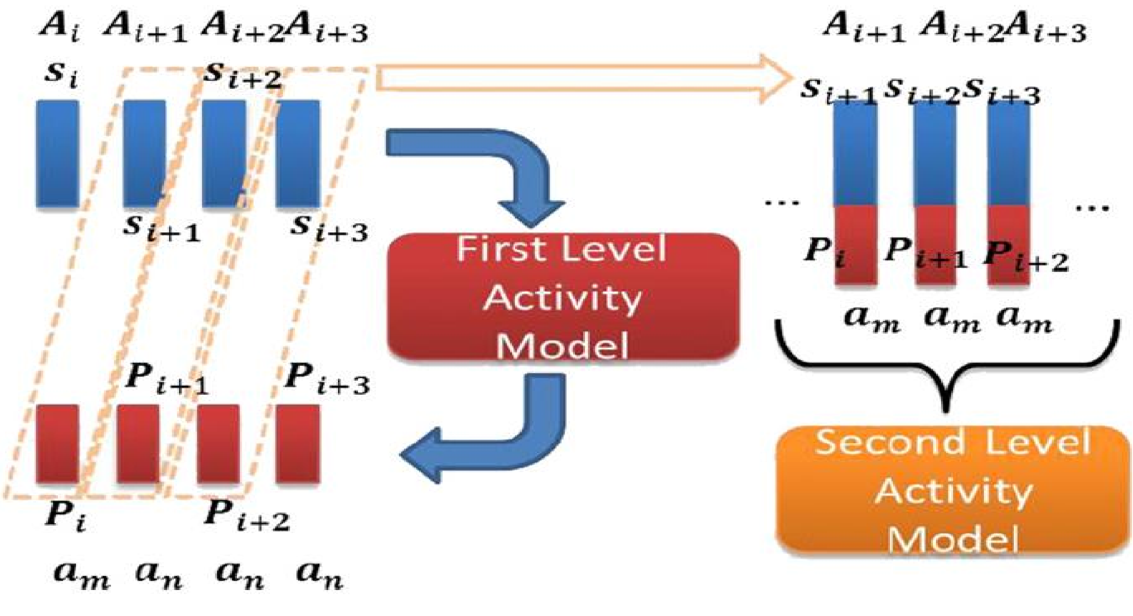


Рисунок 1.5 використання двох моделей

Раніше було зазначено, що сегментація за сенсорами має певні недоліки, а саме можливість потрапити в один сегмент активаціям сенсорів з різних активностей. Для вирішення цієї проблеми автори запропонували використати матрицю взаємної інформації для зважування частот активацій сенсорів.

Матриця визначається наступним чином:

$$M(i,j) = \frac{1}{N}\sum_{k=1}^{N-1} \delta\,(s_k, S_i)(s_{k+1}, S_j) \qquad (1.2)$$

де $\delta = 1$ якщо даний сенсор S_i а наступний S_j. Тобто вираховується імовірність активації двох сенсорів послідовно на потоці даних сенсорів.

Також було запропоновано здійснити сегментацію за часом з різними довжинами вікон, від п’яти секунд до хвилини. Крім того запропоновано метод динамічного вибору довжини вікна. Метод базується на статистичній інформації представленій в даних. Так розраховується найбільш оптимальна довжина вікна з врахуванням інформації про довжини вікон активностей і розподілу сенсорів по цим активностям.

В якості методу класифікації було вибрано метод опорних векторів. Автори публікації дійшли до висновку що найбільша точність була отримана використовуючи комбінацію підходів з матрицею взаємної інформації і інформації про минулу активність.

В іншому дослідженні на відміну від минулої праці запропоновано 2 нових підходи: з додаванням до вектору ознак ідентифікатора останнього сенсора вікна і використання матриці взаємної інформації, яка враховує не імовірність послідовної активації сенсорів, а імовірність активації двох сенсорів в рамках однієї активності [12]:

$$MI(i,j) = \frac{1}{|Q|}\sum_{k=1}^{|Q|} \partial(i,j) \qquad (1.3)$$

де $\partial(i,j)$ має значення 1 в випадку якщо сенсори i та j присутні в рамках однієї активності та 0 якщо ні. Q дорівнює кількості сегментованих активностей. Надалі інформація у матриці використовується для зважування частот у векторі ознак, так же як і в формулі 1.2. Також автори реалізували деякі методи запропоновані в минулій публікації.

В рамках цієї роботи автори також використали метод опорних векторів. Враховуючи, що автори використали інший набір даних ніж той який був представлений у минулій публікації, вони дійшли до висновку, що представлені підходи не принесли значного покращення точності класифікації. Проблема полягає в тому що набори даних CASAS містять лише частину анотованих активностей, так як на потоці даних сенсорів не всі активації сенсорів відповідають конкретній активності. Не анотовані активації можуть відноситись наприклад до переміщення мешканця по будинку, або іншій активності, яка не була розмічена під час проведення дослідження. Як правило не розмічені активації сенсорів позначаються окремим класом, сегментуються і подаються на вхід до класифікатора так само як і інші класи.

Проблема в даному випадку полягає в незбалансованості навчальної вибірки. Так на рисунку 1.6 розподіл активацій сенсорів за відповідними їм класами в одному з наборів даних CASAS.

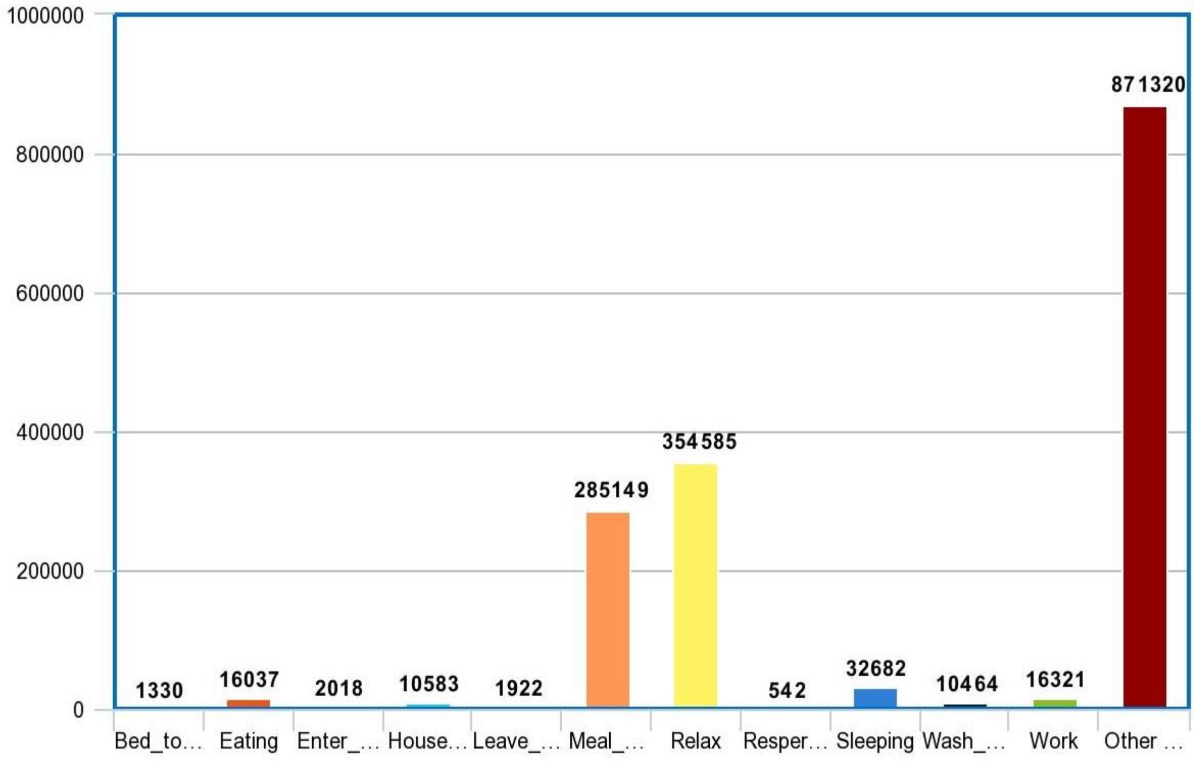


Рисунок 1.6. Розподіл активацій сенсорів за класами

Така незбалансованість призводить до того, що збільшується імовірність того що класи будуть класифіковані як інша, не розмічена активність. Так на рисунку 1.7 зображена матриця помилок дослідження [11].

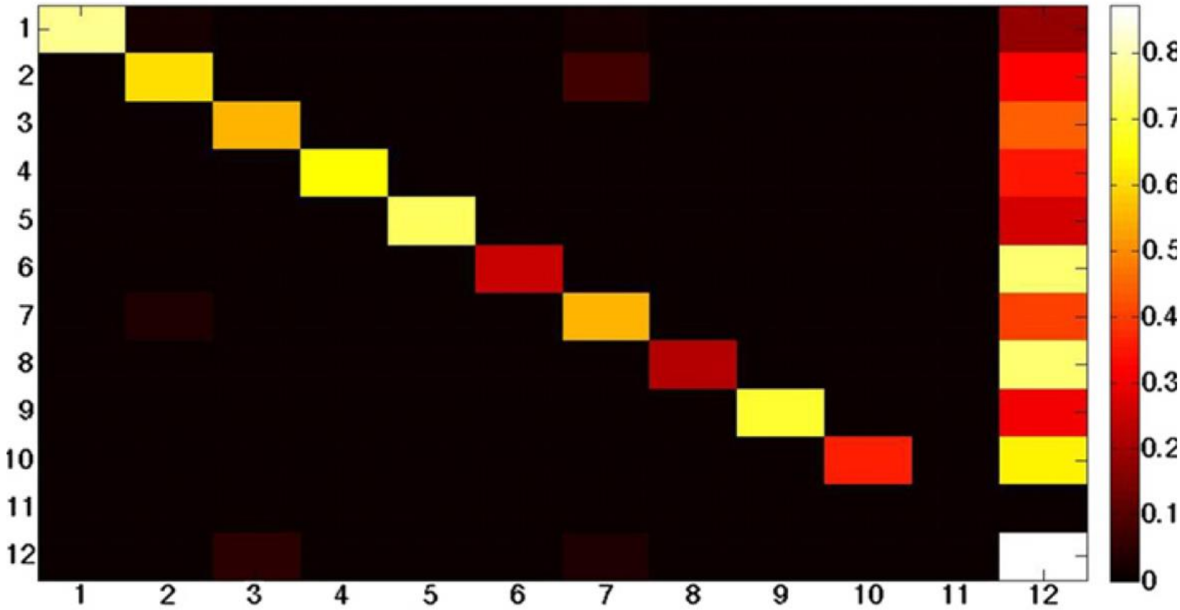


Рисунок 1.7. Матриця помилок дослідження

Матриця помилок містить по вертикалі правильні мітки класів, а по горизонталі – класифіковані. Як можна бачити дійсно великий процент різних класів класифіковано як іншу активність.

Далі ці ж автори у наступному дослідженні вирішили на базі раніше представлених методів формування вектору ознак використати інші методи для здійснення класифікації [13]. Так авторами було запропоновано використати метод к-найближчих сусідів і декілька його модифікацій. Метод полягає у присвоєнні класу вхідному вектору на базі зважування класів до яких належать к найближчих векторів. Даний метод не потребує навчання, так як при класифікації лише здійснюється пошук серед елементів навчальної вибірки. Зокрема в дослідженні були використані наступні модифікації:

– Використання векторів усіх класів для вирішення проблеми незбалансованості навчальної вибірки [22];
– метод на базі гаусівського шара, який також націлений на проблему незбалансованості і включає в себе фазу тренування [23];
– модифікація до минулого методу авторів даного дослідження, для проблеми аналізу активності.

В рамках дослідження на запропонованих методах автори не досягли зростання точності класифікації у порівнянні з минулим дослідженням.

Наступне дослідження пропонує метод включення станів сенсорів до вектору ознак та включення декількох останніх ідентифікаторів сенсорів. Автори використали метод опорних векторів. Результатом є покращення точності класифікації на 5% [14].

У іншому дослідженні будується матриця найбільш репрезентативних сенсорів для кожної активності [15]. Так створюється ряд статистичних ознак таких як: середнє, медіана, стандартне відхилення і інші. Після їх вибору для кожної активності ітеративно розраховується корисність сенсорів на базі приросту інформації [24]. В якості методу класифікації був вибраний метод опорних векторів, який був тренований окремо для кожного класу.

Надалі представимо найкращі результати досліджень і їх характеристики у таблицях. Для оцінки результатів досліджень було використано декілька метрик: точність за класами та метрику F1. Вони є найбільш доцільними в даному випадку із-за проблеми багато класової класифікації і незбалансованості даних. Точність за класами визначається наступним чином [25]:

$$\text{Загальна Точність} = \sum_{j=1}^{A} \frac{TPa_j}{Na_j} \qquad (1.4)$$

де A – кількість класів, TPa_j – кількість вірно класифікованих сегментів для класу a_j, а Na_j – загальна кількість сегментів цього класу.

Іншою використовуваною метрикою є метрика F1. Вона враховує точність та повноту результуючого класу. Точність – частина правильно класифікованих екземплярів у класі, повнота – частина правильних класифікацій відносно усіх наявних в наборі даних екземплярів. Нехай точність класу a_j визначена як Pa_j а його – повнота Ra_j тоді метрика F1 для даної задачі визначається наступним чином:

$$F1\ \text{метрика} = \frac{2}{A} \sum_{j=1}^{A} \frac{Pa_j * Ra_j}{(Pa_j + Ra_j)} \qquad (1.5)$$

На базі двох метрик приведемо результати точності класифікацій досліджень без іншого класу у таблиці 1.1.

Таблиця 1.1. Представлені результати досліджень без іншого класу

Дослідження	[12]	[13]	[15]
Точність	87.55%	86.76%	85%
Метрика F1	69.24%	62.23%	82%

Представлено 3 публікації так як не всі тестували точність не враховуючи інший клас. У таблиці 1.2 представлено результати класифікаторів враховуючи класифікацію іншого класу. Дослідження [15] не включено в таблицю по причині того що дослідження не враховувало клас інший.

Таблиця 1.2 Представлені результати досліджень з іншим класом

Дослідження	[11]	[12]	[13]	[14]
Точність	61%	63%	60%	59%
Метрика F1	48%	47%	47%	58%

Як можна бачити порівнюючи обидві таблиці – класифікація іншої активності значно понижує точність представлених класифікаторів.

Висновки до розділу 1

В рамках першого розділу було:

1. Описано та проаналізовано особливості проекту CASAS. Набір даних був зібраний у реальних умовах і являє собою потік даних сенсорів, тобто у тому вигляді у якому будуть поступати дані в реальних умовах, саме тому може бути використаним для тренування і валідації моделей.
2. Описані методи сегментації потоку даних сенсорів, найбільш доцільним для поставленої задачі є метод сегментації за сенсорами завдяки

можливості здійснювати класифікацію без очікування даних з наступних сенсорів.

3. Описані методології застосовані в різних дослідженнях. Як правило майже в усіх дослідженнях використовували метод опорних векторів з формуванням різних векторів ознак. Саме тому необхідно застосувати інші моделі машинного навчання на різних векторах ознак.
4. Отримано висновок що при включенні іншого класу до класифікації, точність падає до значення, яке не є достатнім для використання представлених в публікаціях підходів у реальних умовах.

2. МОДЕЛЬ КЛАСИФІКАЦІЇ ДАНИХ ПОТОКУ СЕНСОРІВ РОЗУМНОГО БУДИНКУ У РЕАЛЬНОМУ ЧАСІ

Враховуючи особливості підходів представлених у минулому розділі необхідно розробити власний підхід щодо здійснення класифікації. В процесі вибору моделі машинного навчання для здійснення класифікації важливими факторами є як сама модель так і сформований для неї вектор ознак. Враховуючи що не існує моделей машинного навчання які б були здатні вирішити всі задачі – варто побудувати декілька та порівняти їх точність [26].

Важливо проаналізувати точність результатів моделювання та порівняти їх з представленими раніше дослідженнях. В результаті аналізу особливостей набору даних було зрозуміло, що в даному випадку має місце незбалансованість у даних яка стає ще більш вираженою в умовах включення класу інший до набору даних. Крім того дані були розмічені вручну авторами досліджень що свідчить про можливі неточності в даних. Вибрані моделі мають бути стійкими до зазначених особливостей.

2.1 Модель випадкового лісу

Випадковий ліс є ансамблевим класифікатором який складається з певної кількості дерев рішень, кожне з яких тренується на підвибірці з даних. Слід зазначити, що дані які поступають на вхід до кожного з дерев можуть повторюватися. Самі дерева будуються за алгоритмом CART. Даний алгоритм будує бінарне дерево і при побудові на кожному кроці правило в вузлі ділить множину даних на дві частини – ті які відповідають і не відповідають правилу сформованому в вузлі. Дане правило формується при спробі поділити набір даних на групи на базі значення певної ознаки [27].

При побудові використовується критерій невизначеності Джині. Якщо набір даних А містить дані n класів, тоді індекс Джині визначається наступним чином

$$G(\mathrm{A}) = 1 - \sum_{k=1}^{n} p_k^2 \qquad (2.1)$$

де p_k відносна частота класу k в А. Найкращим відповідно вважається таке розбиття яке мінімізує даний критерій.

Після навчання дерев, класифікація здійснюється шляхом голосування кожного дерева і результативний клас є тим класом, за який проголосувало більшість дерев рішень. Кількість дерев рішень як правило вибирається таким, яке надає найкращий результат на даних валідації (Рисунок 2.1).

Рисунок 2.1 Загальний вигляд алгоритму випадкового лісу

Перевагами цього методу є можливість роботи з великою кількістю ознак при чому ці ознаки можуть буть різних типів, можливість паралельного навчання, що в свою чергу зменшує час необхідний на тренування. Крім того необхідно зазначити що дана модель може гарно справлятися з неточними даними [28]. Це є важливою характеристикою, так як існуючі набори даних для даної задачі, як правило, містять в собі неточності.

2.2 Модель нейронної мережі прямого поширення

На сьогоднішній день ріст використання моделей глибинного навчання обумовлений тим фактом що створення більш комплексних нейронних мереж надає змогу описати більш комплексні залежності, які існують в даних. Так наразі завдяки моделям цього типу було вирішено ряд проблем, які раніше були невирішуваними [29].

Загальна модель моделі нейронної мережі прямого поширення зображена на рисунку 2.2. Тобто типова нейронна мережа складається з вхідного слою, певної кількості скритих слоїв та вихідного слою. Кожен шар має свою кількість комірок [30].

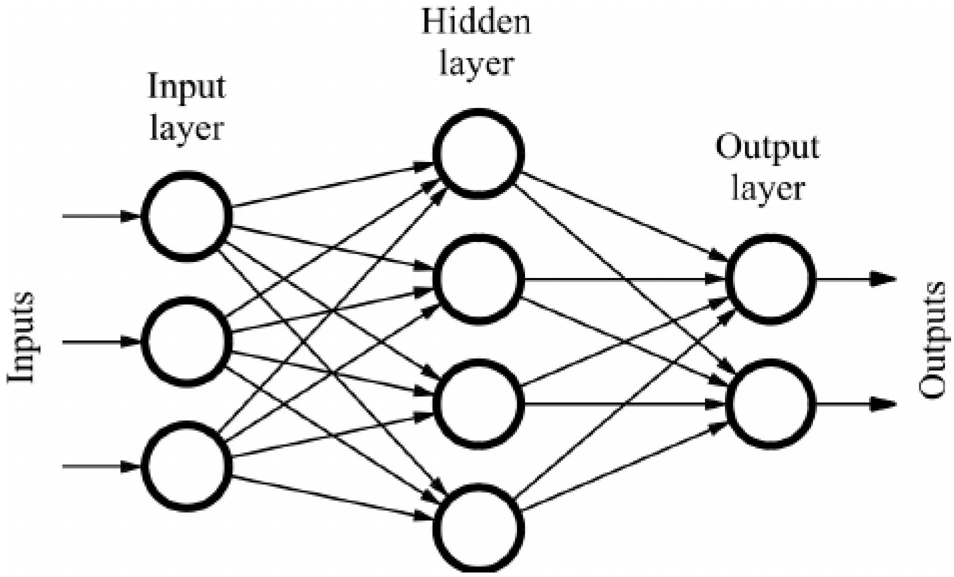


Рисунок 2.2. Модель нейронної мережі прямого поширення

Для кожної комірки розраховується:

$$f\left(w_0 + \sum_{i=1}^{n} x_i w_i\right) \qquad (2.2)$$

де w_i – вага, x_i – вхідне значення вектору. Функція f є функцією активації. Для нейронної мережі в даній роботі було використано функцію ReLU (rectified linear unit, або випрямляч) та Softmax для останнього слою.

Функція активації ReLU визначається як:

$$f(x) = \max(0, x) \qquad (2.3)$$

В той час Softmax має вигляд:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}} \qquad (2.4)$$

де вхідний вектор k трансформується в вектор σ діапазон значень якого від 0 до 1. Дана функція є узагальненням логістичної функції на випадок багатьох класів і є необхідною для здійснення багатокласової класифікації.

В рамках даної роботи була використана наступна архітектура нейронної мережі прямого поширення (Рисунок 2.3):

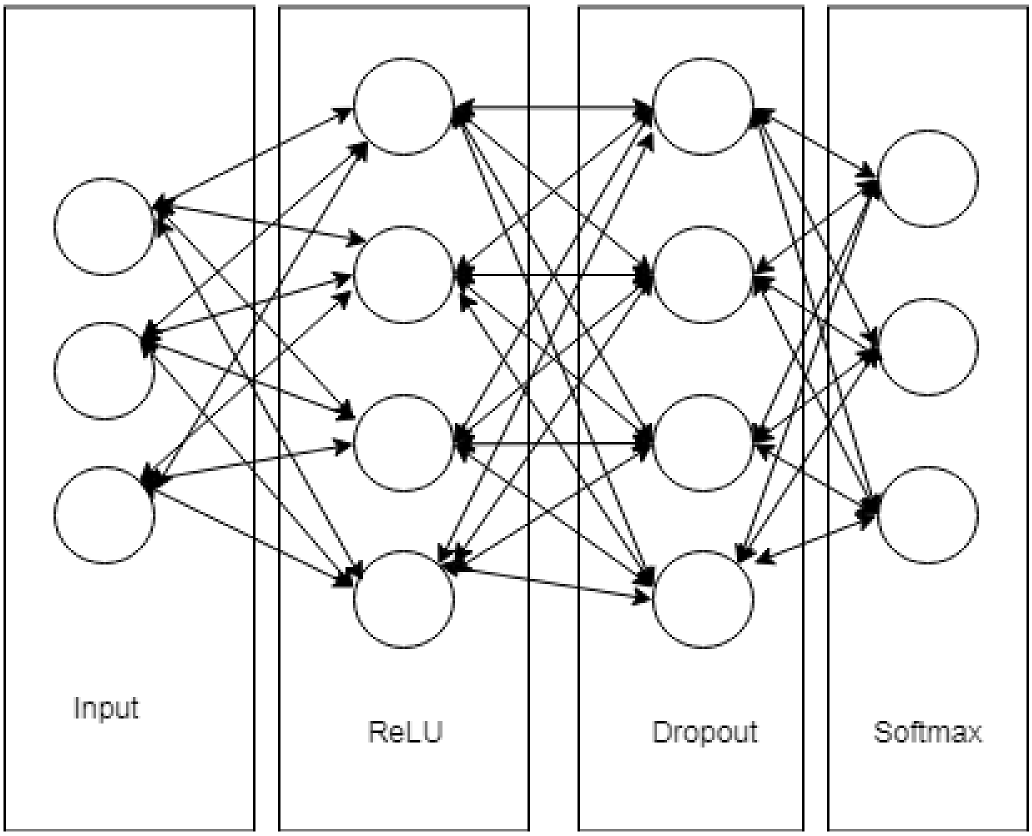


Рисунок 2.3. Використана модель нейронної мережі

Як можна бачити перший шар має функцію активації ReLU, визначеної у формулі 2.3. Наступний шар – шар виключення суть якого полягає у виключенні з певною імовірністю частини вхідних в шар нейронів. Включення даного шару в архітектуру нейронної мережі надає ефект запобігання перенавчанню нейронної мережі [31].

В рамках архітектури ймовірність виключення була встановлена 30%. Так як в даному випадку стоїть задачі класифікації – останній шар містить Softmax функцію активації і кількість нейронів яка відповідає кількості класів.

2.3 Модель рекурентної нейронної мережі

Враховуючи що набір даних являє собою потік сенсорів – важливу роль може грати саме послідовність активації сенсорів. Врахувати таку послідовність може саме рекурентна нейронна мережа [32].

Рекурентні нейронні мережі є типом нейронних мереж які використовують для аналізу потокової інформації. Їх використовують для аналізу тексту, розпізнавання мовлення, тощо. Їх характерною особливістю є можливість зберігати внутрішній стан що і дає можливість обробки потокової інформації. Загальний вигляд зображений на рисунку 2.4

Рисунок 2.4. Загальний рекурентної нейронної мережі

Однак класична рекурентна нейронна мережа мала проблеми зникаючого градієнту, коли градієнтні похибки зникають доволі швидко при навчанні. Саме тому було запропоновано модель довгої короткочасної пам'яті яка б вирішувала цю проблему [33].

Дана модель вводить вентилі зберігання, забування і виходу які контролюють те, яка частина вагів зберігаються або оновлюються під час навчання мережі. Розробка даної моделі дозволила аналізувати більш довгострокові залежності в даних. Її модель зображена на рисунку 2.5

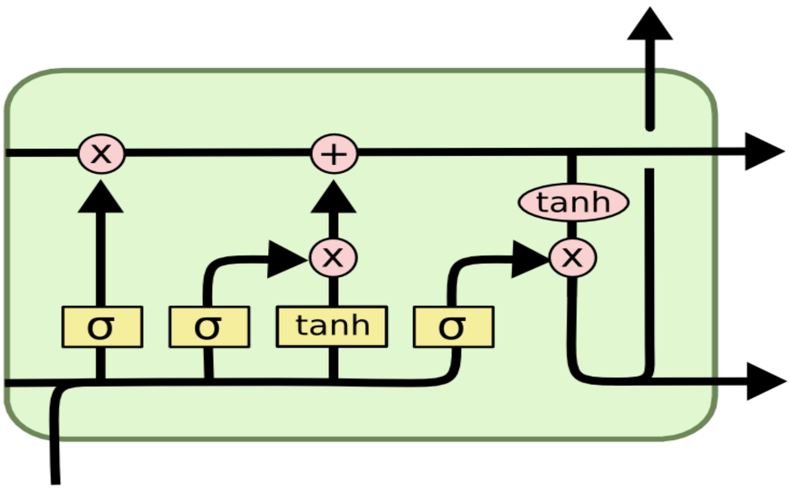


Рисунок 2.5. Модель довгострокової нейронної мережі

Модель визначається наступними рівняннями:

$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right)$$

$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right)$$

$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$

$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \widetilde{C_t}\right)$$

$$\widetilde{C_t} = tanh(x_t U^g + h_{t-1} W^g)$$

$$h_t = \tanh(C_t) * o_t$$

де U – в даному випадку матриця вагів, між входом і поточним шаром, i_t, f_t, o_t – вхідний, забування та вихідний вентилі. Кожен з вентилів має власну матрицю вагів W. Вхідний вентиль визначає, кількість зберігаємої інформації даної комірки. Вентиль забування визначає, частину зберігання стану попередньої комірки. Нарешті, вихідний вентиль визначає, частину внутрішнього стану комірок яка передається у інші шари або на інший крок. До вентилів застосовується сигмоїдальну функція і їх виходом є число від 0 до 1. Сигмоїдальна функція має вигляд:

$$\sigma(x) = \frac{1}{(1 + e^{-x})} \quad (2.5)$$

На базі $\widetilde{C_t}$ та C_{t-1} які являють собою новий стан комірки та минулий вираховується значення C_t. Вихідний стан h_t розраховується на базі перемноження розрахованого C_t до якого застосовується гіперболічна функція і вихідного вентилю.

Розроблена архітектура має наступний вигляд (Рисунок 2.6).

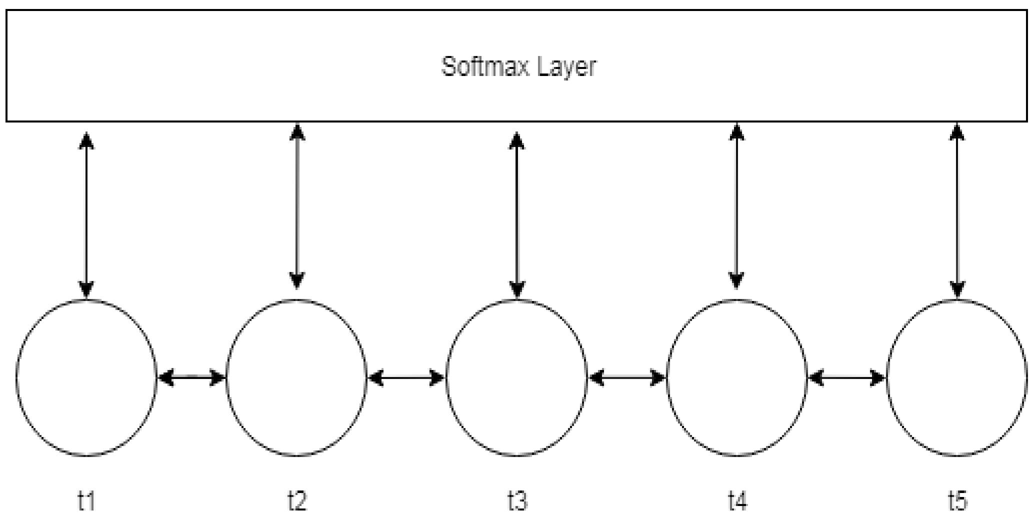


Рисунок 2.6. Використана модель довгострокової мережі

Так для здійснення класифікації вихід з комірок довгої короткочасної пам’яті має бути переданий надалі в і розрахований на базі softmax шару, для можливості багатокласової класифікації.

2.4 Метод опорних векторів

Більшість проаналізованих досліджень в якості методу класифікації обрали метод опорних векторів. Ідея даного методу класифікації полягає у пошуку найбільш оптимальної гіперплощини для поділу класів. Даний метод будує функцію класифікації F, яка визначається наступним чином:

$$F(x) = sign(\langle w, x\rangle + b) \quad (2.6)$$

де w – нормальний вектор до поділяючої площини, b – допоміжний параметр, x – елемент навчальної вибірки. Тобто ті екземпляри для яких F(x) = 1, відносяться до одного класу, а для яких F(x) = −1 – до іншого. Суть методу зображена на рисунку 2.7.

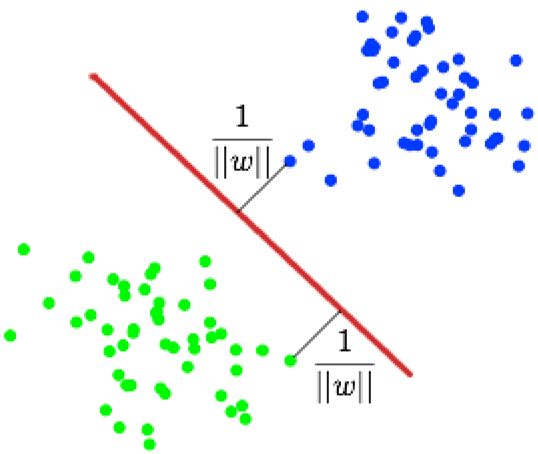


Рисунок 2.7. Пошук гіперплощини у методі опорних векторів

Так надалі необхідно обрати такі значення w та b при яких відстань до кожного з класів – максимальна. Дана відстань дорівнює $\frac{1}{\|w\|}$. Так проблему максимізації $\frac{1}{\|w\|}$ можна визначити як проблему мінімізації $\|w^2\|$ [34]. Так задачу:

$$\begin{cases} \arg min_{w,b} \|w\|^2, \\ y_i(\langle w, x\rangle + b) \geq 1, \qquad i = 1, \ldots, m. \end{cases} \quad (2.7)$$

надалі вирішують методом невизначених множників Лагранжа. Враховуючи той факт, що досить часто зустрічається проблема того, що класи не можна поділити однією лінією – використовують функцію відображення навчальної вибірки у простір більшої розмірності, таким чином, щоб вибірку можна було б розділити лінією. Приклад даних які не розділяються лінією зображений на рисунку 2.8.

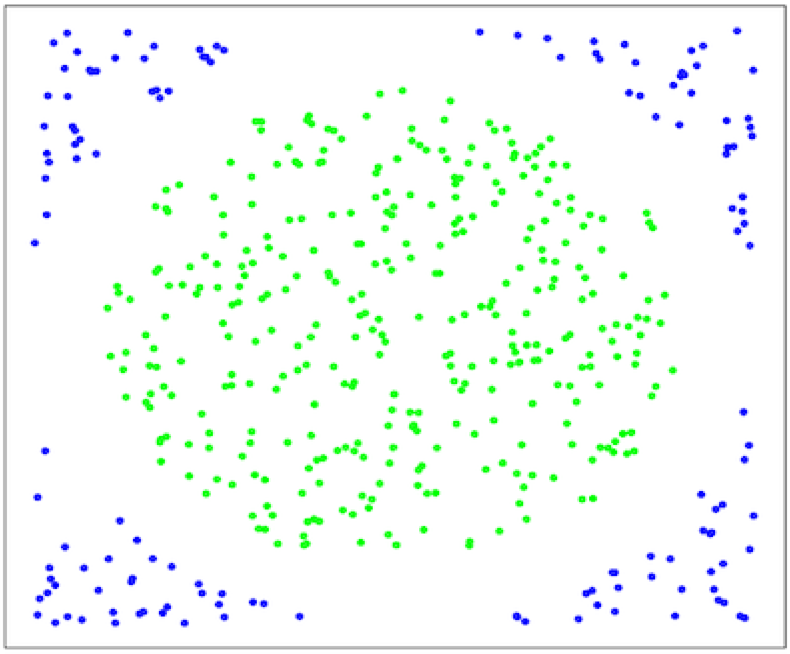


Рисунок 2.8. Проблема нероздільності даних лінією

Таку функцію як правило називають ядром [35]. В представлених дослідженнях було використано радіально-базисне ядро:

$$K(x, x') = e^{\left(-\frac{\|x-x'\|}{2\sigma^2}\right)^2} \tag{2.8}$$

де σ – вільний параметр, а x та x' – вхідні вектори. Слід зазначити, що даний метод розрахований лише на бінарну класифікацію. Для можливості здійснення багатокласової класифікації – даний класифікатор тренують кількість раз яка відповідає кількості класів, здійснюючи кожний раз бінарну класифікацію поточного класу від усіх інших [36]. Враховуючи величину вхідних даних та достатньо велику кількість активностей – тренування даного методу потребує достатньо велику кількість часу.

2.5 Латентне розміщення Діріхле

На сьогоднішній день існує ряд моделей тематичного моделювання для аналізу змісту документів і слів [37]. В рамках цих моделей допускається ідея того що документ являє собою набір слів і має певний розподіл відповідних йому тем. Так моделювання тем залежить від розподілу слів у документах. Приклад такого підходу зображений на рисунку 2.9. Такі моделі можна використовувати не лише в рамках аналізу тексту, а також в багатьох інших предметних областях де є необхідним подібний аналіз.

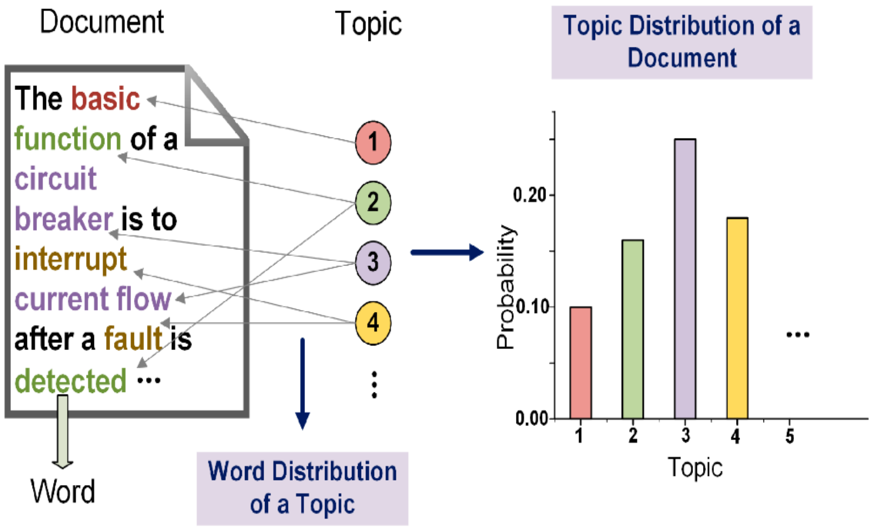


Рисунок 2.9. Приклад моделювання теми

Латентне розміщення Діріхле є однією з таких моделей. Дана модель дещо покращує результати інших існуючих подібних моделей, а саме має стійкість до проблеми перенавчання і можливість добавляти нові документи в модель [38]. Так як кожному документу можна поставити у відповідність певний розподіл тем – в даній моделі апріорі вважається що це розподіл Діріхле.

Враховуючи що у певний момент часу конкретний сегмент дещо обмежений в у інформації, важливо принести частину апріорної інформації з навчальної вибірки. Так якщо вважати що документ – вікно активності, ознаки цього вікна – слова, а сам розподіл активностей є темами – можна використати дану модель для включення в кожний сегмент інформації про імовірнісний розподіл активностей на базі ознак в цьому сегменті.

Отже в даному випадку можна розробити модель класифікації використовуючи розподіл ймовірностей активностей у певному вікні. Так найбільш релевантною можна вважати активність яка максимізує умовну імовірність вектору ознак у певному вікні:

$$P(y_i = A\backslash w_i) = P(A\backslash F_i^*) = P(A)P(F_i^*\backslash A) \quad (2.9)$$

де A – розподіл активностей, w_i – сегмент, F_i^* - ознаки вікна активності. Для можливості створення подібного класифікатору необхідно у кожному вікні активності створити вектори ознак який включає в себе вектор частот активацій сенсорів і час. В даному випадку для цієї моделі час трансформується у 24-мірний вектор, який має значення 1 у місцях які відповідають часу тривалості активності.

З праць [38, 39] можна отримати:

$$P(F^*) = \sum_{Q} \prod_{f_i^* \in F^*} \sum_{A^*} P(f_i^*\backslash A)\, P(A^*\backslash Q)P(Q) \quad (2.10)$$

де Q – вікно активності, A^* - розподіл тем який був отриманий після застосування моделі. Виходом моделі латентного розміщення Діріхле є розподіли $P(F^*\backslash A^*)$ і $P(A^*\backslash Q)$. Враховуючи що дана модель є моделлю без вчителя – її виходом є розподіл власних тем A^*. Надалі необхідно визначити функцію відображення тем моделі і справжніх активностей:

$$A = \varphi(A^*) \quad (2.11)$$

Надалі визначимо для сегменту вектор ознак F^* і використаємо $P(F^*\backslash A^*)$ щоб вивести:

$$P(A\backslash F_i^*) = P(\varphi(A^*))P(F_i^*\backslash A) \quad (2.12)$$

Так як А є множиною:

$$P(A\backslash F_i^*) = \bigcup_{a_j} P\left(a_j\backslash F_i^*\right) \quad (2.13)$$

Відповідно до теореми Байеса, можемо мати [40]:

$$P\left(a_j\backslash F_i^*\right) = P\left(\varphi\left(a_j^*\right)\right)P\left(F_i^*\backslash a_j^*\right) \quad (2.14)$$

Враховуючи що ознаки є незалежними:

$$P\left(F_i^*\backslash a_j^*\right) = \prod_{i=1}^{m} P\left(f_i^*\backslash a_j^*\right) \quad (2.15)$$

Тоді маємо:

$$\prod_{i=1}^{m} P\left(f_i^* \backslash a_j^*\right) = log \prod_{i=1}^{m} P\left(f_i^* \backslash a_j^*\right) = \sum_{i=1}^{m} logP\left(f_i^* \backslash a_j^*\right) \quad (2.16)$$

В свою чергу $\varphi\left(a_j^*\right)$ є апріорною імовірністю активності a_j^* і тому має значення частоти a_j^* в вікнах активності. Так розподіл активностей $P(A \backslash F_i^*)$ надалі буде включено до вектору ознак.

2.6 Особливості формування вектору ознак для обраних моделей і тренування

Для моделей методу опорного векторів, нейронної мережі прямого поширення та випадкового лісу були обрані наступні ознаки: час початку і кінця, часова тривалість сегменту, частота активації сенсорів, ідентифікатори двох останніх сенсорів і ознаки отримані використовуючи модель Діріхле. Так врахування двох сенсорів надасть більше контекстної інформації про вікно моделі. Частоти активацій були зважені використовуючи матрицю взаємної інформації яка визначена у формулі 1.3. Цей метод зважування частот можна вважати більш доцільним так як розраховується в рамках вікна активності. Для можливості використання дати у векторі ознак – сама дата визначається як нормалізована сума часу, хвилин та секунд.

Для рекурентної нейронної мережі в свою чергу на кожному кроці використовується інформація про ідентифікатор сенсору, час активації та значення сенсору.

Для оптимізації нейронної та рекурентної мережі був використаний метод Adam(adaptive moment estimation). Цей алгоритм використовує переваги двох інших методів оптимізації, а саме AdaGrad (Adaptive Gradient Algorithm) – який використовує власні темпи навчання для параметрів для вирішення проблемі розріджених градієнтів та RMSProp (Root Mean Square Propagation) який використовує також власні темпи, при цьому враховуються середнє величина минулих градієнтів що надає можливість навчати модель в умовах наявності викидів в даних. Замість використання лише середнього значення, даний метод також використовує дисперсію. Так вираховуються занепадаюче середнє минулих градієнтів:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \qquad (2.17)$$

А також квадрат минулих градієнтів:

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) {g_t}^2 \qquad (2.18)$$

Де β_1 та β_2 – константи, m_{t-1} та v_{t-1} середнє та квадрат середніх градієнтів, g_t є значенням градієнтів на поточному кроці. Так як вектори m_t та v_t ініціалізуються нулями для уникнення проблем на початковому етапі виконується заміна:

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t} \qquad (2.19)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \qquad (2.20)$$

Так на базі 2.19 та 2.20 визначається правило оновлення параметрів:

$$\theta_{t+1} = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t} + \varepsilon} \widehat{m}_t \qquad (2.21)$$

Де θ – вектор параметрів, ε – константа. Автори дослідження пропонують значення 0.9 для β_1, 0.999 для β_2 і 10^{-8} для ε [41].

Також для врахування проблеми незбалансованості для усіх моделей при навчанні ваги класів були встановлені зворотно пропорційно їх частоті у наборі даних.

2.7 Аналіз результатів моделювання

В рамках роботи було проведено ряд досліджень крім тих що були описані раніше в цьому розділі. Так було запропоновано підхід стекінгу який полягає у використанні декількох класифікаторів для здійснення класифікації [42]. Ідея підходу зображена на рисунку 2.10. В даному випадку використовується перехресна перевірка, суть якої в тому щоб поділити вибірку на підвибірки і надалі обучити класифікатор на одній частині і перевірити на іншій.

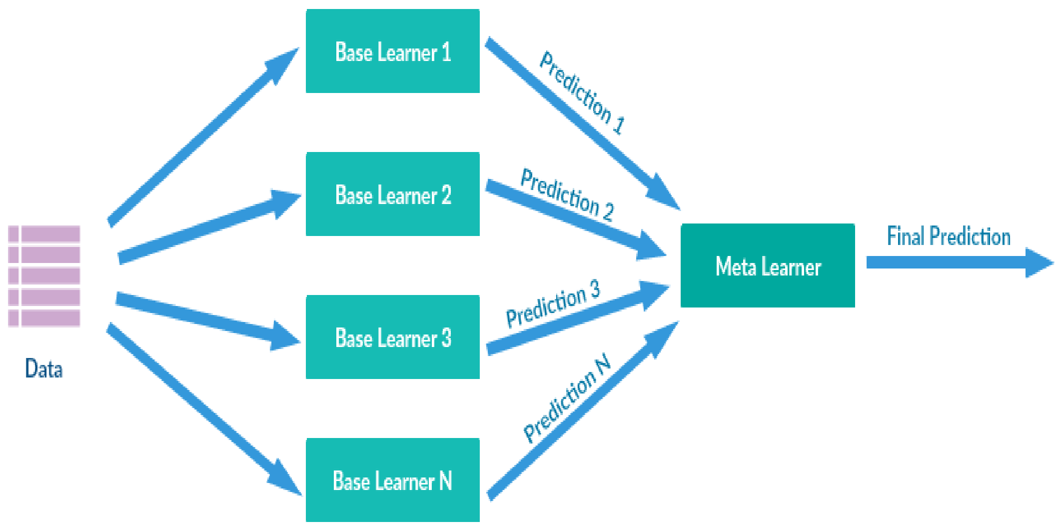


Рисунок 2.10. Стекінг класифікаторів

Так формується нова навчальна вибірка, яка складається з результатів класифікації на тих підвибірках на яких проходила перевірка. На базі сформованої вибірки навчається мета-модель яка і стає новою моделлю класифікації. В рамках дослідження на обраних класифікаторах було зроблено висновок що точність цього підходу не є достатньою для подальшого використання.

Також була спроба використати розподіл ймовірностей класифікації при тренуванні моделі. Так модель класифікації була тренована без іншої активності і для кожної активності було розраховане среднє значення впевненості класифікації. Так якщо впевненість класифікації була нижча ніж среднє значення мінус обрана константа – сегменту присвоювався клас інший. Також цей підхід був удосконалений моделюючі для кожної активності гаусівський розподіл. Однак даний підхід надав досить малу точність, що говорить про те що впевненість у класифікації інших активностей є приблизно аналогічною з анотованими активностями.

Враховуючи існуючу в даних незбалансованість – також було вирішено використати метод зменшення розмірності навчальної вибірки для боротьби з проблемою незбалансованості даних. Крім того зменшення навчальної вибірки могло б прискорити процес тренування моделей. Ідея такого методу в використанні центроїд сформованих методом к-середніх на базі яких надалі формується нова навчальна вибірка меншої розмірності [43]. Використання цього методу також не призвело до покращення результатів, що обумовлено певною неоднорідністю даних.

Крім описаних раніше архітектур було побудовано також архітектури з більшою кількістю комірок так шарів. В результаті проведення дослідження стало зрозуміло що комплексність моделей мереж рекурентної і прямого поширення не сильно залежить від результативної точності класифікації.

В рамках роботи були протестовані усі методи класифікації. Використаємо метрики визначені у формулах 1.4 та 1.5 для оцінки точності. Спочатку необхідно привести точність класифікації без врахування іншого класу (Таблиця 2.1).

Таблиця 2.1. Точність класифікації без іншого класу

	Метод опорних векторів	Модель випадкового лісу	Нейронна мережа прямого поширення	Рекурентна нейронна мережа
Точність	84.7	93.1	78.	91
F1	78.1	87.3	75.2	86.5

Як можна бачити найбільшу точність має метод випадкового лісу. Надалі приведемо точності класифікації з врахуванням іншого класу (Таблиця 2.2).

Так при включенні іншого класу до навчальної вибірки деякі алгоритми втрачають можливість класифікувати інші, менш репрезентативні в даних тренування класи що і обумовлює таке значне падіння точності класифікації. Як можна бачити модель випадкового лісу має достатню точність і при здійсненні класифікації іншого класу. Крім того серед приведених алгоритмів метод опорних векторів навчається найбільшу кількість часу, в той час коли випадковий ліс, завдяки паралелізації побудови дерев – найменшу.

Таблиця 2.2. Точність класифікації з іншим класом

	Метод опорних векторів	Модель випадкового лісу	Нейронна мережа прямого поширення	Рекурентна нейронна мережа
Точність	59	86.8	52.1	68
F1	54.2	79.9	49	54

Отже найбільш оптимальна в даному випадку є модель випадкового лісу. Представлені результати перевищують точність проаналізованих досліджень і модель може бути використана у реальних умовах.

Висновки до розділу 2

В рамках другого розділу було:

1. Для вирішення поставленої задачі було запропоновано ряд моделей машинного навчання і приведено їх опис. Так кожна з представлених моделей має такі характеристики які роблять її придатною для вирішення даної проблеми.
2. Було описано використання ідеї моделювання тем для даної проблеми, так, використовуючи метод Латентного Розміщення Діріхле в кожний сегмент включено інформацію про імовірнісний розподіл активностей на базі ознак в цьому сегменті. Тобто в умовах певної обмеженості представленої інформації в сегменті – включається більше інформації про активності.
3. Описані особливості формування векторів ознак моделей і тренування з врахуванням характеристик набору даних.
4. Описано результат експериментальних підходів які були протестовані для даної задачі. Так дана задача має певні характеристики, які не дають

застосувати деякі існуючі підходи для неї.

5. Приведено результат класифікації описаних моделей. Так в умовах відсутності іншого класу точність усіх класифікаторів є достатньо високою, але в умовах включення іншого класу точність значно падає. Це обумовлено тим що класифікатори втрачають можливість класифікувати ряд класів. Серед усіх класифікаторів найбільш оптимальним можна вважати модель випадкового лісу точність якого перевищує результати приведені в дослідженнях, що дає можливості використання цього методу у реальних умовах.

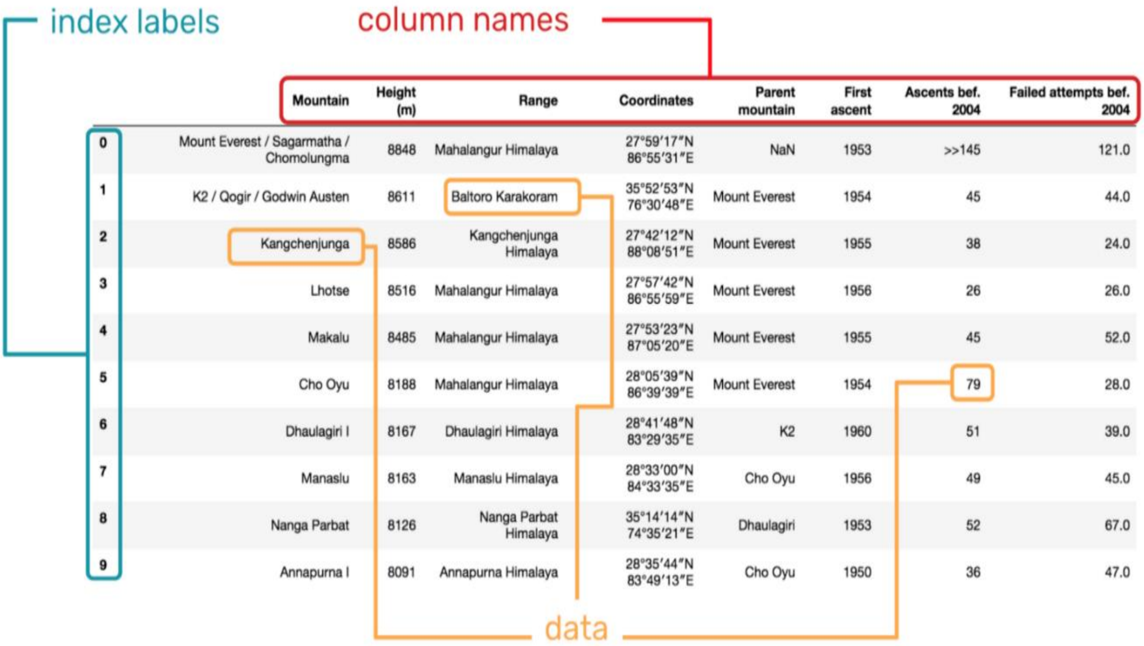
3. РОЗРОБКА СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ РОЗУМНОГО БУДИНКУ

Вибір засобів розробки є важливим етапом при розробці програмної системи, так як від них залежить якість результуючого продукту. Так від вибору бібліотек для машинного навчання може залежати швидкодія системи, так як існуючі бібліотеки мають різні реалізації алгоритмів. Крім того важливим є засіб реалізації користувацького інтерфейсу для створення найбільш комфортних умов застосування системи. Також важливо привести загальну схему функціонування системи, структуру системи і діаграму класів для пояснення особливостей програмної архітектури розробленої системи.

3.1 Програмні засоби розробки

Розробка даного рішення велася у середовищі розробки PyCharm, використовуючи Python 3.6. Python - об'єктно-орієнтована, інтерпретована мова програмування з динамічною типізацією. На сьогоднішній день дана мова використовується в багатьох галузях: автоматизація, наукові розрахунки, веб-розробка, аналітика і машинне навчання, тощо [44]. Її вибір обумовлений тим, що на сьогоднішній день існує велика екосистема з бібліотек саме для машинного навчання що і є необхідним в рамках даної роботи.

Для здійснення маніпуляцій з даними було використано бібліотеку Pandas. Це бібліотека з відкритим кодом яка пропонує ряд можливостей щодо більш зручної роботи з даними [45]. Основною структурою даних є DataFrame яка підтримує індексацію. Певним чином дана структура є схожою на sql таблиці. На рисунку 3.1 зображено приклад даної структури.



	Mountain	Height (m)	Range	Coordinates	Parent mountain	First ascent	Ascents bef. 2004	Failed attempts bef. 2004
0	Mount Everest / Sagarmatha / Chomolungma	8848	Mahalangur Himalaya	27°59′17″N 86°55′31″E	NaN	1953	>>145	121.0
1	K2 / Qogir / Godwin Austen	8611	Baltoro Karakoram	35°52′53″N 76°30′48″E	Mount Everest	1954	45	44.0
2	Kangchenjunga	8586	Kangchenjunga Himalaya	27°42′12″N 88°08′51″E	Mount Everest	1955	38	24.0
3	Lhotse	8516	Mahalangur Himalaya	27°57′42″N 86°55′59″E	Mount Everest	1956	26	26.0
4	Makalu	8485	Mahalangur Himalaya	27°53′23″N 87°05′20″E	Mount Everest	1955	45	52.0
5	Cho Oyu	8188	Mahalangur Himalaya	28°05′39″N 86°39′39″E	Mount Everest	1954	79	28.0
6	Dhaulagiri I	8167	Dhaulagiri Himalaya	28°41′48″N 83°29′35″E	K2	1960	51	39.0
7	Manaslu	8163	Manaslu Himalaya	28°33′00″N 84°33′35″E	Cho Oyu	1956	49	45.0
8	Nanga Parbat	8126	Nanga Parbat Himalaya	35°14′14″N 74°35′21″E	Dhaulagiri	1953	52	67.0
9	Annapurna I	8091	Annapurna Himalaya	28°35′44″N 83°49′13″E	Cho Oyu	1950	36	47.0

Рисунок 3.1 Приклад датафрейму

Крім того дана бібліотека надає можливості щодо запису та читання даних з файлів, здійснення аналізу даних а також ряд методів щодо зручної маніпуляції даними. В цілому бібліотека є досить ефективною та оптимізованою що надає можливість її застосування в системі.

Відкрита бібліотека Scikit-learn містить реалізацію більшої кількості існуючих класичних методів машинного навчання регресії, кластеризації, класифікації, регресії, ансамблевих моделей, а також зменшення простору [46]. Зокрема серед них: метод опорних векторів, випадковий ліс, градієнтний бустінг, к-середніх, DBSCAN, метод головних компонент тощо. Крім того бібліотека надає ряд допоміжних функцій. Зокрема різні можливості щодо обробки даних, тобто здійснення нормалізації і трансформації ознак. Також є інструментарій щодо вибору параметрів моделі. Більша частина бібліотеки написана на мові Python, однак для ефективності частина реалізована на Cython – мові програмування, для інтеграції з кодом на С. Завдяки ефективності цієї бібліотеки – вона використовується великою кількістю компаній, саме тому була обрана ця бібліотека. В даній роботі використана реалізація моделі випадкового лісу даної бібліотеки і методу опорних векторів. Слід зазначити що в бібліотеці цей метод реалізований на базі LibSVM – однією з найбільш ефективних та оптимальних реалізацій даного методу [47].

Для побудови нейронних мереж була використана бібліотека Keras. Keras – відкрита бібліотека для зручної та ефективної розробки моделей глибокого навчання створена на мові Python [48]. Вона може функціонувати поверх інших існуючих бібліотек таких як Tensorflow, Theano або Microsoft Cognitive Toolkit. Так в даній бібліотеці реалізовано найбільш використовувані блоки нейронних мереж. Серед них різні типи шарів такі як рекурентні або згорткові, функції активації та оптимізатори. Крім цього реалізовано ряд інструментів щодо здійснення обробки та трансформації даних. Загалом розробка в рамках даної бібліотеки є досить зручною завдяки модульності створюваної моделі і можливості добавляти в неї власні модулі. В рамках роботи Keras було використана для побудови нейронної мережі прямого поширення та рекурентної нейронної мережі.

Для використання моделі Латентного Розміщення Діріхле була використана відкрита бібліотека з відповідною назвою – lda. Основна частина даної бібліотеки реалізована на C – тому є досить ефективною. Дана бібліотека надає подібній до Scikit-learn інтерфейс, що робить її досить зручною у використанні.

Для створення користувацького інтерфейсу була використана бібліотека Dash – відкрита Python бібліотека для створення веб інтерфейсів Як правило використовується для створення графічного інтерфейсу для застосунків у сфері аналізу даних та моделювання. При написанні застосунків непотрібно писати Javascript кодом або HTML так як дана бібліотека надає Python інтерфейсу до великої кількості веб-компонентів.

Застосунки на Dash являють собою веб-сервера на базі бібліотеки Flask і функціонують на базі передачі пакетів JSON через HTTP-запити [49]. Для відображення компонент використовується React.js – Javascript бібліотеки для інтерфейсів користувача. Як було сказано компоненти Dash являють собою Python класи які зберігають властивості та значення React компонента і надалі серіалізуються як JSON. Dash надає набір інструментів для простої реалізації компонентів React як компонентів, які можуть бути використані в Dash. Цей набір інструментів використовує динамічне програмування для автоматичного створення стандартних класів Python. Отримані класи Python представляють собою компоненти Dash і є зручними для користувачів: вони мають автоматичну перевірку аргументів, документацію до них тощо. Перевагою використання Dash є змога використовувати додатки Dash у багатозадачному режимі: кілька користувачів можуть мати власні сеанси під час взаємодії з додатком Dash одночасно. Розробники мають можливість використання каскадної таблиці стилів для компонент для можливості створення ще більше насичених інтерфейсів. Для візуалізації даних Dash містить компонент Graph який будує графіки використовуючи plotly.js. Plotly.js - відкрита, швидка бібліотека яка підтримує повний спектр наукових, фінансових та бізнес-діаграм. Plotly.js побудований на базі D3.js (для якості графіків, експорту векторних зображень) та WebGL (для високопродуктивної візуалізації). Враховуючи усі плюси даної бібліотеки – вона ідеально підходить для візуалізації результатів моделювання у даній роботі.

3.2. Загальна схема функціонування системи

Входом системи є дані з потоку сенсорів необхідного формату зібрані протягом певного періоду часу. Ці вхідні дані можна поділити на дані тренування і тестування моделі. В даному випадку передбачається що даними тренування є анотовані дані зібрані під час життя мешканця будинку, а даними тестування є ті, що будуть поступати у реальному часі. Однак дані тестування можуть мати теж анотації для можливості подальшого аналізу результатів класифікації.

Необхідно описати схему функціонування на випадок завантаження даних тренування і тестування окремо – так як послідовність роботи буде дещо відрізнятися. Схема роботи програмної системи при завантаженні даних тренування описана на рисунку 3.2.

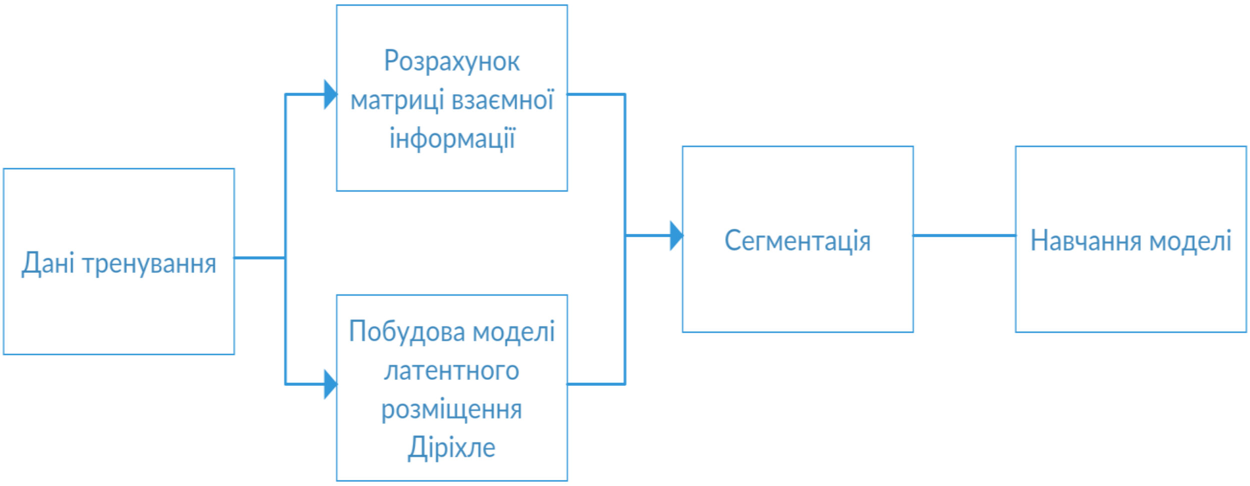


Рисунок 3.2. Схема функціонування системи на даних тренування

Як можна бачити при завантаженні розраховується матриця взаємної інформації і будується модель латентного розміщення Діріхле на базі вікон активностей. Після цього відбувається сегментація і навчання моделі. Надалі тренована модель застосовується при здійсненні класифікації даних тестування. Схема роботи програмної системи при завантаженні даних тестування описана на рисунку 3.3.

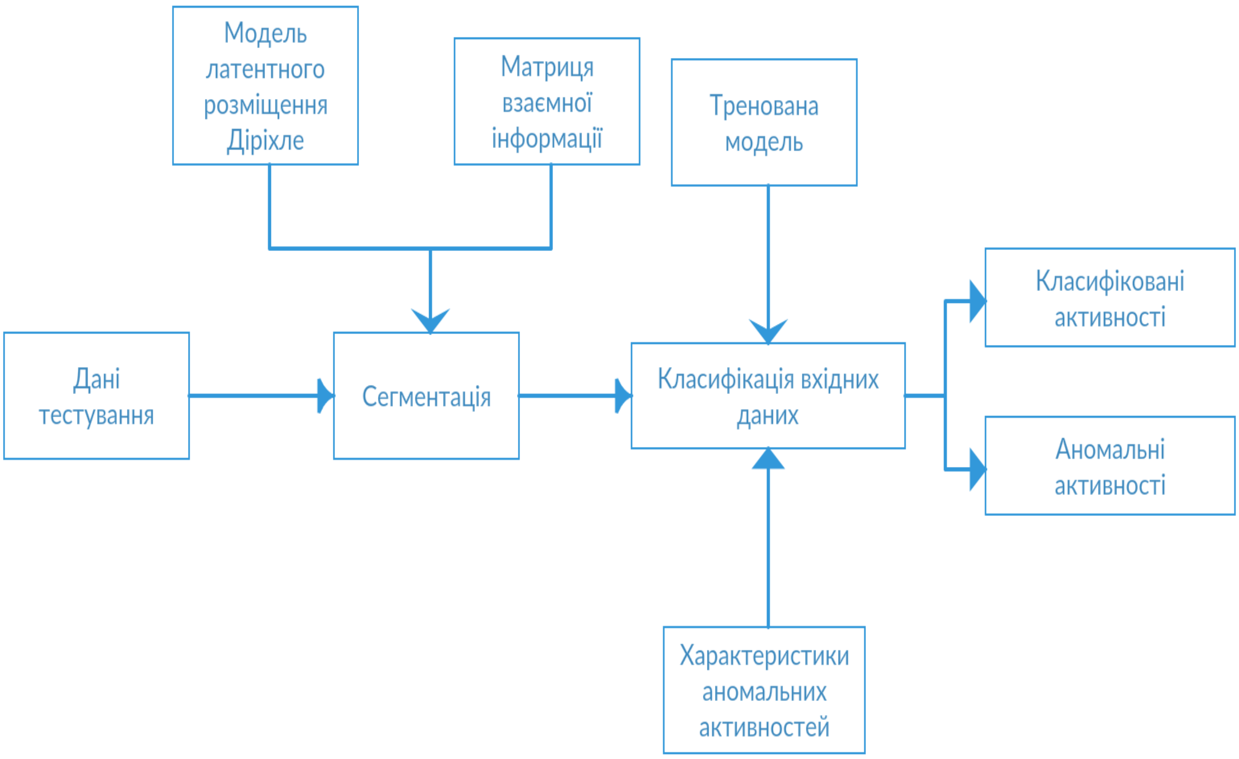


Рисунок 3.3. Схема функціонування системи на даних тестування

При завантаженні даних тестування відбувається сегментація використовуючи розраховані раніше матрицю взаємної інформації і модель латентного розміщення Діріхле. Сегменти класифікуються і подаються на вхід тренованій моделі. При наявності визначених характеристик аномальних активностей – є можливість переглянути перелік знайдених аномальних активностей в даних.

3.3. Сценарії використання системи

Для опису сценаріїв використання системи застосуємо діаграму використання. Так на даній діаграмі у відповідність кожному користувачу поставлена множина здійснюваних можливих атомарних дій в рамках системи [50]. Діаграма використання розробленої системи представлена на рисунку 3.4.

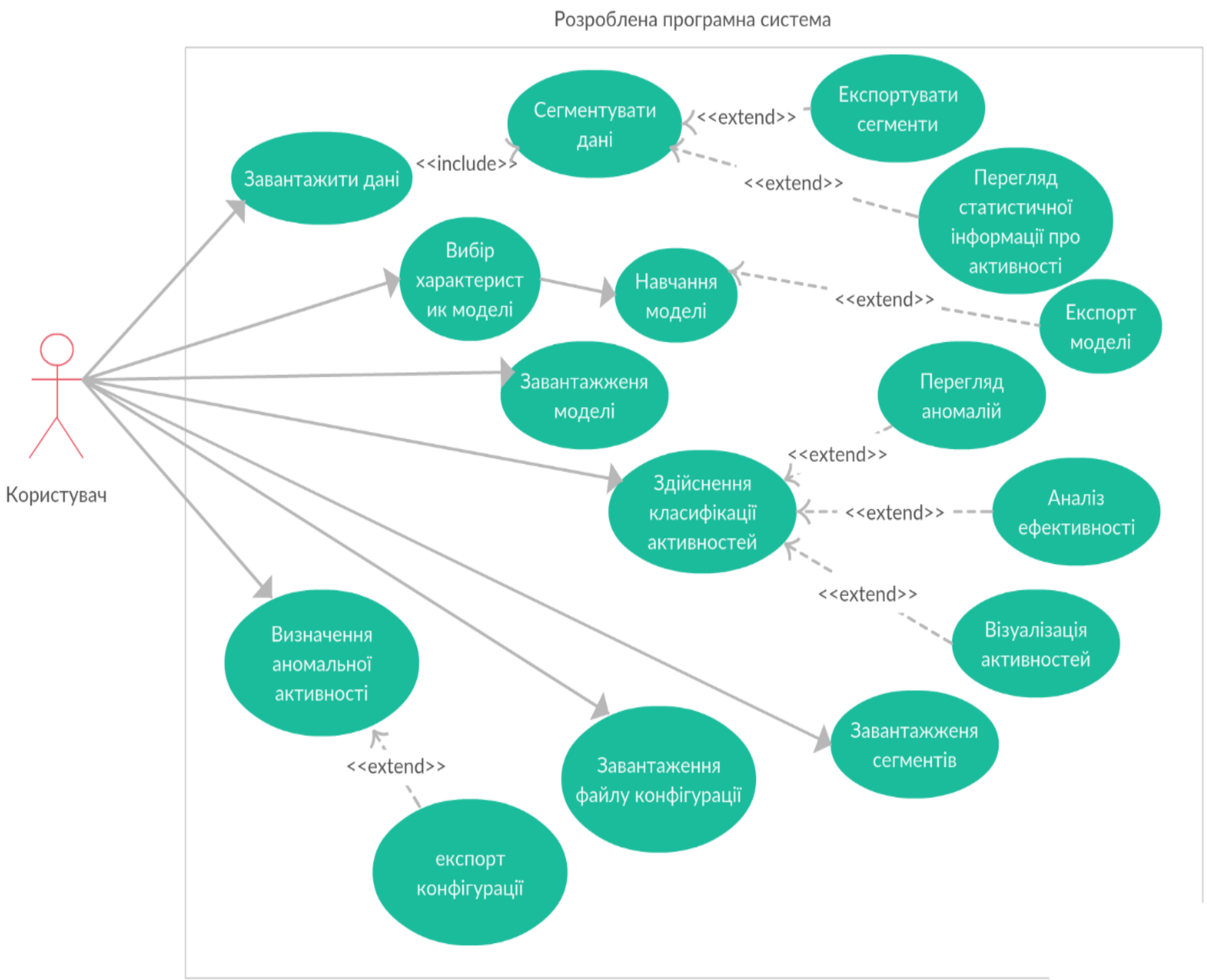


Рисунок 3.4. Діаграма використання розробленої системи

В рамках системи існує всього лише один користувач. При завантаженні даних тренування та тестування, відразу відбувається їх сегментація. Після процесу завантаження користувач має можливість експортувати результуючі сегменти, а також переглянути статистичну про характеристики розподілу даних, а саме розподіл активностей, розподіл сенсорів, загальну інформацію а також матрицю взаємної інформації.

Користувач має можливість вибрати модель між описаними в розділі 2 а також їх параметри для навчання. Після навчання користувач може здійснити експорт моделі – і при необхідності імпорт в майбутньому. Також користувач може визначити характеристики аномальних активностей і при необхідності експортувати файл з ними. Після здійснення класифікації активностей, користувач може переглянути візуалізовані активності, проаналізувати ефективність моделі та при наявності визначених характеристик аномалій – переглянути наявні в даних аномалії. Користувач замість завантаження даних і сегментації може відразу завантажити сегменти для заощадження часу при великих об’ємах даних.

3.4. Структура програмного забезпечення

Існуючі пакети в розробленій програмній системі приведені на рисунку 3.5.

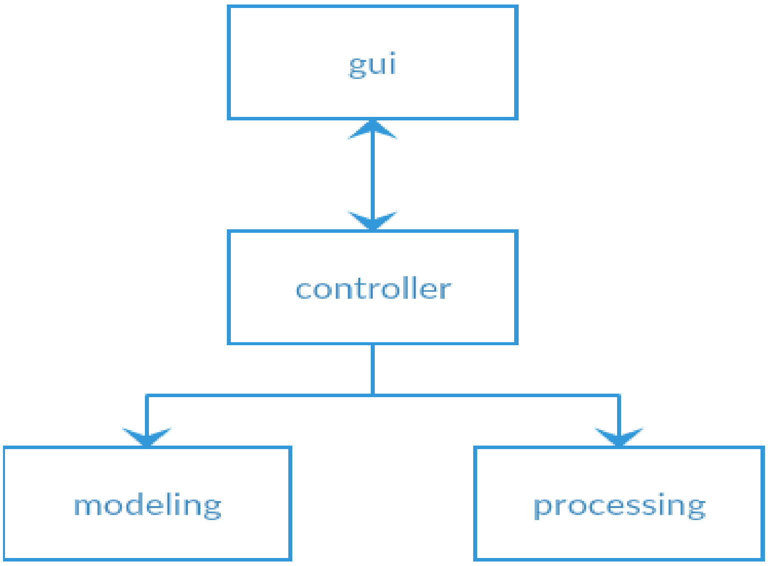


Рисунок 3.5. Пакети програмної системи

В програмній системі існує чотири пакети: gui для користувацького інтерфейсу, controller для зв'язування моделі і користувацького інтерфейсу, модуль modeling містить реалізації моделей класифікації, а processing – реалізації сегментації та формування вектору ознак.

Також опишемо основні класи розробленої програмної системи використовуючи діаграму класів. На даній діаграмі зображені класи, відношення між ними а також їх методи та атрибути [51]. Діаграма класів розробленої системи представлена на рисунку 3.6.

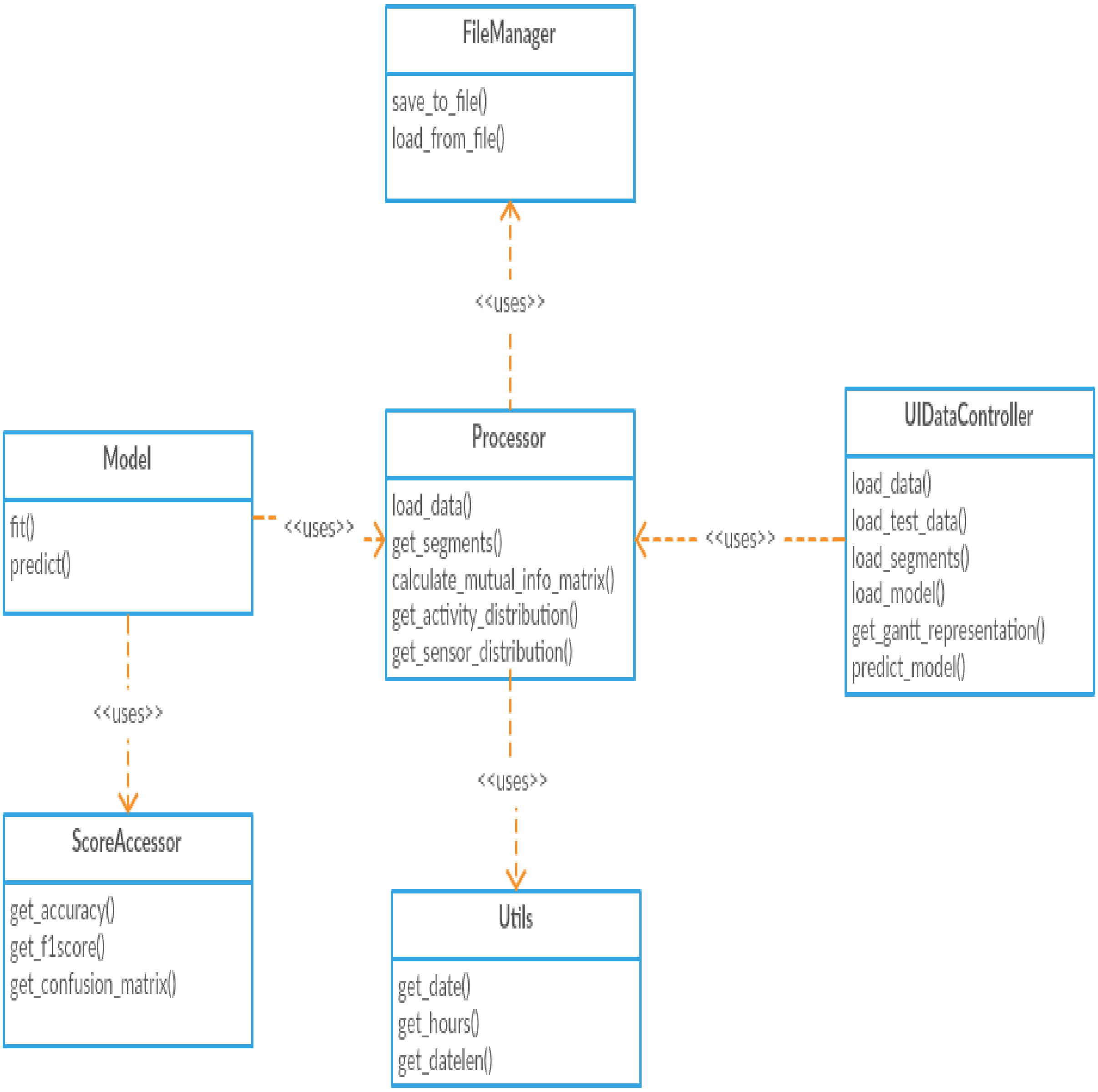


Рисунок 3.6. Діаграма класів розробленої системи

Так на діаграмі представлено 5 основних класів різних пакетів а також основні методи необхідні для опису функціональності класу. Клас Processor виконує основні функції щодо обробки даних та їх розподіл на сегменти. Цей клас використовує клас FileManager для завантаження набору даних або вже оброблених даних та їх збереження. Зберігання даних є важливим по причині того, що із-за розмірності даних їх обробка займає багато часу, що свою чергу затримує час необхідний на тренування і верифікацію різноманітних моделей. Клас Utils містить деякі допоміжні методи необхідні для обробки даних, в своїй більшості для роботи з датами. Клас Model містить безпосередньо саму прогнозуючу модель та містить методи для навчання і прогнозування даних. Клас ScoreAccessor має методи для оцінки ефективності розробленого алгоритму за декількома метриками. Клас UIDataController, був розроблений для зв'язку з графічним інтерфейсом і містить загалом методи обробки подій.

Висновки до розділу 3.

В рамках третього розділу було:

1. Описані обрані засоби розробки, проаналізовані їх особливості та переваги які стали умовою вибору. Так були обрані саме ті засоби які забезпечують найбільшу ефективність і найбільш зручні умови використання результуючої програмної системи.
2. Описана загальна схема роботи окремо при завантаженні даних тестування і тренування. Так при завантаженні даних тренування розраховується матриця взаємної інформації та модель Латентного Розміщення Діріхле і тренується модель класифікації. При завантаженні даних тестування – виконується сегментації та класифікації, використовуючи інформацію отриману при тренуванні. Виходом класифікації є множина активностей та перелік аномальних

активностей на базі заданих характеристик.

3. Описані сценарії роботи користувача з розробленою системою використовуючи діаграму використання. Так користувач має ряд можливостей щодо тренування моделі, визначення аномальних активностей, здійснення класифікації та перегляду її результатів. Особливості візуалізації результату класифікації і знайдених аномальних активностей буде приведено в наступному розділі.
4. Наведені пакети програмного забезпечення і на базі діаграми класів – приведено основні класи програмної системи.

4. МЕТОДИКА РОБОТИ КОРИСТУВАЧА

Для можливості зручного використання розробленої програмної системи надалі важливо детально описати технічні вимоги користувача і розроблений графічний інтерфейс.

4.1 Технічні вимоги

Розроблене рішення може бути запущене на будь-якій операційній системі для якої портований інтерпретатор Python, серед яких Linux, macOS або Windows. Для запуску програми необхідно, щоб на робочій машині користувача був встановлений Python та перелік бібліотек необхідної версії які були зазначені у розділі 3. Для спрощеної роботи з бібліотеками був використаний дистрибутив Anaconda, який використовується як правило для аналітичних застосунків, які використовують машинне навчання [52]. Саме встановлення даного дистрибутиву рекомендується користувачу. Враховуючи специфіку програмного продукту користувачам рекомендується мати 8 гігабайт оперативної пам'яті із-за величини наборів даних даної предметної області.

4.2 Інтерфейс програмного продукту

Розроблена програмна система являє собою багатосторінковий веб-застосунок навігація по якому відбувається в меню зверху програми. Меню зображене на рисунку 4.1.

Рисунок 4.1 Меню розробленого програмного застосунку

Робота користувача з системою починається на сторінці Control Panel, яка надає користувачу можливості щодо завантаження даних потоку сенсорів, завантаження сегментів та їх експорту. Дана сторінка зображена на рисунку 4.2.

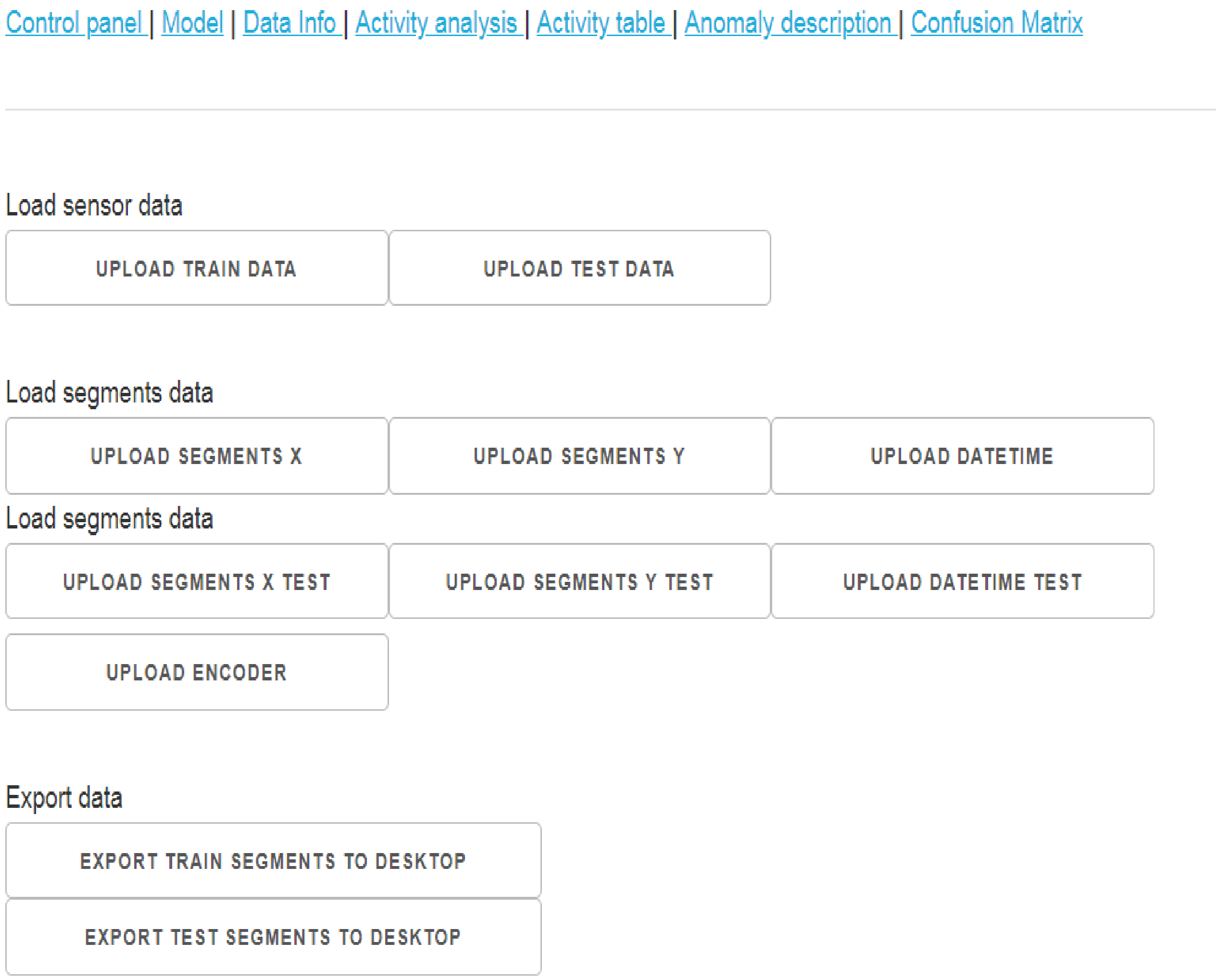


Рисунок 4.2 Контрольна панель програмного застосунку

Після здійснення завантаження вхідних даних користувач має можливість переглянути статистичну інформацію про розподіл сенсорів, активностей та загальну інформацію про вхідні дані.

На рисунку 4.3 зображено загальну інформацію про набір даних. А саме кількість активацій сенсорів та кількість днів.

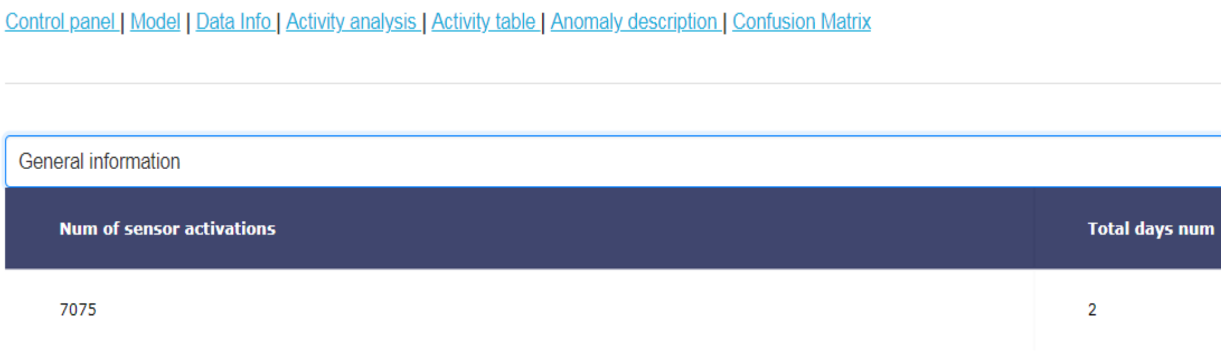


Рисунок 4.3 Загальна статистична інформація

На рисунку 4.4 зображено розподіл сенсорів на даних тренування.

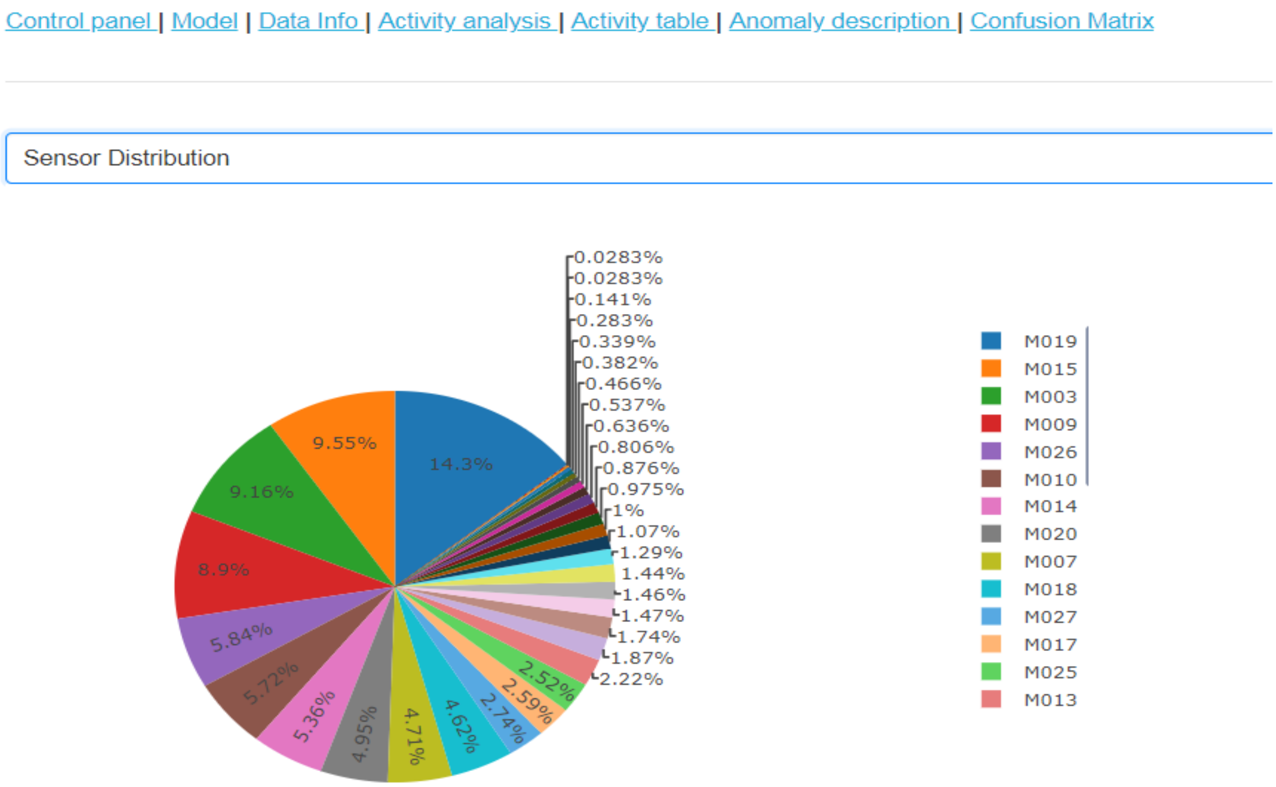


Рисунок 4.4 Розподіл сенсорів на даних тренування

На рисунку 4.5 Зображено розподіл активностей на даних тренування.

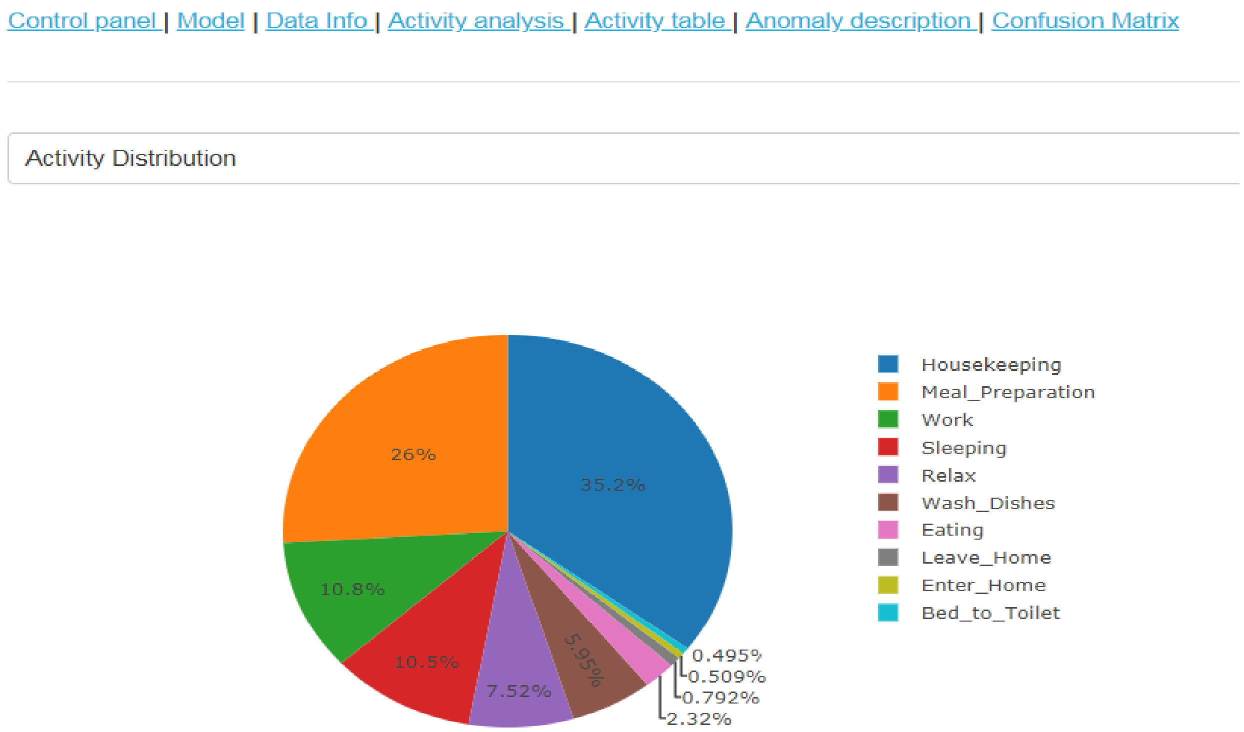


Рисунок 4.5 Розподіл активностей на даних тренування

Також користувач має можливість переглянути матрицю взаємної інформації (Рисунок 4.6).

Control panel | Data Info | Activity analysis | Activity table | Anomaly description | Confusion Matrix

Mutual Matrix Info

	D001	D002	D004	M001	M002	M003	M004	M005	M006	M007	M008	M009	M010	M011	M012	M013	M014	M015
D001	0.01	0.0	0.0	0.0	0.0	0.0	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
D002	0.0	0.03	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.01	0.0
D004	0.0	0.0	0.23	0.0	0.0	0.0	0.01	0.01	0.01	0.01	0.01	0.0	0.0	0.0	0.01	0.01	0.02	0.02
M001	0.0	0.0	0.0	0.02	0.0	0.01	0.02	0.02	0.02	0.02	0.02	0.01	0.01	0.0	0.02	0.02	0.02	0.01
M002	0.0	0.0	0.0	0.0	0.08	0.08	0.0	0.02	0.0	0.08	0.0	0.0	0.0	0.0	0.01	0.0	0.0	0.0
M003	0.0	0.0	0.0	0.01	0.08	0.11	0.01	0.03	0.02	0.11	0.02	0.01	0.01	0.0	0.02	0.01	0.02	0.01
M004	0.01	0.0	0.01	0.02	0.0	0.01	0.08	0.08	0.03	0.08	0.03	0.02	0.02	0.01	0.02	0.03	0.03	0.02
M005	0.01	0.0	0.01	0.02	0.02	0.03	0.08	0.1	0.03	0.1	0.03	0.02	0.03	0.01	0.03	0.03	0.03	0.02
M006	0.01	0.0	0.01	0.02	0.0	0.02	0.03	0.03	0.05	0.04	0.05	0.02	0.03	0.01	0.03	0.04	0.04	0.02
M007	0.01	0.0	0.01	0.02	0.08	0.11	0.08	0.1	0.04	0.18	0.04	0.02	0.03	0.01	0.03	0.03	0.04	0.02
M008	0.01	0.0	0.01	0.02	0.0	0.02	0.03	0.03	0.05	0.04	0.05	0.02	0.03	0.01	0.03	0.04	0.05	0.03
M009	0.01	0.0	0.0	0.01	0.0	0.01	0.02	0.02	0.02	0.02	0.02	0.2	0.1	0.01	0.06	0.19	0.05	0.02
M010	0.01	0.0	0.0	0.01	0.0	0.01	0.02	0.03	0.03	0.03	0.03	0.1	0.11	0.01	0.06	0.1	0.03	0.02
M011	0.01	0.0	0.0	0.0	0.0	0.0	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
M012	0.01	0.0	0.01	0.02	0.01	0.02	0.02	0.03	0.03	0.03	0.03	0.06	0.06	0.01	0.08	0.08	0.04	0.02
M013	0.01	0.0	0.01	0.02	0.0	0.01	0.03	0.03	0.04	0.03	0.04	0.19	0.1	0.01	0.08	0.24	0.11	0.05
M014	0.01	0.01	0.02	0.02	0.0	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.03	0.01	0.04	0.11	0.37	0.19
M015	0.01	0.0	0.02	0.01	0.0	0.01	0.02	0.02	0.02	0.02	0.03	0.02	0.02	0.01	0.02	0.05	0.19	0.22

Рисунок 4.6 Перегляд матриці взаємної інформації

Після завантаження даних тестування та здійснення класифікації користувач має можливість переглянути та порівняти істині активності та класифіковані. Така візуалізація відбувається у вкладці Activity Analysis. Так на рисунку 4.7 зображена візуалізація істинних активностей, а на рисунку 4.8 – візуалізація класифікованих активностей.

Слід зазначити що користувач має можливість візуально порівняти класифіковані активності та реальні, відобразивши істині та класифіковані активності на одному графіку. Порівняння зображено на рисунку 4.9.

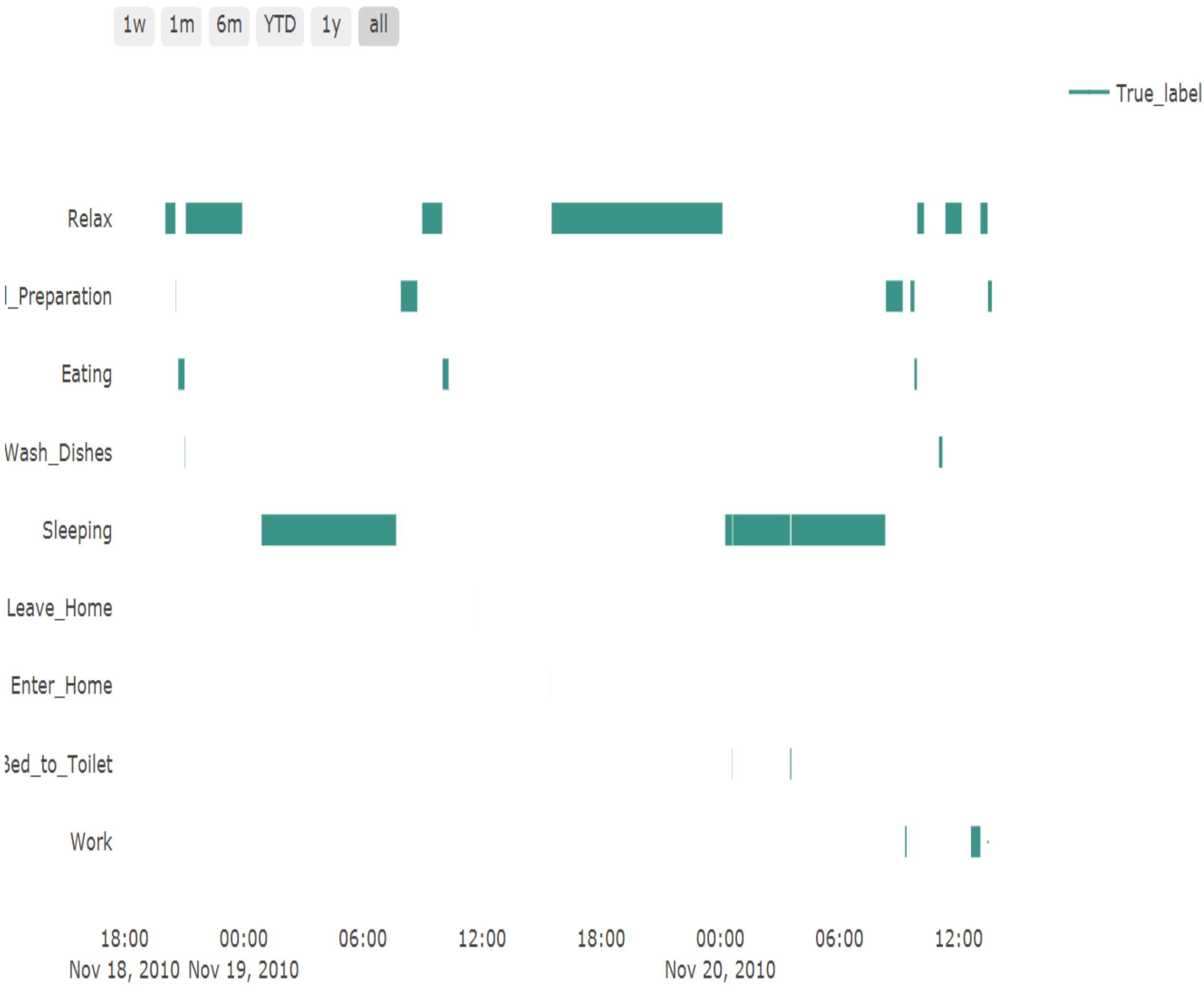


Рисунок 4.7 Візуалізація істинних активностей

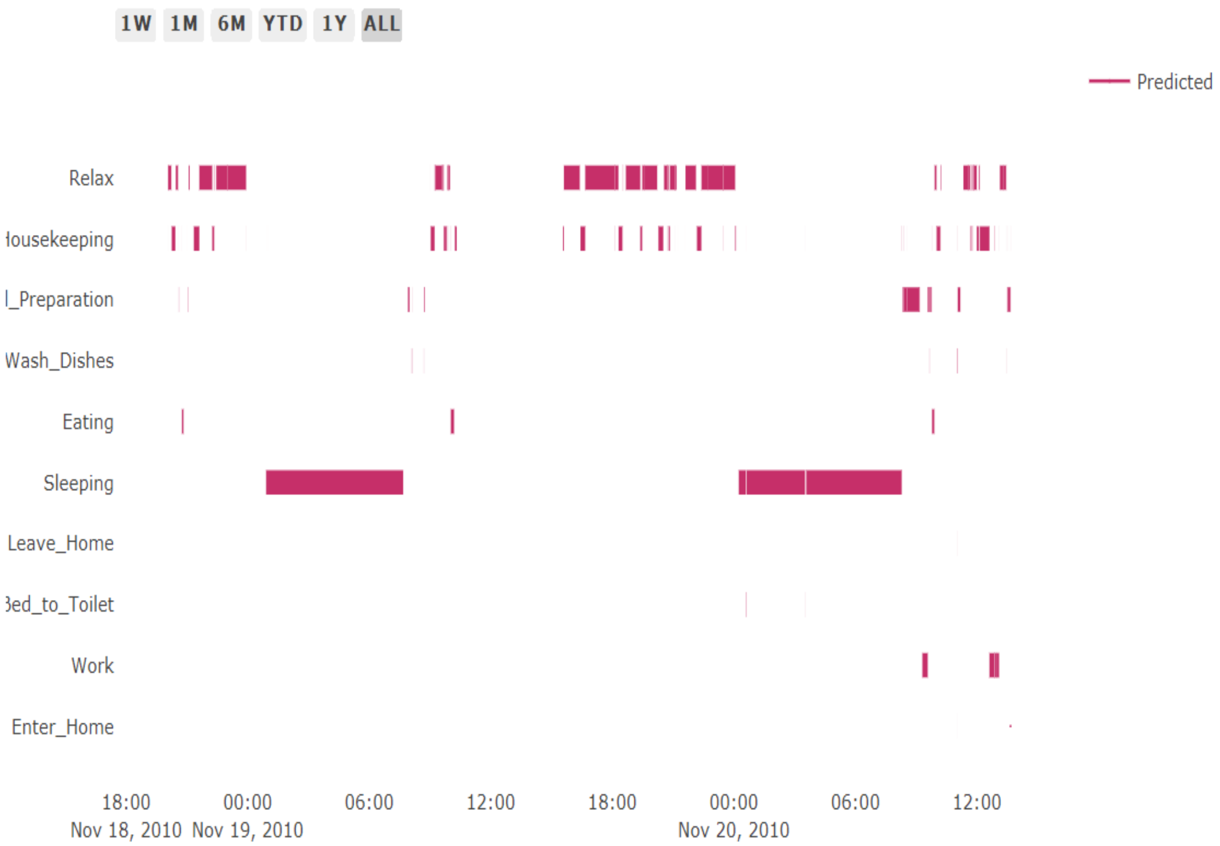


Рисунок 4.8 Візуалізація класифікованих активностей

Рисунок 4.9 Візуалізація порівняння активностей

Також користувач може переглянути класифіковані активності у форматі таблиці. Для цього необхідно перейти до вкладки Activity table. Дана таблиця містить колонки назви активності, її початку і кінця. Таблиця активностей зображена на рисунку 4.10. Відкрив вкладку Confusion Matrix користувач має можливість переглянути матрицю помилок для оцінки ефективності моделі. Матриця помилок зображена на рисунку 4.11.

Control panel | Model | Data Info | Activity analysis | Activity table | Anomaly description | Confusion Matrix

activity	end	start
Relax	2010-11-18 20:04:38	2010-11-18 20:03:40
Housekeeping	2010-11-18 20:04:40	2010-11-18 20:04:39
Relax	2010-11-18 20:13:58	2010-11-18 20:04:41
Housekeeping	2010-11-18 20:25:47	2010-11-18 20:15:12
Relax	2010-11-18 20:33:49	2010-11-18 20:27:18
Housekeeping	2010-11-18 20:35:43	2010-11-18 20:35:39
Meal_Preparation	2010-11-18 20:36:17	2010-11-18 20:35:43
Meal_Preparation	2010-11-18 20:36:49	2010-11-18 20:36:21
Wash_Dishes	2010-11-18 20:42:41	2010-11-18 20:42:38
Housekeeping	2010-11-18 20:42:55	2010-11-18 20:42:43
Eating	2010-11-18 20:49:59	2010-11-18 20:45:19
Housekeeping	2010-11-18 21:01:38	2010-11-18 21:01:36
Meal_Preparation	2010-11-18 21:02:29	2010-11-18 21:02:28
Meal_Preparation	2010-11-18 21:03:26	2010-11-18 21:02:37
Relax	2010-11-18 21:08:26	2010-11-18 21:05:36

Рисунок 4.10 Таблиця активностей

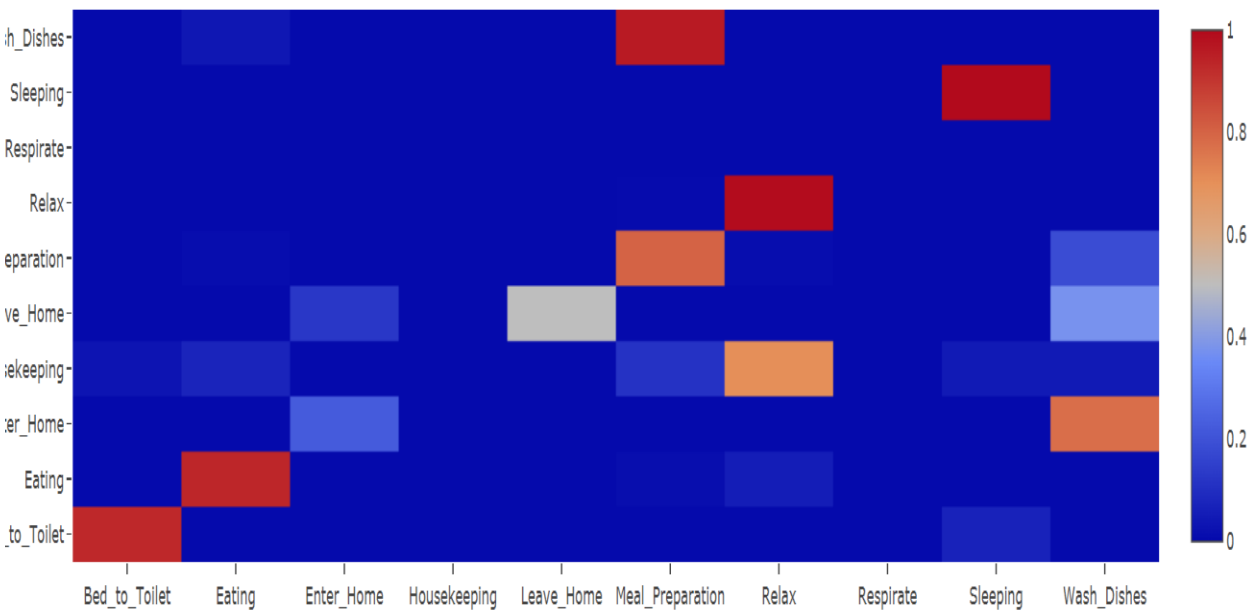


Рисунок 4.11. Матриця помилок

У вкладці Anomaly description користувач відбувається визначення аномальних активностей. Опис файлу та вікна приведено в наступному розділі.

У вкладці Model користувач може вибрати бажану модель, задати параметри для неї, побудувати її та здійснити класифікацію, а також імпорт та експорт моделі. Вкладка Model зображена на рисунку 4.12.

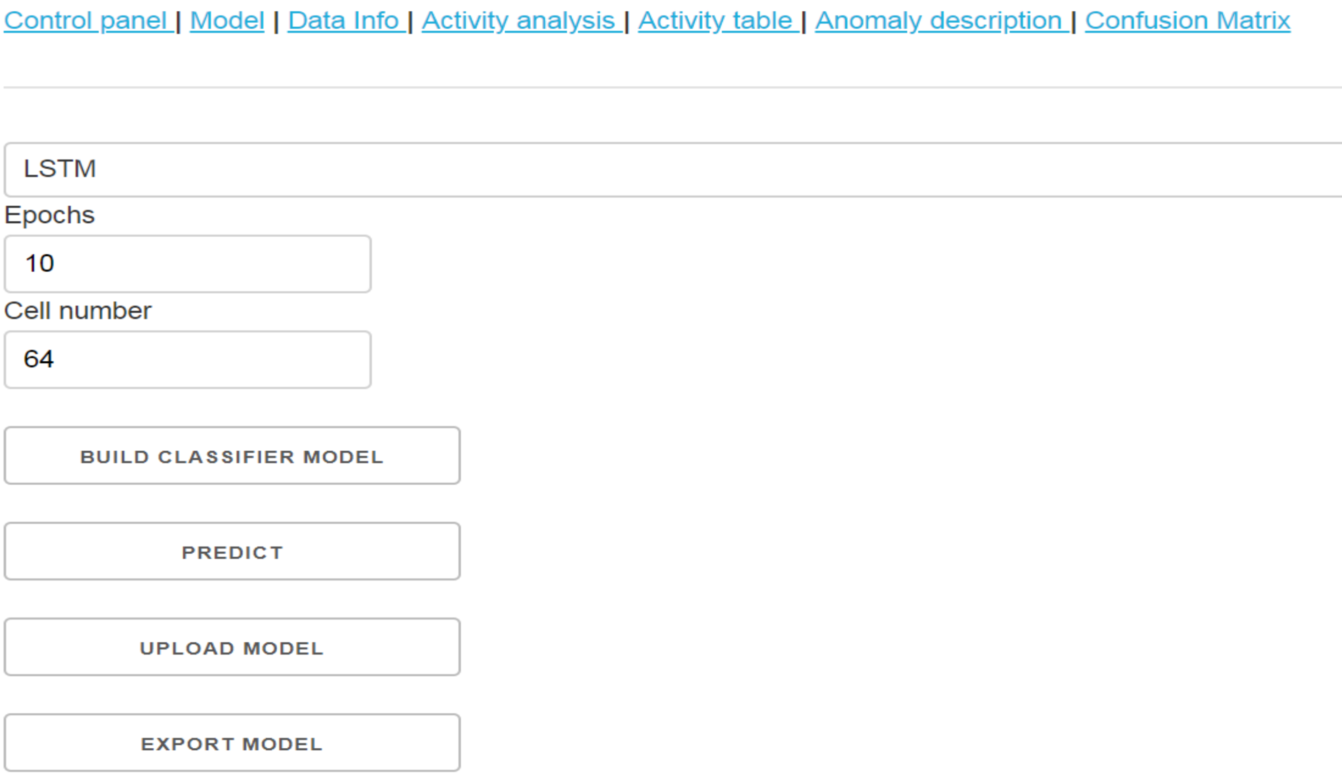


Рисунок 4.12. Вибір моделі класифікації

4.3 Робота з аномальними активностями

Враховуючи той факт, що мешканці будинків мають власні особисті характеристики – необхідно дати користувачу можливість власноруч визначити що характеристики аномальних властивостей.

Користувач має можливість завантажити JSON файл конфігурації для визначення характеристик аномальних активностей або створити його власноруч використовуючи форму представлену в програмі. Так перейшовши до вкладки Anomaly Description користувач може вибрати активність та задати для неї характеристики аномальності (Рисунок 4.13).

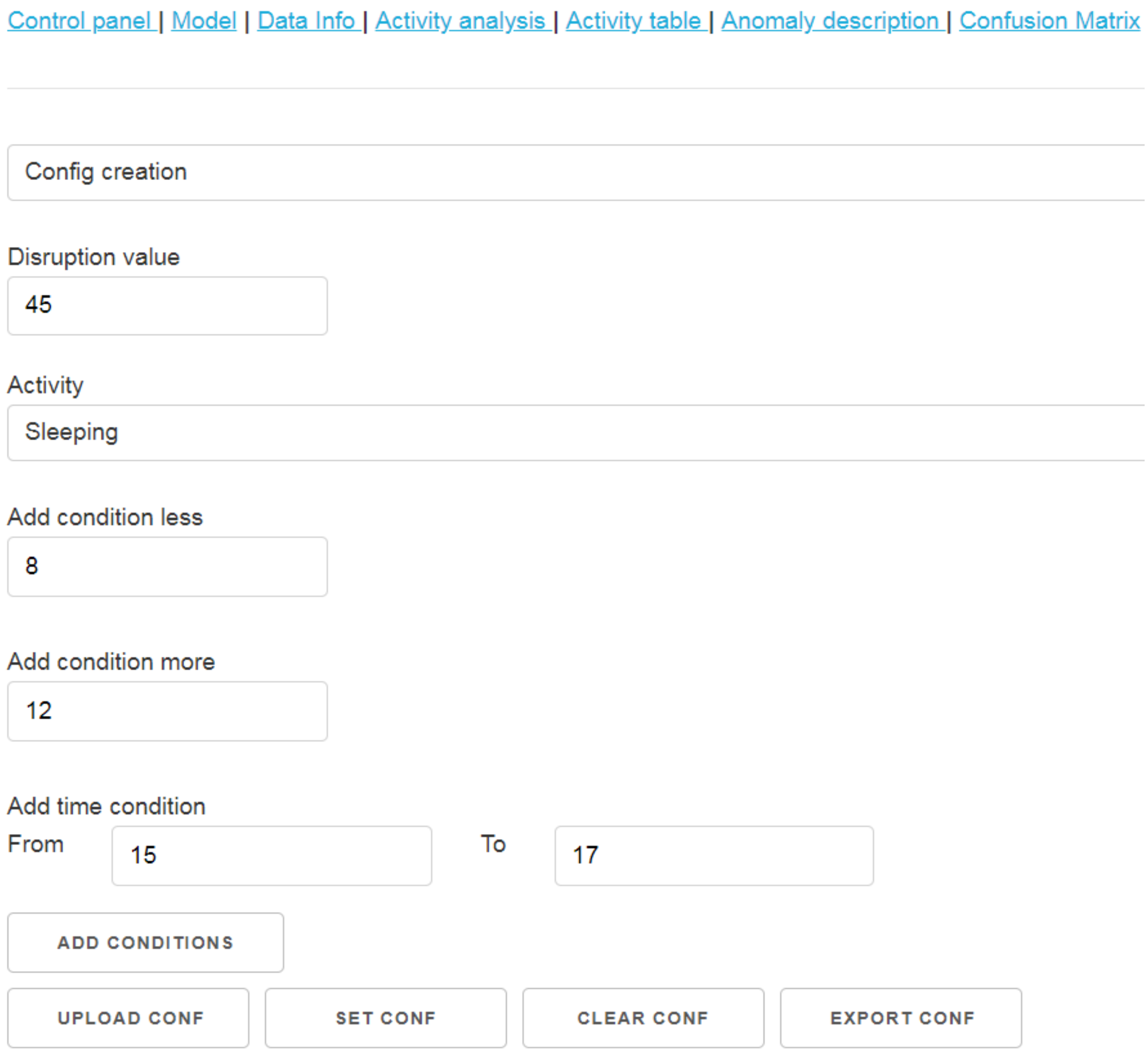


Рисунок 4.13 Вкладка створення конфігурації аномалій

Серед запропонованих характеристик аномальності користувач може визначити аномальну тривалість активності, тобто якщо тривалість активності вище або менше заданої в залежності від вибору користувача – активність вважається аномальною. Також є можливість визначити часовий інтервал в межах якого активність вважається аномальною.

Крім цього користувач має можливість визначити яка кількість часу має пройти між активностями щоб вважати їх окремими. Так якщо мешканець посеред ночі прокинувся а через 15 хвилин ліг спати не доцільно надалі розбити сон на дві окремі активності і визначити їх аномальними.

Після створення конфігурації користувач може здійснити експорт цього файлу для можливості використання його в майбутньому. На рисунку 4.14 представлено експортований файл конфігурації.


```
{
    "time_disrupt_mins":"45",
    "len_less": [
        {
            "activity": "Sleeping",
            "value": "8"
        }
    ],
    "len_more": [
        {
            "activity": "Sleeping",
            "value": "12"
        }
    ],
    "time_interval": [
        {
            "activity": "Sleeping",
            "lower_bound": "15",
            "upper_bound": "17"
        }
    ]
}
```


Рисунок 4.14 Файл конфігурації аномальних активностей

Після визначення характеристик аномальних активностей, або імпорту цих характеристик з файлу, користувач має можливість переглянути аномальні активності серед класифікованих. Перелік знайдених аномальних активностей за заданими характеристиками зображений на рисунку 4.15.

activity	length	start
Sleeping	6.0	2010-11-19 00:58:45

Рисунок 4.15 Перелік аномалій активнсотей

Висновки до розділу 4

На базі четвертого розділу:

1. Встановлені технічні вимоги для можливості роботи користувача з розробленою програмною системою.
2. Приведений опис вікон програмної системи.
3. Описано особливості задання характеристик аномальних активностей.
4. Можна прийти до висновку, що розроблена програмна система надає користувачу ряд різноманітних функцій і дозволить провести повноцінну аналітику щодо вхідних даних, вибрати модель класифікації, задати її характеристики і в випадку необхідності оцінити її точність. Крім того користувач має можливість щодо здійснення аналізу класифікованих активностей на аномалії задавши їх характеристики.

5. РОЗРОБКА СТАРТАП ПРОЕКТУ

В даному розділі аналізується можливість розробки стартап проекту на базі створеної програмної системи аналізу даних. Проект направлений на інтелектуальне використання даних в розумному будинку для створення більш комфортних умов існування в ньому і буде являти собою програмний сервіс. Надалі необхідно здійснити багатофакторний аналіз на базі якого визначити особливості виходу на ринок даної ідеї. Так розроблена система буде являти собою програму яку будуть поставляти розробники шлюзів в розумному будинку. Шлюз – центральний пристрій в розумному будинку який контролює обмін даними між пристроями, ці дані і будуть входом системи. Так як ці розробники вже існують на ринку – певним чином вирішена проблема пошуку початкових клієнтів [53].

5.1 Опис ідеї проекту

Важливо описати яку користь розроблювана система несе різним категоріям користувачів. В таблиці 5.1 описані ідеї проекту.

Таблиця 5.1 Опис ідеї проекту

Зміст ідеї	Напрямки застосування	Сегменти споживачів	Цінність для споживачів
Аналіз даних в розумному будинку.	1. Персоналізація розумного будинку для потреб мешканців.	Масовий ринок(Користувачі розумних будинків).	Більш комфортні умови проживання в розумному будинку.
	2. Застосування у сфері охорони здоров’я для покращеного моніторингу життя пацієнтів.	Нішевий (Медичні та соціальні працівники).	Нові можливості щодо нагляду за життям людей з вадами здоров’я.

Подібний інтелектуальний сервіс є унікальною ідеєю і не має конкурентів. Існуючих гравців на ринку розумних будинків (Google, Amazon і т.д) не слід вважати конкурентами, так як основна цінність, яку дає даний проект відрізняється і не конкурує з тим що пропонують ці компанії.

5.2 Технологічний аудит проекту

Для аналізу технологічної реалізованості проекту проведемо опишемо ідеї та їх можливість реалізації. Технологічний аудит приведений в таблиці 5.2.

Таблиця 5.2 Технологічний аудит проекту

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Система аналізу даних користувачів	Множина алгоритмів машинного навчання та data mining.	Алгоритми самі по собі – відомі, однак необхідно їх об'єднати або доробити.	В даному випадку технологія має бути розробленою.
2	Розробка API та клієнтських програм	Мови програмування.	Технологія існує.	Технологію можна використати для розробки продукту.

Як можна бачити з таблиці для технологічної реалізації проекту необхідно спроектувати систему аналітики та розробки API для клієнтських програм. Це цілком реально, однак необхідно розробити систему аналізу на базі існуючих алгоритмів.

5.3. Аналіз ринкових можливостей стартап-проекту

Важливо, на базі сучасного становища ринку, проаналізувати ринкові можливості та загрози проекту для подальшого спрямування напрямку розвитку [54].

Перш за все необхідно проаналізувати попит на проект. А саме його наявність динаміку та обсяг. Характеристика ринку представлена у таблиці 5.3.

Таблиця 5.3 Попередня характеристика стартап проекту

Показники стану ринку	Характеристика
Загальна потреба в продукції	Існує
Можливі річні обсяги випуску в натуральних показниках	Необмежено
Річні обсяги випуску в вартісних показниках	50000 – 10000$
Динаміка ринку (якісна оцінка)	Зростає
Наявність обмежень для входу	Можлива консервативна позиція розробників розумного будинку щодо нового концепту проекту Проблема конфіденційності даних
Специфічні вимоги до стандартизації та сертифікації	ISO 9126 та ISO 25010
Середня норма рентабельності в галузі (або по ринку), %	82%

На базі представленої таблиці а також на фоні росту ринку розумних будинків і його рентабельності можна сказати що даний ринок є привабливим. Однак враховуючи що складовою системи є дані користувачів слід врахувати проблему конфіденційності. Крім того необхідно довести, що технологія принесе користь так як є новою і раніше не зустрічалась, що може дещо ускладнити вхід на ринок.

Важливим етапом при аналізі ринку є виділення клієнтів та визначення їх особливостей та потреб. Дана характеристика представлена у таблиці 5.4.

Надалі важливо скласти таблиці факторів проекту, які сприяють ринковому впровадженню проекту і факторів, що йому перешкоджають [54]. Так у таблиці 5.5 представлено фактори загроз, а в таблиці 5.6 фактори можливостей.

Таблиця 5.4 Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія	Особливості поведінки споживачів	Вимоги споживачів до товару
Інтелектуальний аналіз даних.	Мешканці розумних будинків.	Користувачі не мають сервісів персоналізації та автоматизації.	Ефективність функціонування системи.
	Люди з вадами здоров'я та похилого віку	Необхідний моніторинг за активностями мешканців з можливістю оперативного втручання.	Можливість класифікації активностей на аномальні.

Таблиця 5.5 Фактори загроз

Фактор	Зміст загрози	Можлива реакція компанії
Поява конкурентів	Після розробки сервісу і його успіхи можлива поява конкурентів які	Постійне проведення досліджень у даній сфері та тримання особливостей розроблених алгоритмів у секреті.
Конфіденційність	Система використовує дані користувача із-за чого можуть бути ініційовані юридичні питання.	Консультація з юридичними спеціалістами на предмет введення в експлуатацію такої системи.
Занепад ринку.	Можливий занепад самого ринку.	Переорієнтація на менш масовий ринок.

Таблиця 5.6. Фактори можливостей

Фактор	Зміст можливості	Можлива реакція компанії
Відсутність конкурентів	На ринку не існує подібних сервісів	Унікальність проекту надає йому привабливість для інвестування.
Постійний ріст ринку розумних будинків.	Можна спостерігати значний ріст ринку на сьогоднішній день.	Здійснення маркетингу для збільшення кількості клієнтів.
Можлива інтеграція з Smart City	Разом з ринком розумних будинків також виникає ринок Smart City. Даний проект може бути інтегрований в концепцію Smart City.	Співпраця з компаніями які працюють з Smart City для збільшення кількості сервісів.
Заключення більш вигідних контрактів з розробниками шлюзів	При розвитку проекту з’явиться можливість щодо заключення більш вигідних контрактів з розробниками шлюзів.	Постійний моніторинг існуючих компаній розробників шлюзів для можливості заключення інших контрактів.

Необхідно проаналізувати конкурентоспроможність проекту для можливості виходу на ринок. Так необхідно оцінити прямих та потенційних конкурентів, товари-замінники та особливості співпраці з постачальниками та клієнтами [54]. Слід зазначити що прямих конкурентів не має і відсутні товари-замінники.

Надалі необхідно описати характери конкуренції з потенційними компаніями-конкурентами (Таблиця 5.7).

Таблиця 5.7. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії)
Тип конкуренції	Тип конкуренції чистий – залежить від розробки потенційних аналогів.	Постійне покращення якості та розробка нових сервісів.
За рівнем конкурентної боротьби	На міжнародному ринку	Необхідно враховувати характеристики різних міжнародних ринків.
За галузевою ознакою	Внутрішньогалузева – лише для власників розумних будинків	Необхідно вивчити потреби саме цього сегменту клієнтів
Конкуренція за видами товарів	В даному випадку – за маркою, можуть бути розроблені аналогічні сервіси в великих компаніях	Постійний аналіз даного ринку
За характером конкурентних переваг	В даному випадку не цінова	Необхідно зосередитись на якості самого сервісу.
За інтенсивністю	Марочна, сфера росте тому велика ймовірність появи конкуренкції	Проводити постійний контроль якості сервісів і пропозиції фірм конкурентів

Надалі аналіз умов конкуренції в галузі за моделлю 5 сил Портера представлено на таблиці 5.8.

Таблиця 5.8. Аналіз конкуренції в галузі за М.Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Клієнти	Постачальники	Товари-замінники
Складові галузі	Відсутні	Потенційні конкуренти мають провести ряд довготривалих досліджень.	Мають велике значення так як саме вони залишать перші відгуки від використання нової системи.	Розробники шлюзів є певним мостом між клієнтами та проектом.	Відсутні.
Висновки	Наразі не існує конкурентів на ринку.	Наявні усі можливості входу на ринок. Потенційні конкуренти можуть бути великими компаніями такі як Google.	Важливо щоб запропонований сервіс задовільнив надії клієнтів на початковому етапі.	Розробники шлюзів можуть впливати початковому етапі.	Немає обмежень

Отже з аналізу зрозуміло, що необхідно прикласти зусилля на початкових етапах проекту із-за відсутності репутації у технології та компанії та відносно слабкою позицією перед компаніями розробниками шлюзів розумного будинку. При цьому при умові відсутності аналогів проект є досить привабливим для інвесторів так як надає унікальну ціннісну пропозицію що певним чином збільшує шанси виходу на ринок. На базі приведеного аналізу необхідно описати фактори конкурентоспроможності (таблиця 5.9.)

Таблиця 5.9. Аналіз конкуренції в галузі за М.Портером

Фактор конкурентоспроможності	Обґрунтування
Відсутність конкурентів	На сьогоднішній день не існує конкурентів які б надавали подібний сервіс
Декілька можливостей щодо застосування	Проект надає можливості використання як для автоматизації так і у сфері охорони здоров'я
Концепція виходу на ринок	Наявність клієнтів на початковому етапі завдяки співпраці з фірмами – розробниками шлюзів
Ріст якості з часом	Враховуючи що система накопичує дані – її ефективність буде рости з часом
Поява нових сервісів на базі технології	Так на базі даних мешканців надалі можна буде створювати нові сервіси або здійснювати на базі даних інтеграції з можливостями інших подібних компаній
Складність виходу на ринок для інших компаній	Так потенційні компанії конкуренти мають провести ряд довготривалих досліджень перед виходом на ринок і ще певний час для отримання репетації

На базі виділених загроз та можливостей проведемо SWOT аналіз проекту (таблиця 5.10).

Таблиця 5.10 SWOT аналіз проекту

Сильні сторони (S):	Слабкі сторони (W):
– Постійний ріст якості сервісу з ростом кількості клієнтів – Розробники шлюзів надають доступ до клієнтської бази – Декілька різних сегментів користувачів	– Потрібно зібрати певний масив даних від мешканців перед тим як сервіс буде функціонувати – Необхідно тримати в секреті результати досліджень – Час на проведення досліджень

Таблиця 5.10 (Продовження)

Можливості (О):	Загрози (Т):
– Постійний ріст ринку розумних домів. – Відсутність конкурентів. – Заключення більш вигідних контрактів при розвитку проекту. – Інтеграція даного сервісу з проектами Smart City.	– Можлива поява конкурентів. – Конфіденційність даних мешканців. – Занепад ринку розумних будинків.

На базі проведеного SWOT-аналізу необхідно розробити альтернативу ринкової поведінки для виведення стартап-проекту на ринок [54]. Необхідно визначити оптимальний час їх ринкової реалізації. Альтернативи ринкового впровадження стартап-проекту приведені у таблиці 5.11. Так інвестиції можливі з двох джерел: інвесторів та через збирання кошти на краудфандінгових платформах.

Таблиця 5.11. Альтернативи ринкового впровадження стартап-проекту

Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
Пошук інвесторів	80%	5-6 місяців
Використання платформ краудфандінгу	65%	Більше року

Більш доцільним в даному випадку є пошук інвесторів із-за менших строків реалізації і більшого шансу отримати ресурси.

5.4 Розроблення ринкової стратегії проекту

Для розроблення ринкової стратегії для початку необхідно описати цільові групи потенційних споживачів. Цільові групи описані в таблиці 5.11.

Таблиця 5.11. Вибір цільових груп потенційних споживачів

Опис цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в сегменті	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
Мешканці розумного будинку	Потребують	Попит наявний	Конкуренти відсутні	Просто
Люди похилого віку та люди з вадами здоров'я	Потребують	Попит наявний	Конкуренти відсутні	Помірно

Важливою особливістю системи є те що технічно розробка системи може охопити обидві цільові групи. Враховуючи наявний попит та відсутність конкурентів надалі доцільно працювати зі всіма групами.

Для можливості роботи в обраних сегментах ринку сформуємо стратегію розвитку, яка представлена у таблиці 5.12.

Таблиця 5.12. Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Пошук інвесторів	Стратегія диференціації	Охоплення декілька різних сегментів користувачів.	Стратегія диференціації

На базі представлених вимог споживачів з обраних сегментів а також в залежності від обраної базової стратегії розвитку необхідно розробити стратегію позиціонування. Ця стратегія представлена на таблиці 5.13.

Таблиця 5.13. Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
Не досить висока ціна, ефективність системи.	Стратегія диференціації	Конкуренти відсутні.	– Унікальність; – Ефективність; – Доступність;

Тобто найбільш оптимальним для даного стартапу є стратегія диференціації виконаний враховуючи декілька сегментів користувачів.

5.5 Розроблення маркетингової програми стартап-проекту

Перш за все необхідно сформувати маркетингову концепцію товару. Для цього у таблиці 5.14 приведені ключові переваги концепції потенційного товару.

Таблиця 5.14. Визначення ключових переваг концепції потенційного товару

Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
Персоналізаця розумного будинку	Покращення умов проживання в рамках розумного будинку.	Конкуренти відсутні.

Таблиця 5.14 (Продовження)

Моніторинг життя мешканців	Можливість моніторингу активності мешканців для виявлення аномалій в активностях і можливості оперативного втручання.	Конкуренти відсутні.

Надалі необхідно розробити трирівневу маркетингову модель товару (таблиця 5.14).

Таблиця 5.14 Опис трьох рівнів моделі товару

Рівні товару	**Сутність та складові**
I. Товар за задумом	Система аналізу даних мешканців розумного будинку для подальшого використання для персоналізації життя мешканців розумного будинку і у медичні сфері для моніторингу активностей.
II. Товар у реальному виконанні	Розроблений проект надає можливість аналізу даних мешканців будинку на активність, виділення аномальних активностей, надає зручний графічний інтерфейс. Може бути інтегрований розробниками шлюзів для аналізу даних в реальному часі. Основні характеристики: можливість надійного функціонування у реальному часі, ефективність з точки зору високої точності аналізу даних,
	Відповідає вимогам ISO 9126 та ISO 25010
	Марка: Smart Home Inhabitant Data Analysis
III. Товар із підкріпленням	Загальна інформація про проект представлена у ряді публікації відповідних конференцій.

На базі рівнів доходів власників розумних будинків важливо визначити верхню та нижню межі встановлення ціни (таблиця 5.15).

Таблиця 5.15. Визначення меж встановлення ціни

Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни
–	1000 – 3500 $	30-65 $

Важливим кроком є визначення системи збуту, тобто використовується власна чи залучена система збуту, глибина збуту і вибір посередників (таблиця 5.16).

Таблиця 5.16. Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Клієнти мають мати можливість безкоштовного випробовування системи перед придбанням.	Створення мосту між даними з шлюзу і розробленою системою	Однорівневий : через розробників шлюзів	В якості місячної підписки на сервіс.

В даному випадку специфіка наведена для обох сегментів клієнтів так як основні наведені характеристики з точки зори збуту в них однакові. Останньою частиною маркетингової програми є розробка, на базі вибраної стратегії позиціонування, концепції маркетингових комунікацій, яка представлена у таблиці 5.17.

Таблиця 5.17. Концепція маркетингових комунікацій

No п/п	Специфіка поведінки цільових клієнтів	Канали комунікації̈, якими користуються цільові клієнти	Ключові позиції̈, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Бажання впевнитись в ефективності системі перед придбанням.	Через розробників шлюзів	Ефективність і функціонування у реальному часі.	Довести що система зробить життя в будинку значно зручнішим.	Нові стандарти в умовах персоналізації розумних будинків.
2	Бажання впевнитись в ефективності системі перед придбанням.	Через розробників шлюзів	Можливість моніторингу та аналіз аномальних активностей.	Довести що моніторинг змінить сферу охорони здоров’я.	Тепер стало можливим віддалений моніторинг за життям.

На базі аналізу можна зробити висновок сервіс буде розповсюджуватись як платна підписка на певний час з можливістю безкоштовного тестування для визначення ефективності. Ціна встановлена з огляду на середній дохід потенційних користувачів.

Висновок до розділу 5

В рамках розділу проаналізовано особливості створення стартап проекту на базі темі диплому. На ринку не існує сервісів які б надавали подібні можливості щодо здійснення аналізу що є великим плюсом при виході на ринок. Враховуючи що ідея реалізації проекту базується на співпраці з розробниками шлюзів – частково вирішено проблема пошуку перших клієнтів.

Так якщо врахувати постійний ріст ринку та відсутність конкурентів – можна прийти до висновку, що вихід на ринок цілком можливий. Для цього на початкових етапах функціонування необхідно довести клієнтам що технологія буде їм корисною. Даний проект має дві потенційні групи клієнтів: мешканців розумних будинків та людей з вадами здоров'я. Технічні особливості проекту надають можливість роботи одночасно з двома сегментами.

При аналізі можливих альтернатив виходу на ринок – було обрано вийти через інвестиції із-за шансу та менших строків порівняно з іншим варіантом. На базі проведеного аналізу можна зробити висновок що подальша імплементація проекту є можливою.

ВИСНОВКИ

На базі виконання магістерської дисертації можна зробити наступні висновки:

1. Проаналізовані існуючі в дослідженнях методи та підходи щодо здійснення аналізу активності людини в розумному будинку у реальному часі використовуючи дані бінарних сенсорів руху. Запропоновані в дослідженнях методи стали базою для подальшого удосконалення. Описані особливості наборів даних, на які необхідно звернути увагу при удосконаленні методу.
2. Удосконалено метод розпізнавання активності на базі даних бінарних сенсорів руху у реальному часі за рахунок використання моделі випадкового лісу в якості класифікатора і моделі Латентного Розміщення Діріхле для включення в сегмент інформацію про розподіл активностей, що призвело до зростання точності класифікації. У середньому можна відмітити ріст точності класифікації на 20%. Також описані та проаналізовані результати експериментів з рядом інших підходів, а також точність класифікації інших моделей.
3. Розроблена програмна система, яка використовуючи удосконалену модель може здійснювати візуалізацію результатів розпізнавання і аналізу активностей на аномалії. Для розробки програмної системи були обрані засоби та бібліотеки з врахуванням результативної ефективності. Описана програмна структура розробленої системи використовуючи діаграми пакетів та класів, сценарії використання системи на базі діаграми активностей, а також наведено методику роботи користувача з системою.
4. Враховуючи можливість функціонування у реальному часі – розроблена система може бути використана для здійснення автоматизації розумного будинку. Іншим важливим застосунком системи є моніторинг за життям людей з вадами здоров’я. Для такого застосунку у розробленій системі передбачена можливість визначення аномальних активностей.
5. Розроблений стартап проект який включає в себе багатофакторний аналіз можливості виходу проекту на ринок. У результаті аналізу ціннісної пропозиції,

яку несе система, можливості технічного впровадження, потенційних сегментів клієнтів та їх потреб до програмної системи, конкурентного становища на ринку, можливих потенційних конкурентів, сильних та слабких сторін проекту, можливих альтернатив ринкового впровадження та особливостей здійснення маркетингової компанії було зроблено висновок що є доцільним подальша імплементація проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Smart home - Statistics & Facts [Електронний ресурс] — Режим доступу: https://www.statista.com/topics/2430/smart-homes/
2. Tapia E. Activity Recognition in the Home Using Simple and Ubiquitous Sensors / E. Tapia, I. Stephen, K. Larson // International Conference on Pervasive Computing / E. Tapia, I. Stephen, K. Larson., 2004. – С. 158–175.
3. Sensor-Based Activity Recognition / [C. Liming, H. Jesse, C. Diane та ін.] // IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS / [C. Liming, H. Jesse, C. Diane та ін.]., 2012. – С. 790–808.
4. Abdallah Z. Activity Recognition with Evolving Data Streams: A Review / Z. Abdallah, M. Medhat // ACM Computing Surveys (CSUR) / Z. Abdallah, M. Medhat., 2018. – С. 1–36.
5. Mehr H. Resident activity recognition in smart homes by using artificial neural networks / H. Mehr, P. Huseyin, A. Cetin // International Istanbul Smart Grid Congress and Fair (ICSG) / H. Mehr, P. Huseyin, A. Cetin., 2016.
6. Two-Layer Hidden Markov Model for Human Activity Recognition in Home Environments / K.Humayun, R. Hoque, T. Keshav, S. Yang // International Journal of Distributed Sensor Networks / K.Humayun, R. Hoque, T. Keshav, S. Yang., 2016.
7. Serge T. User Activity Recognition in Smart Homes Using Pattern Clustering Applied to Temporal ANN Algorithm [Електронний ресурс] / T. Serge, Y. Yoo, M. Bourobou // Open Access Journals Sensors. – 2015. – Режим доступу до ресурсу: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4481973/
8. Fahad L. Activity Recognition in Smart Homes Using Clustering Based Classification / L. Fahad, S. Tahir, M. Rajarajan // 22nd International Conference on Pattern Recognition / L. Fahad, S. Tahir, M. Rajarajan., 2014.

9. Kasteren T. Hierarchical Activity Recognition Using Automatically Clustered Actions / T. Kasteren, G. Englebienne, B. Krose // International Joint Conference on Ambient Intelligence / T. Kasteren, G. Englebienne, B. Krose., 2011. – С. 82–91.
10. Park J. IEEE 4th World Forum on Internet of Things (WF-IoT) / J. Park, J. Kiyoung, Y. Sung-Bong // Deep neural networks for activity recognition with multi-sensor data in a smart home / J. Park, J. Kiyoung, Y. Sung-Bong., 2018.
11. Cook D. Activity recognition on streaming sensor data / D. Cook, N. Krishnan // Pervasive and Mobile Computing / D. Cook, N. Krishnan., 2014. – С. 138–154.
12. Nawel Y. Feature extraction for human activity recognition on streaming data / Y. Nawel, F. Belkacem, A. Fleury // International Symposium on Innovations in Intelligent SysTems and Applications (INISTA) / Y. Nawel, F. Belkacem, A. Fleury., 2015.
13. Nawel Y. Towards improving feature extraction and classification for activity recognition on streaming data / Y. Nawel, F. Belkacem, A. Fleury // Journal of Ambient Intelligence and Humanized Computing / Y. Nawel, F. Belkacem, A. Fleury., 2016. – С. 177–189.
14. An experimental study on new features for activity of daily living recognition / D.Ferretti, E. Principi, S. Squartini, L. Mandolini // International Joint Conference on Neural Networks (IJCNN) / D.Ferretti, E. Principi, S. Squartini, L. Mandolini., 2016.
15. Machalot F. Activity Recognition in Sensor Data Streams for Active and Assisted Living Environments / F. Machalot, A. Haj Mosa, A. Mouhannad // IEEE Transactions on Circuits and Systems for Video Technology / F. Machalot, A. Haj Mosa, A. Mouhannad., 2017. – С. 2933 – 2945.
16. Iterative Design of Visual Analytics for a Clinician-in-the-loop Smart Home. / [A. Ghods, K. Caffrey, D. Cook та ін.] // IEEE Journal of Biomedical and Health Informatics / [A. Ghods, K. Caffrey, D. Cook та ін.]., 2018.
17. Kirienko, A. Human Behaviour Analysis in Context of Smart Environment

Automation / A. Kirienko,, I. Soloviev, // Computer tools in education / A. Kirienko,, I. Soloviev,. – St. Petersburg, 2017. – С. 15–29

18. Cook D. Collecting and disseminating smart home sensor data in the CASAS project / D. Cook, M. Schmitter-Edgecombe, C. A // Proceedings of the CHI Workshop on Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research / D. Cook, M. Schmitter-Edgecombe, C. A., 2009.
19. WSU CASAS Datasets [Електронний ресурс] — Режим доступу: http://casas.wsu.edu/datasets/
20. CASAS: A smart home in a box / D.Cook, A. Crandall, B. Thomas, N. Krishnan // IEEE Computer /D.Cook, A.Crandall, B. Thomas, N. Krishnan., 2013. – С.26–33.
21. Тарнавський Ю. А. Інтелектуальний аналіз даних в умовах розумного будинку / Ю. А. Тарнавський, М. С. Малишев // СТАЛИЙ РОЗВИТОК — XXI СТОЛІТТЯ: УПРАВЛІННЯ, ТЕХНОЛОГІЇ, МОДЕЛІ / Ю. А. Тарнавський, М. С. Малишев., 2018. – С. 448–450.
22. Voulgaris Z. Extensions of the k nearest neighbour methods for classification problems / Z. Voulgaris, G. Magoulas // The 26th IASTED conference on artificial intelligence and applications / Z. Voulgaris, G. Magoulas., 2008. – С. 23–28.
23. Yuxuan L. Improving k Nearest Neighbor with Exemplar Generalization for Imbalanced Classification / L. Yuxuan, Z. Xiuzhen // Pacific-Asia Conference on Knowledge Discovery and Data Mining / L. Yuxuan, Z. Xiuzhen.. – С. 321–332.
24. Novakovic J. Using Information Gain Attribute Evaluation to Classify Sonar Targets / Jasmina Novakovic // 17th Telecommun. Forum TELFOR / Jasmina Novakovic., 2009. – С. 24–26.
25. Sokolova M. AI 2006: Advances in Artificial Intelligence / M. Sokolova, N. Japkowicz, S. Szpakowicz // Beyond Accuracy, F-score and ROC: a Family of Discriminant Measures for Performance Evaluation / M. Sokolova, N. Japkowicz, S. Szpakowicz., 2006. – С. 1015–1021.
26. Bishop C. Pattern Recognition and Machine Learning / Christopher Bishop. – New York, 2006. – 738 с.

27. Breinman L. Classification and regression trees / L. Breinman, J. Friedman, R. Olshen // Wadsworth & Brooks/Cole Advanced Books & Software / L. Breinman, J. Friedman, R. Olshen., 1984.
28. Belgiu M. Random forest in remote sensing: A review of applications and future directions / M. Belgiu, L. Dragut // ISPRS Journal of Photogrammetry and Remote Sensing / M. Belgiu, L. Dragut., 2016. – С. 24–31.
29. Goodfellow I. Deep Learning (Adaptive Computation and Machine Learning) / I. Goodfellow, B. Yoshua, A. Courville., 2016. – 775 с.
30. Schmidhuber J. Deep learning in neural networks: An overview / Jurgen Schmidhuber // Neural Networks / Jurgen Schmidhuber., 2015. – С. 85–117.
31. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. // Journal of Machine Learning Research 15 / , 2014. – С. 1929–1958.
32. A Novel Connectionist System for Unconstrained Handwriting Recognition / [A. Graves, M. Liwicki, S. Fernandez та ін.] // IEEE Transactions on Pattern Analysis and Machine Intelligence / [A. Graves, M. Liwicki, S. Fernandez та ін.]., 2009. – С. 855–868.
33. LSTM: A Search Space Odyssey / [K. Greff, K. Srivastava, J. Koutnik та ін.] // TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS 1 / [K. Greff, K. Srivastava, J. Koutnik та ін.]., 2015. – С. 2222 – 2232.
34. Vapnik V. Support-vector networks / V. Vapnik, C. Corinna // Machine learning / V. Vapnik, C. Corinna., 1995. – С. 273–297.
35. Patle A. SVM kernel functions for classification / A. Patle, C. Deepak // International Conference on Advances in Technology and Engineering (ICATE) / A. Patle, C. Deepak., 2013.
36. Chih-Wei H. A Comparison of Methods for Multi-class Support Vector Machines / H. Chih-Wei, L. d Chih-Jen // IEEE Transactions on Neural Networks / H. Chih-Wei, L. d Chih-Jen., 2002. – С. 415 – 425.
37. Blei D. Probabilistic Topic Models / David Blei // Communications of the ACM / David Blei., 2012. – С. 77–84.

38. Blei D. Latent Dirichlet Allocation / D. Blei, A. Ng, M. Jordan // Journal of Machine Learning Research / D. Blei, A. Ng, M. Jordan., 2003. – С. 993–1022.
39. Steyvers M. Probabilistic Topic Models / M. Steyvers, T. Griffiths // Handbook of latent semantic analysis / M. Steyvers, T. Griffiths., 2007. – С. 427–448.
40. Jeffreys H. Scientific Inference / Harold Jeffreys., 2011. – 282 с.
41. Kingma D. Adam: A Method for Stochastic Optimization / D. Kingma, J. Ba // International Conference on Learning Representations / D. Kingma, J. Ba., 2015.
42. Jedrzejowicz P. Stacked Ensemble Models for Improved Prediction Accuracy / P. Jedrzejowicz, I. Czarnowsk // IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA) / P. Jedrzejowicz, I. Czarnowsk.. – (2017).
43. Beckmann M. A KNN Undersampling Approach for Data Balancing / M. Beckmann, N. Ebecken // Journal of Intelligent Learning Systems and Applications / M. Beckmann, N. Ebecken., 2015. – С. 104–116.
44. McKinney W. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython / Wes McKinney., 2017. – 550 с.
45. Chen D. Pandas for Everyone: Python Data Analysis / Daniel Chen., 2017. – 406 с.
46. Garreta R. Learning scikit-learn: Machine Learning in Python / R. Garreta, G. Moncecchi., 2013. – 118 с.
47. Chih-Chung C. LIBSVM: A Library for Support Vector Machines / C. Chih-Chung, L. Chih-Jen., 2001. – 39 с.
48. Gulli A. Deep Learning with Keras: Implementing deep learning models and neural networks with the power of Python / Antonio Gulli., 2017. – 318 с.
49. Grinberg M. Flask Web Development: Developing Web Applications with Python / Miguel Grinberg., 2014. – 258 с.
50. Gemino A. Use case diagrams in support of use case modeling: Deriving understanding from the picture / A. Gemino, D. Parker // Journal of Database Management / A. Gemino, D. Parker., 2009. – С. 1–24.
51. Fowler M. UML Distilled Second Edition A Brief Guide to the Standard Object

Modeling Language / M. Fowler, K. Scott., 1999. – 224 с.

52. Yan Y. Hands-On Data Science with Anaconda / Y. Yan, J. Yan., 2018. – 364 с.
53. Малишев М. С. ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ В УМОВАХ "РОЗУМНОГО БУДИНКУ" / М. С. Малишев, Ю. А. Тарнавський // СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ / М. С. Малишев, Ю. А. Тарнавський., 2018. – С. 195.
54. Розроблення стартап-проекту [Електронний ресурс] : Методичні рекомендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей / За заг. ред. О.А. Гавриша. – Київ : НТУУ «КПІ», 2016. – 28 с.

ДОДАТОК А

Апробації

Інтелектуальний аналіз даних в умовах розумного будинку

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ ТР32265_18М

Аркушів 4

2018

1. Малишев М. С. ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ В УМОВАХ "РОЗУМНОГО БУДИНКУ" / М. С. Малишев, Ю. А. Тарнавський // СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ / М. С. Малишев, Ю. А. Тарнавський., 2018. – С. 195.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів
м. Київ, 24-27 квітня 2018 року,

ТОМ 2

Київ- 2018

1

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ В УМОВАХ "РОЗУМНОГО БУДИНКУ"

На сьогоднішній день можна спостерігати посилений інтерес до ринку розумних будинків зі сторони великих компаній, таких як Google, Samsung, Amazon та інших. Передбачається що ринок розумних будинків перевищить 40 мільярдів доларів до 2020 року [1].

Розумним будинком вважають будівлю що містить у собі серію пристроїв, які фактично здатні виконувати дії і вирішувати певні повсякденні завдання без участі людини. Розумні будинки, як правило, складаються з сенсорів, пристроїв, що накопичують дані про навколишнє середовище(температуру, вологість або рух людини), актуатори, які безпосередньо виконують команди(розумні вимикачі, розумні розетки, розумні клапани для труб, сирени, клімат-контролери) та контролер – керуючий пристрій, що з'єднує всі елементи системи один з одним і зв'язує її з зовнішнім світом.

Дані у сенсорах представлені у вигляді часових рядів, тому для їх аналізу використовуються такі моделі, як ARIMA, рухоме середнє і інші. Однак у багатьох дослідженнях використовують як класичні методи машинного навчання, такі як метод опорних векторів, прихована марковська модель так і нейронні мережі [2].

Проблема сучасних досліджень в цій сфері в тому, що вони націлені лише на один набір даних, який є частиною дослідження. В багатьох таких дослідженнях не використовується інформація про саму інфраструктуру розумного будинку. Також серед моделей – використовують лише найбільш прості нейронні мережі, хоча часто об’єми даних дозволяють використати більш глибокі мережі.

Нами проаналізовано декілька відкритих наборів даних таких REFIT: SMART HOMES та WSU CASAS та засобів генерації цих даних таких як Open Smart Home Simulator та розроблено серію підходів по визначенню корисних часових шаблонів у межах різних інтервалів(тиждень, місяць, рік), які відображають закономірності у середовищі існування мешканців будинку і в їх поведінці. Як приклад однією з таких закономірностей є визначені дії людини на базі даних сенсорів руху.

Перелік посилань:

1. Smart home - Statistics & Facts [Електронний ресурс] — Режим доступу: https://www.statista.com/topics/2430/smart-homes/

2. Data Science for Internet of Things (IoT): Ten Differences From Traditional Data Science [Електронний ресурс] — Режим доступу: https://www.kdnuggets.com/2016/09/data-science-iot-10-differences.html

ДОДАТОК Б

Публікації

Інтелектуальний аналіз даних в умовах розумного будинку

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ ТР32265_18М

Аркушів 7

2018

Тарнавський Ю. А. Інтелектуальний аналіз даних в умовах розумного будинку / Ю.А. Тарнавський, М. С. Малишев // СТАЛИЙ РОЗВИТОК — XXI СТОЛІТТЯ: УПРАВЛІННЯ, ТЕХНОЛОГІЇ, МОДЕЛІ / Ю. А. Тарнавський, М. С. Малишев., 2018. – С. 448–450.

Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Національний університет "Києво-Могилянська академія"
Вища економіко-гуманітарна школа (Польща)

СТАЛИЙ РОЗВИТОК — XXI СТОЛІТТЯ: УПРАВЛІННЯ, ТЕХНОЛОГІЇ, МОДЕЛІ

Дискусії 2018

Колективна монографія

Київ, Україна
2018

Інтелектуальний аналіз даних в умовах розумного будинку

Актуальність. На сьогоднішній день можна спостерігати посилений інтерес до ринку розумних будинків зі сторони великих компаній, таких як Google, Samsung, Amazon та інших. Передбачається що ринок розумних будинків перевищить 40 мільярдів доларів до 2020 року[1]. Однак, не зважаючи на ріст ринку, розумні будинки мають певну кількість обмежень серед яких неможливість аналізувати поведінку мешканців будинку. На даний момент відсутні рішення та підходи які б були в змозі вирішити дану проблему. Саме тому було поставлено задачу розробити програмну систему аналізу активності мешканців будинку. Подібна система може бути використана у декількох напрямках, серед яких автоматизація та охорона здоров'я для моніторингу життя людей з вадами здоров’я і людей похилого віку.

Основна частина. Інформацію про оточуюче середовище розумний будинок отримує з сенсорів. Цими сенсорами можуть бути як прості сенсори руху так і камери. Для аналізу було вибрано саме бінарні сенсори руху по причині невеликої вартості самих сенсорів так і вартості їх зберігання. Крім того, інформація зібрана з сенсорів не є настільки критичною для потенційних мешканців будинку у порівнянні наприклад з інформацією з камери. Так як у відповідність показникам сенсорів має бути поставлена певна активність – було вирішено використати модель машинного навчання для здійснення класифікації. Набори даних для тренування і тестування моделей були взяті з проекту CASAS, метою якого є дослідження можливості використання методик машинного навчання в рамках розумного будинку[2]. Приклад даних представлено на рисунку 1.

2010-11-04	05:40:51.303739	M004	ON	Bed_to_Toilet	begin
2010-11-04	05:40:52.342105	M005	OFF		
2010-11-04	05:40:57.176409	M007	OFF		
2010-11-04	05:40:57.941486	M004	OFF		
2010-11-04	05:43:24.021475	M004	ON		
2010-11-04	05:43:26.273181	M004	OFF		
2010-11-04	05:43:26.345503	M007	ON		
2010-11-04	05:43:26.793102	M004	ON		
2010-11-04	05:43:27.195347	M007	OFF		
2010-11-04	05:43:27.787437	M007	ON		
2010-11-04	05:43:29.711796	M005	ON		
2010-11-04	05:43:30.279021	M004	OFF	Bed_to_Toilet	end
2010-11-04	05:43:45.7324	M003	ON	Sleeping	begin
2010-11-04	05:43:52.044085	M003	OFF		
2010-11-04	05:43:53.185335	M002	ON		
2010-11-04	05:43:53.253809	M003	ON		
2010-11-04	05:43:59.493281	M002	OFF		
2010-11-04	05:44:04.048766	M003	OFF		
2010-11-04	05:44:06.14204	M003	ON		
2010-11-04	05:44:11.229146	M003	OFF		

Рисунок 1. Частина набору даних проекту CASAS

Як можна бачити набір даних являє собою послідовно зібрані показники сенсорів. Кожен рядок містить інформацію про час активації сенсору, його унікальний ідентифікатор і значення. Крім того послідовність сенсорів розмічена активностями, тобто кожній активації сенсорів поставлена у відповідність конкретна

[1] Smart home - Statistics & Facts [Електронний ресурс] — Режим доступу: https://www.statista.com/topics/2430/smart-homes/

[2] WSU CASAS Datasets [Електронний ресурс] — Режим доступу: http://casas.wsu.edu/datasets/

активність. Однією з вимог до розроблюваної програмної системи є можливість функціонування у реальному часі. Для цього потік сенсорів має бути сегментований. Можно виділити такі види сегментації: за активностями, за часом і за сенсорами[3]. Види сегментації представлені на рисунку 2.

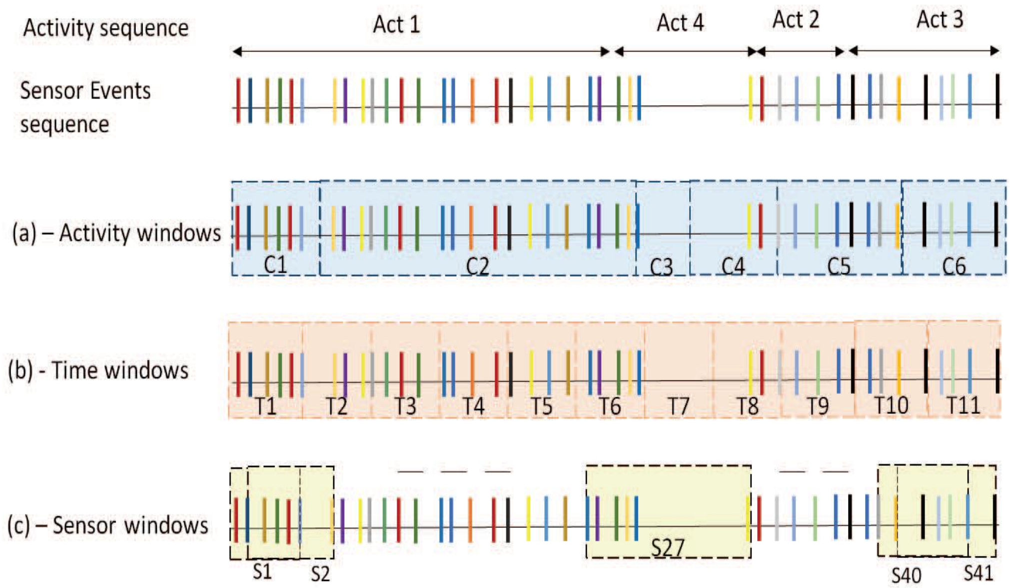


Рисунок 2. Види сегментації сенсорів

При сегментації за активностями вхідні сенсори спочатку діляться на вікна різної довжини і передбачається що кожне вікно містить певну активність. Проблемами такого підходу є те, що кожне вікно може містити декілька активностей і при функціонуванні у реальному часі такий спосіб має очікувати певну кількість активацій сенсорів, для того, щоб здійснити класифікацію.

При сегментації за часом потік сенсорів ділиться на вікна однакової часової довжини. Проблемою такого підходу є те, що в залежності від вибору довжини вікна, кожне вікно може як не мати в собі достатньої інформації для класифікації активності, так і мати в собі інформацію що відповідає декільком активностям.

При сегментації за сенсорами, потік сенсорів ділиться на вікна однакової кількості активацій сенсорів. В даному випадку передбачається класифікація саме останнього сенсору в вікні, минулі сенсори в даному випадку визначають контекст цього вікна. Даний метод теж має певні недоліки: вікно може містити в собі активації сенсорів між якими великі проміжки часу. Крім цього при наявності в будинку декількох мешканців, кожне вікно може містити інформацію про декілька паралельних активностей. Цей метод був вибраний по причині того, що для здійснення класифікації непотрібно очікувати певну кількість сенсорів, тобто класифікація здійснюється в момент надходження кожної активації сенсору.

[3] Krishnan N. Activity recognition on streaming sensor data / N. Krishnan, D. Cook // Pervasive and Mobile Computing / N. Krishnan, D. Cook., 2014. – С. 138–155.

Надалі для можливості тренування моделі кожний сегмент має бути трансформований в вектор ознак. Були вибрані наступні ознаки: час початку і кінця вікна, тривалість вікна, частота активації кожного з сенсорів і унікальний ідентифікатор останнього сенсору. Як було зазначено вище кожне вікно може містити певну кількість активацій, які відносяться до паралельної активності. Для того щоб зменшити вплив не релевантних сенсорів навчається матриця взаємної інформації, розмірність якої N × N де N є кількістю сенсорів. Матриця визначається наступним чином:

$$MI(i,j) = \frac{1}{|Q|}\sum_{k=1}^{|Q|} \partial(i,j) \qquad (1)$$

Де $\partial(i,j)$ має значення 1 в випадку якщо і сенсори і та j присутні в рамках однієї активності та 0 якщо ні. Q дорівнює кількості сегментованих активностей. Надалі інформація у матриці використовується для зважування частот у векторі ознак.

В рамках роботи було побудовано декілька різних моделей, серед яких метод опорних векторів, байесовський класифікатор і інші. Найкращу точність показав випадковий ліс. Випадковий ліс є ансамблевим класифікатором який складається з певної кількості дерев рішень, кожне з яких тренується на підвибірці з даних. Слід зазначити що дані які поступають на вхід до кожного з дерев можуть повторюватися. Самі дерева будуються за алгоритмом CART[4]. Перевагами цього методу є можливість роботи з великою кількістю ознак при чому ці ознаки можуть буть різних типів, можливість паралельного навчання, що в свою чергу зменшує час необхідний на тренування. Крім того необхідно зазначити що дана модель може гарно справлятися з неточними даними. Це є важливою характеристикою, так як існуючі набори даних для даної задачі, як правило, містять в собі неточності. Після навчання, класифікація здійснюється шляхом голосування кожного дерева і результативний клас є тим класом, за який проголосувало більшість дерев рішень. Кількість дерев рішень вибирається таким, яке надасть найкращий результат на даних валідації. При побудові моделі було вибрано 20 вирішуючих дерев. Модель була тренована на перших шести тяжнях набору даних і валідована на двух останніх. Отримана точність класифікації сягає 81%.

Розроблена програмна система надає можливість щодо візуалізації передбачених активностей у вигляді тривалостей активностей на часовому інтервалі. Приклад такої візуалізації зображений на рисунку 3.

---

[4] Belgiu M. Random forest in remote sensing: A review of applications and future directions / M. Belgiu, L. Draˇgut // ISPRS Journal of Photogrammetry and Remote Sensing / M. Belgiu, L. Draˇgut., 2016. – С. 24–31.

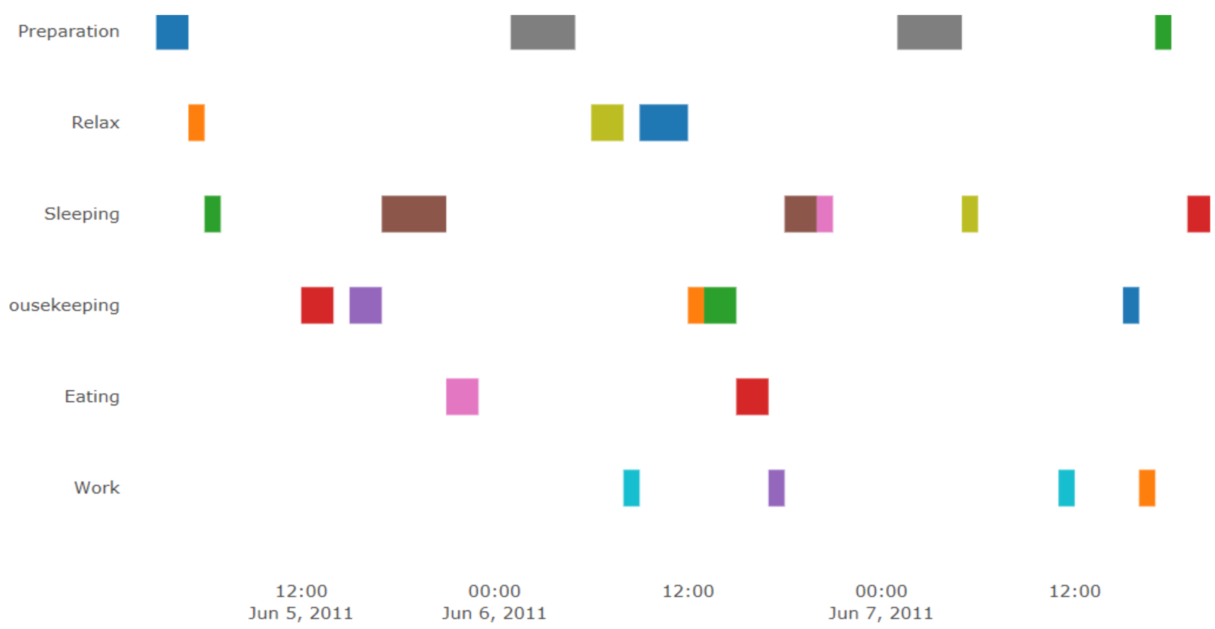


Рисунок 3. Візуалізація передбачених активностей

Висновки. В рамках роботи була проаналізована проблема аналізу активності мешканців розумного будинку. Було запропоновано формалізувати цю проблему як проблему класифікації. Були проаналізовані підходи щодо сегментації сенсорів та описаний обраний алгоритм машинного навчання.

Для вирішення поставленої проблеми запропонована розроблена програмна система, яка надає можливість здійснення класифікації у реальному часі та відображення результатів на часовому інтервалі.

ДОДАТОК В

Акт впровадження

Інтелектуальний аналіз даних в умовах розумного будинку

УКР.НТУУ”КПІ”_ТЕФ_АПЕПС_ ТР32265_18М

Аркушів 2

2018