

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

Кафедра інженерії програмного забезпечення

Пояснювальна записка до дипломного проєкту

магістра

(освітній ступінь)

на тему «Аналіз ефективності використання методів штучного інтелекту для
забезпечення безпеки руху водіїв автотранспорту»

XAI.603.6-95ПЗ1.121.168918.200

Виконав: студент 6 курсу групи № 6-95пз1
Спеціальність 121 – Інженерія програмного
забезпечення

(код та найменування)

Освітня програма Хмарні обчислення та
Інтернет речей

(найменування)

Якименко О.В.

(прізвище й ініціали студента)

Керівник Шостак І.В.

(прізвище та ініціали)

Рецензент Ляшенко О.С.

(прізвище та ініціали)

Харків – 2020

Міністерство світи і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

(повне найменування)

Кафедра інженерії програмного забезпечення

(повне найменування)

Рівень вищої освіти другий (магістерський)

Спеціальність 121 – інженерія програмного забезпечення

(код та найменування)

Освітня програма хмарні обчислення та Інтернет речей

(найменування)

ЗАТВЕРДЖУЮ

Завідувач кафедри

І. Б. Туркін

(підпис)

(ініціали та прізвище)

“ ”

2020 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТУ

Якименко Олексію Володимировичу

(прізвище, ім'я, по батькові)

1. Тема дипломного проекту Аналіз ефективності використання методів штучного інтелекту для забезпечення безпеки руху водіїв автотранспорту

керівник дипломного проекту Шостак Ігор Володимирович, д.т.н., професор

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “ ” 2020 року №

2. Термін подання студентом роботи

3. Вихідні дані до роботи: методи штучного інтелекту

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

дослідити причини виникнення ДТП та показників, котрі можуть бути використані для аналізу; дослідити методи штучного інтелекту для аналізу фізіологічних даних отриманих від водія з метою визначення показників самопочуття; розробити метод для аналізу показників водіїв та генерації порад з водіння щодо поліпшення безпеки руху автотранспорту; розробити прототип мобільного додатку для аналізу показників самопочуття водіїв та генерації порад з водіння

5. Перелік графічного матеріалу

РПЗ – стор. 78, рисунків – 11 шт., таблиць – 2 шт., презентація – 15 слайдів

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Шостак І.В., проф. каф. 603		
2	Шостак І.В., проф. каф. 603		
3	Шостак І.В., проф. каф. 603		

8. Нормоконтроль _____ В.А. Постернакова « ____ » _____ 2020 р.
(підпис) (ініціали та прізвище)

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту	Строк виконання етапів проекту	Примітка
1	Отримання і затвердження теми диплому	03.09.2019	
2	Аналіз предметної області	04.09.2019	
3	Постановка задачі	20.11.2019	
4	Проведення теоретичних досліджень	22.11.2019	
5	Розробка прототипу ПЗ	02.09.2020	
6	Підготовка пояснювальної записки	22.10.2020	
7	Оформлення пояснювальної записки до дипломного проекту	10.11.2020	
8	Передзахист дипломного проекту	27.11.2020	
9	Захист дипломного проекту	24.12.2020	

Студент

_____ (підпис)

Якименко О.В.

(прізвище та ініціали)

Керівник роботи

_____ (підпис)

Шостак І.В.

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломного проєкту містить 78 стор., 11 рис., 1 додаток, 23 джерела.

Об'єкт дослідження - процес підтримки прийняття рішень по забезпеченню безпеки руху автотранспорту.

Предмет дослідження - моделі та методи штучного інтелекту в задачах забезпечення безпеки руху автотранспорту.

Мета дослідження полягає в підвищенні ефективності підтримки прийняття рішень щодо забезпечення безпеки руху автотранспорту шляхом розробки мобільного додатку для аналізу показників самопочуття водіїв та генерації порад з водіння.

Наукова новизна. Удосконалено метод підтримки прийняття рішень щодо забезпечення безпеки руху автотранспорту, який на відміну від існуючих використовує фреймову модель для логіки аналізу даних системи, що дає змогу виконувати функції аналізу даних про стан водія з невеликими затратами ресурсів пристрою.

Практична значимість отриманих результатів. Розроблений прототип мобільного додатку контролює стан водія та процес руху транспортного засобу. Подібний додаток повинен поліпшити безпеку водія та процес пересування транспортних засобів на вулицях міст.

ШТУЧНИЙ ІНТЕЛЕКТ, МЕТОДИ ІНЖЕНЕРІЇ ЗНАНЬ, БЕЗПЕКА ВОДІЯ ПІД ЧАС РУХУ, ПРИЧИНИ ДОРОЖНЬО-ТРАНСПОРТНИХ ПРИГОД, МОБІЛЬНИЙ ДОДАТОК

ABSTRACT

Explanatory note to the master's thesis 78 pp., 11 fig., 1 app., 23 sources.

The object of study - the process of supporting decision-making to ensure traffic safety.

The subject of research - models and methods of artificial intelligence in the problems of traffic safety.

The aim of the study is to increase the effectiveness of decision support for traffic safety by developing a mobile application for analyzing the well-being of drivers and generating driving advice.

Scientific novelty. The method of decision support for traffic safety has been improved, which, unlike the existing ones, uses a frame model for the logic of data analysis of the system, which allows to perform the functions of analysis of data on the state of the driver with low device resources.

The practical significance of the results obtained. The developed prototype of the mobile application monitors the condition of the driver and the process of vehicle movement. Such an application should improve driver safety and the movement of vehicles on city streets.

ARTIFICIAL INTELLIGENCE, KNOWLEDGE ENGINEERING METHODS, DRIVER SAFETY DURING DRIVING, CAUSES OF TRAFFIC ACCIDENTS, MOBILE APP

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	5
ВСТУП.....	5
1 АНАЛІЗ ПРОБЛЕМНОЇ ГАЛУЗІ.....	7
1.1 Актуальність роботи	7
1.2 Огляд причин виникнення ДТП	8
1.3 Огляд засобів для поліпшення безпеки руху	11
1.3.1 Система Drive Safe & Save	12
1.3.2 Система Safe drive	14
1.3.3 Компанія Safe Drive Systems	16
1.4 Огляд факторів впливаючих на безпеку руху	18
1.5 Огляд методів обробки даних	19
1.5.1 Огляд методів машинного навчання	20
1.5.2 Огляд методів інженерії знань	26
1.6 Постановка задачі.....	31
2 МЕТОД ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ЩОДО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ РУХУ АВТОТРАНСПОРТУ	33
2.1 Модель проведення дослідів.....	33
2.2 Критерії вибору методу	40
2.3 Проведення порівнянь альтернатив	43
2.4 Моделювання системи на основі правила та знань.....	55
3 РОЗРОБКА ПРОТОТИПУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	58
3.1 Загальні відомості про систему	58

3.3 Інтерфейс та функціонал	60
ВИСНОВКИ.....	78
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	79
ДОДАТОК А.....	82

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

ДТП – дорожньо-транспортне порушення.

ПЗ – програмне забезпечення.

ПП – програмний продукт.

ВСТУП

В Україні за весь 2017 рік трапилось 158776 дорожньо-транспортних пригод. Усього постраждало 26782 людини, а загинуло 3410 людини. Найбільша кількість дорожньо-транспортних пригод у Київському регіоні [1].

Відповідно до даних всесвітньої організації охорони здоров'я, щороку в усьому світу внаслідок дорожньо-транспортних пригод на дорогах гине приблизно 1.35 мільйонів людей. Від двадцяти до п'ятдесяти мільйонів людей отримують не смертельні травми та багато з них лишаються інвалідами, внаслідок отриманих поранень. Найрозповсюдженішими причинами дорожньо-транспортних пригод є перевищення швидкісного обмеження, керування транспортного засобу (ТЗ) під впливом наркотичних речовин, погані метео умови, а також керування ТЗ з поганим самопочуттям [2].

Наприклад, 23 липня 2018 року пасажирський автобус, котрий прямував з Білорусії до України виїхав з траси та потрапив у кювет, саме через занадту втомленість водія. Водій автобусу просто заснув за кермом [3]. На щастя в цей раз ніхто не постраждав та не помер, але цю ситуацію можливо було б попередити, якби водій був оснащений системою, котра могла б виявити через мірну втомленість та виконати необхідні заходи, щоб завадити аварії.

На жаль, неможливо попередити абсолютно всі дорожньо-транспортні пригоди та врятувати життя і здоров'я усіх людей, але завдяки сучасним цифровим технологіям цілком можливо попередити більшість з них.

Сучасні технології дозволяють збирати точні дані про навколишнє середовище та біометричні дані про стан людини, а стрімкий розвиток інформаційних технологій та засобів штучного інтелекту забезпечують потужний апарат для обробки та аналізу отриманої інформації, в котрій можна визначити небезпечні ситуації. Це дозволяє побудувати програмну систему, що

оброблюватиме й аналізуватиме отримані дані про стан водія та допоможе поліпшити безпеку керування ТЗ.

Об'єкт дослідження - процес підтримки прийняття рішень по забезпеченню безпеки руху автотранспорту.

Предмет дослідження - моделі та методи штучного інтелекту в задачах забезпечення безпеки руху автотранспорту.

Мета дослідження полягає в підвищенні ефективності підтримки прийняття рішень щодо забезпечення безпеки руху автотранспорту шляхом розробки мобільного додатку для аналізу показників самопочуття водіїв та генерації порад з водіння.

Наукова новизна. Удосконалено метод підтримки прийняття рішень щодо забезпечення безпеки руху автотранспорту, який на відміну від існуючих використовує фреймову модель для логіки аналізу даних системи, що дає змогу виконувати функції аналізу даних про стан водія з невеликими затратами ресурсів пристрою.

Практична значимість отриманих результатів. Розроблений прототип мобільного додатку контролює стан водія та процес руху транспортного засобу. Подібний додаток повинен поліпшити безпеку водія та процес пересування транспортних засобів на вулицях міст.

1 АНАЛІЗ ПРОБЛЕМНОЇ ГАЛУЗИ

1.1 Актуальність роботи

У всьому світі транспорт являється однією з найбільших базових галузей економічної, виробничої і соціальної інфраструктури. Автомобіль дозволив людям мати майже необмежену мобільність, а також можливість вільно пересуватися. Особистий автомобіль скорочує витрачений на дорогу час, дозволяє бути незалежним у виборі напрямку руху і часу, не доведеться підлаштовуватися під розклад і маршрут руху автобуса, що робить поїздки більш комфортним.

Особистий автомобіль скорочує витрачений на дорогу час, дозволяє бути незалежним у виборі напрямку руху і часу, не доведеться підлаштовуватися під розклад і маршрут руху автобуса, робить поїздки більш комфортним.

Перший у світі автомобіль був винайдений на початку 18 століття. Протягом двохстах років автотранспорт поширився навколо світу та сьогодні неможливо уявити будь-яке сучасне міст у якому не було хоча б одного транспортного засобу. Перше зареєстроване ДТП сталося у США в 1896 році. Зі збільшенням популяції автомобілей у світі, значно збільшилась кількість аварій та жертв внаслідок ДТП.

Щорічно в світі від дорожньо-транспортних пригод гине близько 1.0 млн. чоловік. За даними Комісії Європейського співтовариства, внаслідок отриманих травм в ДТП гине близько 45 тис. європейців та 1.6 млн. отримують поранення, приблизно один з кожних 80 осіб у віці до 40 років. На кожного загиблого в ДТП припадає приблизно 25-35 поранених, багатьом з яких потрібна госпіталізація [4].

Проаналізувавши статистичні данні про кількість ДТП в Україні можна побачити, що протягом 2018 року в Україні сталося 162 тисячі дорожньо-транспортні пригоди. Для порівняння, за 2017 рік цей показник був 174 тисячі ДТП. Таким чином, в 2018 році на 7% автомобільних аварій сталося менше. Така тенденція однозначно є позитивною, проте ця проблематика все ще являється дуже

гострою. Адже загалом на дорогах України, за 2018 рік, загинуло 3432 людини. Ця цифра є великою, адже кожного дня на дорогах України помирає приблизно 9 людей [5].

За рівнем автомобілізації країн світу, Україна посідає тільки 81 місце. Найбільш автомобілізованими країнами є Сан-Маріно, Монако та США. Рівень автомобілізації показує кількість індивідуальних автомобілів в країні, що припадають на 1000 осіб та свідчить про рівень розповсюдженості автотранспорту у країні або у світі, а також про рівень заможності населення. Наприклад, в Україні на 1000 громадян тільки 173 людини мають власне авто. У Монако 899 осіб та в США 797. Це означає, що в Україні, з 1000 осіб, 173 користуються автомобілем та мають ризик стати учасником ДТП, а в більш розвинутих країнах, наприклад США, ця цифра більше 790 осіб [6]. Такі значні цифри свідчать про те, що автотранспорт став невід'ємною частиною побуту сучасних людей.

На основі розглянутої вище інформації, можна стверджувати, що дорожньо-транспортні пригоди є розповсюдженою проблемою, багато людей в усьому світі користуються автомобілями тому мають ризик стати учасником або навіть жертвою ДТП. Таким чином поліпшення безпеки дорожнього руху є сучасною та актуальною проблемою. Вирішення або навіть поліпшення цієї проблеми допоможе врятувати безліч людей, а завдяки сучасним інформаційним технологіям таке завдання цілком можливо вирішити.

1.2 Огляд причин виникнення ДТП

Керування автомобілем, особливо в умовах завантажених вулиць мегаполіса, вимагає від водія граничної уваги та концентрації. Дуже часто цих якостей багатьом не вистачає для забезпечення безпечного водіння. Найбільш поширені причини ДТП:

- порушення правил дорожнього руху водієм;
- помилка водія;
- погані погодні умови;
- погане почуття водія;
- поганий стан дорожнього покриття.

На жаль, дуже рідко розглядаються першопричини дорожніх пригод. Тим часом, однією з головних причин серйозних аварій часто стає погане самопочуття водія. Проблеми зі здоров'ям можуть призводити до зниження уваги і до повної втрати керованості.

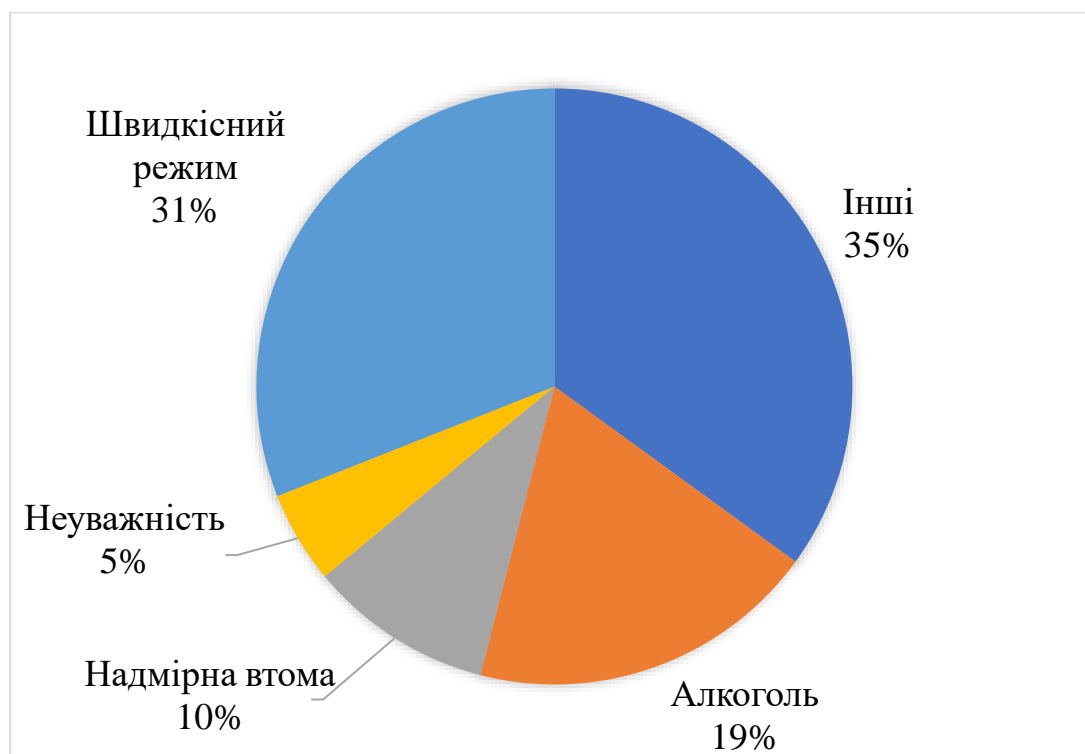
Будь-яке захворювання водія може істотно підвищувати ризик ДТП. Найбільш рідкісними, але і найбільш небезпечними, є випадки різкого загострення або нападів, які відбуваються з людьми, страждаючими хронічними захворюваннями. У подібних ситуаціях людина сідає за кермо в нормальному, або відносно нормальному стані, а загострення може статися безпосередньо на дорозі. У таких випадках водій може повністю втратити керування, що призводить до ДТП з тяжкими наслідками.

Найчастіше водій сідає за кермо з легким нездужанням. Навіть звичайна застуда може призводити до істотного зниження уваги і сповільненій реакції. Таким же чином можуть позначатися на стані водія і побічні ефекти від прийому деяких медикаментів. Результатом часто стає ДТП більшою чи меншою мірою тяжкості. Якщо розглядати види захворювань і поганого стану здоров'я водіїв, то найбільш поширеними причинами аварій стають: серцево-судинні захворювання, невралгічні захворювання, захворювання зору, епілепсія, цукровий діабет, недосипання.

Австралійська організація "Roadside assist" визначила [7], що основні причини нещасних випадків (див. діаграму 1.1) на дорогах Австралії в 2016 році були: прискорення, керуванням авто у нетверезому стані, втома водія, неуважність під час керування авто, а також набір інші причини, що стали причинами дорожньо-

транспортних пригод, до котрих також відноситься керування авто у хворобливому стані.

Всі основні п'ять причин можливо запобігти та проконтролювати людиною. Оскільки автомобілі стають більш автоматизованими є надія на те, що буде спостерігатися значне скорочення кількості дорожньо-транспортних пригод.



Діаграма 1.1 – відсоткове співвідношення між причинами ДТП за даними організації "Roadside assist"

Перевищення швидкості транспортних засобів робить складнішим маневрування, порівнюючи з повільнішими транспортними засобами, і чим швидше рухається авто, тим легше зіткнутися з іншими об'єктами. Ведення на більш високих швидкостях збільшує ризик аварій через занадто короткий час відгуку водія на будь-які несподівані дорожні умови.

Споживання алкоголю значно погіршує здатність організму людини обробляти та реагувати на несподівані ситуації. Вживання алкоголю впливає на

концентрацію, координацію, рефлексі та здатність водія приймати правильні судження та рішення, які необхідні для безпечного керування будь-яким транспортним засобом. Керування автомобілем у нетверезому стані є розповсюдженою причиною дорожньо-транспортних пригод на дорогах.

Аварії, які виникають, коли водій втомлюється зазвичай трапляються через те, що людина була позбавлена сну протягом доби або якщо їде пізно вночі. Водії потрапляють в смертельну аварію в чотири рази частіше внаслідок втоми між годинами з 10 вечора до світанку. До особливої групи ризику відносять водії дальніх рейсів, наприклад водії вантажних фур та пасажирських автобусів, що перевозять людей. Надмірна втома значно погіршує людську увагу та рефлексі, а також знижує можливість розсуджувати та працювати, тому необхідно пильно слідкувати за рівнем втоми водія.

Аварії внаслідок відволікання водія, стали більш поширеними з моменту створення смартфонів, які відволікають увагу людини, що керує ТЗ. Дослідження показали, що використання мобільного телефону під час керування ТЗ збільшує ризик ДТП в 4 рази [8]. Існують також інші фактори, котрі відволікають водія під час керування ТЗ: відстеження та коригування параметрів автомобіля (наприклад швидкість), пасажери та інші зовнішні чинники.

1.3 Огляд засобів для поліпшення безпеки руху

Найбільш слухними пристроями, котрі можуть бути використаними водієм для допомоги керування транспортним засобом є вбудований в автомобіль бортовий комп'ютер, смартфон або розумні часи. Так як майже всі люди є власниками смартфонів сьогодні, та не всі автомобілі мають бортовий комп'ютер, мобільна платформа є цільовою платформою для використання застосування, котре забезпечує безпеку руху.

Важливою вимогою до такого застосування – є автономність, тобто система повинна виконувати свої функції в умовах відсутності з'єднання з інтернетом, а також в обмежених умовах продуктивності. Тобто програмне забезпечення та алгоритм повинен правильно функціонувати на тому апаратному забезпеченні, що має звичайний мобільний телефон. Це означає, що важливою вимогою до алгоритму є низьке споживання ресурсів.

Сучасна система, котра забезпечує безпеку руху, повинна визначати проблемні ситуації та допомагати водієві уникати їх, а також попереджати надзвичайні служби про виникненні аварійні ситуації. Під час аналізу існуючих систем, котрі займаються поліпшенням безпеки руху, була лише одна система, котра брала до уваги стан водія. Таким чином можна зробити висновок, що система для поліпшення безпеки руху, котра має широкі функціональні можливості ще не створена у повному об'ємі, тому дослідження та реалізація такої системи є актуальною задачею.

1.3.1 Система Drive Safe & Save

Головна мета продукту Drive Safe & Save записувати звички керування транспортним засобом водія, аналізувати їх, виділяти ризиковані звички, наприклад занадто різке гальмування та генерувати порад щодо покращення манери керування транспортним засобом.

Система Drive Safe & Save складається з:

- кросплатформенного програмного додатку для мобільних платформи Android та iOS;
- Bluetooth-маяка, котрий встановлюється в середині автомобіля та зчитує дані протягом руху автомобіля.

Для того щоб отримати Bluetooth-маяк клієнтові необхідно виконати єдино разовий платіж, після котрого компанія Drive Safe & Save оформитиме посилку та за допомогою поштових служб відправить придбаний товар до кінцевого користувача.

Мобільний додаток (див. рис. 1.1) переглядає поїздки на карті, інформує про надмірне прискорення, різкі повороти та жорстке гальмування. Додаток також генерує поради щодо покращення показників водіння на основі ключових характеристик керування [9].

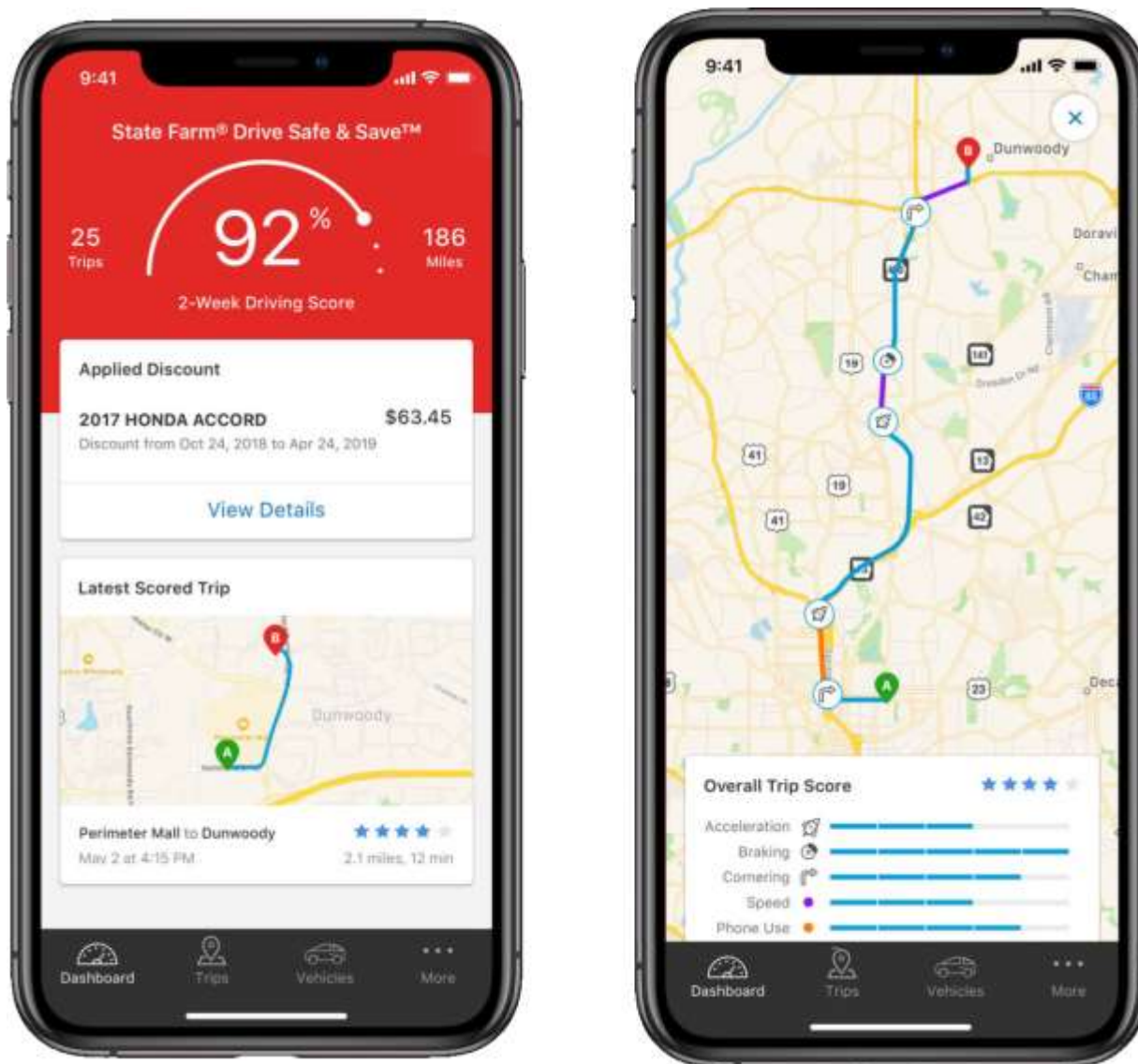


Рисунок 1.1 – Знімки з екрану додатку Drive Safe & Save

Раніше було сказано, що дана система забезпечує свої функціональні можливості завдяки Bluetooth маяку (див. рис. 1.2).

Bluetooth-маяк представляє собою невелику пластикову коробку оснащену різноманітними сенсорами. Такий маяк містить в собі сенсори прискорення та гальмування, сенсор освітлення та гіроскоп. Ці сенсори зчитують інформацію з навколишнього середовища та передають дані до мобільного додатку, що дозволяє забезпечити більш коректну роботу мобільного застосування.



Рисунок 1.2 – Bluetooth маяк системи Drive Safe & Save

Дана система достатньо якісна та підтримує обидві мобільні платформи, але, нажаль, інтегрується лише з автомобілем та не бере в увагу самопочуття водія. Саме через це вона не може гарантувати повноцінну безпеку руху.

1.3.2 Система Safe drive

У 2017 році компанія Mercedes Benz вирішила провести турнір, запропонувавши будь-яким програмістам з усього світу розробити додаток, котрий інтегрується з новими автомобілями Mercedes Benz. За вимогами конкурсу додаток повинен реалізовувати цікаву ідею та використовувати технічні можливості автомобіля. Переможець отримав 15 тисяч євро. За рішенням комісії Mercedes Benz Challenge перемогла команда, котра реалізувала мобільний додаток Safe Drive (див рис. 1.3).

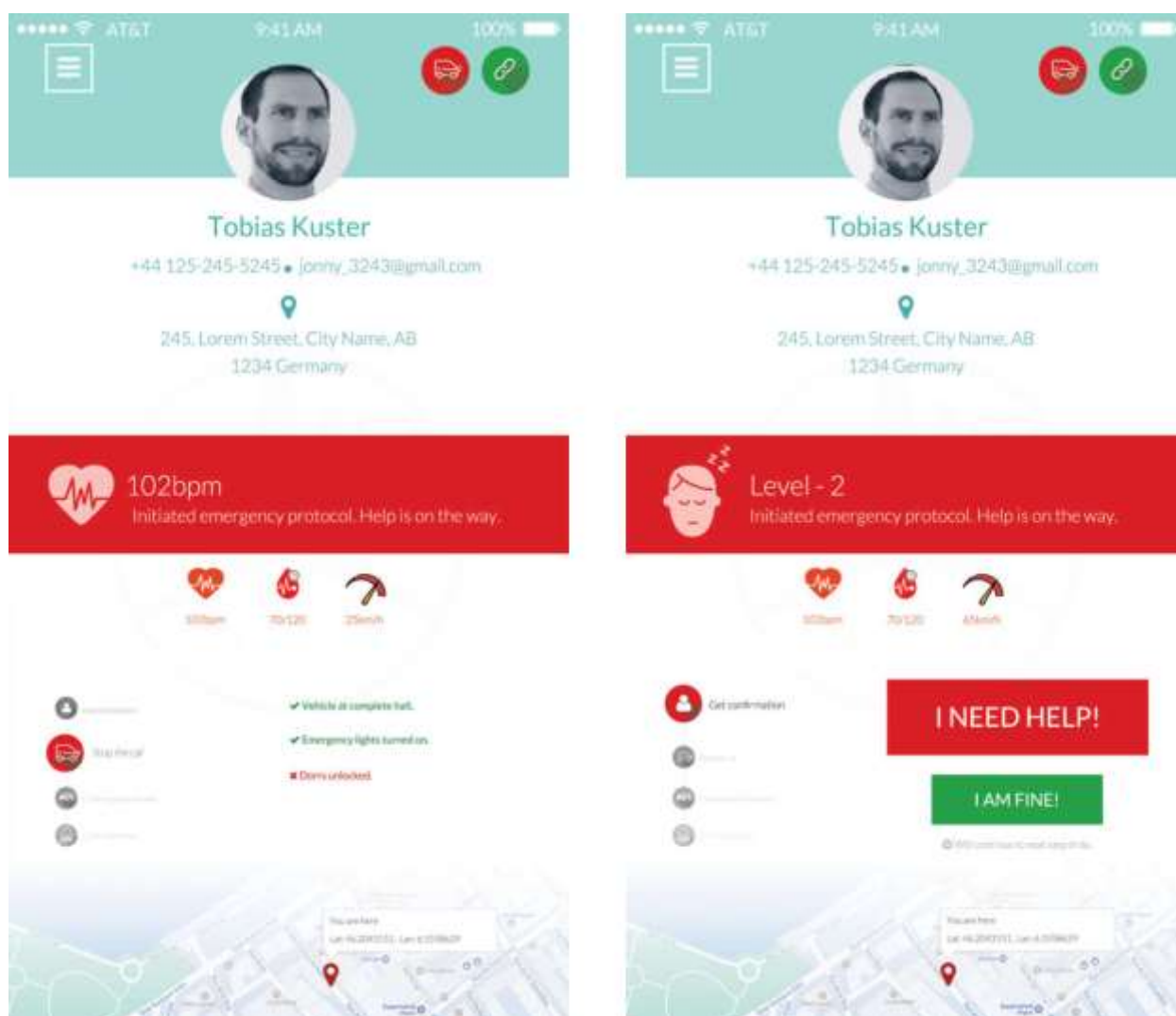


Рисунок 1.3 – Знімок з екрану головної сторінки додатку SafeDrive

Цей мобільний додаток пильно слідкує за станом водія, чи може він керувати автомобілем, чи відчуває він себе добре, чи втомився він. Якщо додаток зауважує, що життя водія під загрозою, система виконує дії для підвищення безпеки водія. За допомогою голосових, світлових та текстових сповіщень додаток намагається відновити увагу водія та перевіряє його здатність висловлюватися та керувати автомобілем. Якщо водієві стало погано система виконує заходи для забезпечення безпеки, наприклад: зупинка автомобіля, відкриття дверей та включення аварійних сигналів. У той же час, при необхідності, застосування попереджає аварійну службу, посилаючи останні дані про автомобіль та водія: інформація про місцезнаходження та стан здоров'я. Також програма надає інформацію про автомобіль та про здоров'я водія, котра буде відображатися на дисплеях приладової панелі [10].

Даний додаток реалізує багатий функціонал, що забезпечує безпеку керування транспортним засобом, але на жаль дана система є тісно пов'язаною з автомобілями компанії Mercedes Benz та будь-яких інших можливостей для інтегрування.

1.3.3 Компанія Safe Drive Systems

Safe Drive Systems є єдиною компанією в США, яка пропонує радіолокаційні та камерні систем запобігання зіткненням. Її основною продукцією є радари RD-140 та RD-140-RDR (див рис 1.4). Обидва продукти RD-140 та RD-140-RDR збудовані на основі радіолокаційних систем, але також для продукту RD-140 є модифікація розроблена на основі камери.

Системи безпечного приводу компанії Safe Drive Systems, розроблені системи на основі систем FMS, що інтегруються з системами захисту від зіткнень та можуть передбачати поведінку водія.

Компанія Safe Drive Systems була заснована у 2008 році з амбіційною місією надання післяопераційних систем запобігання зіткненням для водіїв дорогах країни, які шукають передові технології для попередження дорожньо-транспортних пригод та поліпшення безпеки руху. З 2008 року компанія продала понад 200 000 одиниць по всьому світу. Увійшовши на ринок США у 2013 році, їх безпрецедентний ріст продажів допоміг компанії розширитися до більш ніж 15 країн по всій Америці в 2015 році, захищаючи сім'ї від Нью-Йорка до Каліфорнії [11].



Рисунок 1.4 – Зовнішній вигляд радару RD-140

Ця система є досить перспективною, так як використовує сучасні радари для отримання даних про навколишнє середовище, але на жаль система орієнтована тільки на взаємодію між автомобілями, та не бере будь-які інші зовнішні показники або показники самопочуття водія.

1.4 Огляд факторів впливаючих на безпеку руху

Існують різноманітні фактори, котрі впливають на керування транспортним засобом, починаючи від самопочуття водія та закінчуючи часом доби. Однією з найрозповсюдженіших першопричин ДТП є перевищення швидкості, тому данні про швидкість ТЗ є дуже важливими.

Час доби дуже важливий показник. Від часу доби залежить освітленість дороги. Тому під час керування авто вночі, напруга, котру відчуває водій набагато більша, бо від поганої освітленості швидше втомлюються очі, а також нервова система. Також вночі набагато менша відстань до об'єктів, котрі бачить водій.

Неможливо абстрагуватися від метеорологічних умов під час керування автомобіля. Наприклад, швидкість 60 кілометрів на годину під час снігопаду взимку та під час ясного сухого дню влітку несе різні ризики.

Керування авто у нездоровому стані достатньо небезпечно. Наприклад, підвищена температура водія може свідчити про погане самопочуття та необхідність відпочинку.

Нові автомобілі Mercedes Benz вимірюють рівень втоми водія на основі кількості часу проведеного за кермом. Оскільки за даними організації "Roadside assist" 10% відсотків дорожньо-транспортних пригод трапляються через надмірну втому водія, контролювання рівня втомленості людини може позитивно вплинути на безпеку руху. Вимірювання тиску крові водія може більш точно вказати на його рівень втомленості, бо гіпертонія є одним з показником перевтомлення здорової людини.

Вимірювання показника пульсу є дуже важливою метрикою. Високе значення пульсу в медицині має назву тахікардія. Кажуть про виникнення даного симптому, при збільшенні серцевих ударів до значення - ста ударів у хвилину.

Людина може не відчувати жодного дискомфорту при умашеному пульсу або ж навпаки відчувати занепокоєння, запаморочення, головний біль. При таких симптомах керування автомобіля становиться занадто небезпечним. Також вимірювання пульсу водія може ідентифікувати симптоми інфаркту, що допоможе завчасно сповістити екстрені служби, котрі зможуть швидко прибути для надання допомоги.

1.5 Огляд методів обробки даних

Невід'ємною частиною системи для поліпшення безпеки є логіка для аналізу даних про стан водія отриманих з сенсорів, під час руху автомобіля, для подальшого прийняття рішень. Це означає, що система повинна розуміти данні та стан процесу руху в якому знаходиться водій для того, щоб чітко визначити можливі загрози для здоров'я та життя водія . Розв'язання проблем "розуміння" інформації комп'ютером, в комп'ютерних науках, займається розділ інформатики – штучний інтелект.

Штучний інтелект займається символічним моделювання розумових процесів, машинним навчанням, робототехнікою, машинною творчістю та іншими задачами. До штучного інтелекту відносять ряд алгоритмів та програмних систем, відмінною властивістю яких є можливість вирішувати різноманітні завдання так, як це робить людина, що розмірковує над їх вирішенням [12].

На прикладі цієї роботи, штучний інтелект даної системи повинен чітко визначати самопочуття водія та на основі поставленого "діагнозу" передбачати загрози для руху транспортного засобу.

Будь-які системи штучного інтелекту зберігають знання таким чином, щоб алгоритми могли обробляти їх та бути подібними людському інтелекту. Вирішення такого завдання є основною метою штучного інтелекту. Одне з найважливіших

напрямок дослідження в області штучного інтелекту – є дослідження моделей подання або представлення знань. Без знань штучний інтелект не може існувати, а саме моделі представлення задають структуру зберігання даних, які згодом будуть оброблятися для прийняття штучним інтелектом рішення.

У контексті цієї роботи нас цікавлять методи штучного інтелекту, що дозволяють аналізувати вхідні дані та на основі існуючих знань, визначити чи відносяться вони до аномальних даних, що представляють небезпечну ситуацію. Така задача відповідає завданню класифікації. Завдання класифікації – це завдання, що має безліч об'єктів (ситуацій), розділених деяким чином на класи та головною метою є визначення до якого класу відноситься той чи інший алгоритм.

Вирішенням подібних завдань займається один з напрямків штучного інтелекту – машинне навчання, концепція котрого полягає в набутті інтелектуальних знань системою протягом її роботи [13].

Альтернативою машинному навчанню є напрям інженерії знань, що має методи та алгоритми, котрі дозволяють розробити набір правил, що моделюють предметну область. Такі правила формують експертні системи та мають можливість на основі знань класифікувати вхідні дані та робити висновки. Таким чином правильно реалізована експертна система дозволить визначити відхилення від нормальних показників стану водія та запропонувати варіанти поліпшення ситуації.

1.5.1 Огляд методів машинного навчання

Машинне навчання – підгалузь штучного інтелекту, яка застосовує статистичні прийоми для надання комп'ютерам здатності поступово покращувати продуктивність у певній задачі, без явного перепрограмування [14].

Головна мета машинного навчання – створення алгоритмів, що дозволяють узагальнити досвід минулих спостережень для обробки нових випадків, що не зустрічалися раніше, результат яких невідомий.

Найважливішою вимогою для використання алгоритмів ML – наявність великої кількості навчальної інформації. Як правило в якості такої виступає набір ситуаційних прикладів з минулого з відомим результатом.

Алгоритми машинного навчання мають високі вимоги до вхідних даних: обсяг початкової вибірки повинен бути великим, а данні повинні бути максимально правильні. Навчальна вибірка є основним об'єктом роботи будь-якого методу ML. Її розмір суттєво впливає на процес машинного навчання.

Великий об'єм вибірки даних дозволяє отримати надійніші результати, використовувати складніші моделі алгоритмів, оцінювати точність навчання, але буде призводити до зростання часу навчання.

При використанні невеликих об'ємів даних існує можливість використовувати лише прості моделі алгоритмів. В такому випадку швидкість навчання буде максимальна, але якість набагато менша. Малі вибірки підвищують ймовірність перенавчання при помилці в виборі моделі [15].

Дуже популярним напрямом машинного навчання – є дослідження засобів реалізації медичних систем, що дозволяють ставити діагнози хворим. Визначення діагнозу через машинне навчання працює, коли вхідні умови можуть бути зведеними до завдання класифікації на основі фізіологічних даних.

У рамках цієї атестаційної роботи магістра в першу чергу інтерес привертає підходи рішення завдання класифікації за допомогою машинного навчання. Вирішення завдання класифікації дозволяє на основі набору даних та визначених результатів «навчити» цільову функцію визначити аномальні ситуації та генерувати чіткі кроки для вирішення виявлених загроз.

У машинному навчанні, класифікація – це підхід навчання алгоритму з учителем за допомогою якого комп'ютерна програма вивчає дані, що вводяться їм, а потім використовує навчання для класифікації нового спостереження. Цей набір

даних може бути двокласним (наприклад, визначення того, чи є людина чоловіком чи жінкою або що електронний лист є спамом або не спамом), або може бути багатокласним [16].

Основні алгоритми класифікації в машинному навчанні [17]:

1. логістична регресія;
2. метод Баєса;
3. метод k найближчих сусідів;
4. метод дерев рішень;
5. метод випадкового лісу;

Логістична регресія. Логістична регресія є алгоритмом машинного навчання для класифікації. У цьому алгоритмі ймовірності, що описують можливі результати одного випробування, моделюються з використанням логістичної функції, котра була створена під час вивчення змін у популяції населення.

Перевагами логістичної регресії є те, що вона від самого початку призначена для завдання класифікації та є найбільш корисною для розуміння впливу декількох незалежних змінних на одну змінну результату.

Недоліком логістичної регресії є те, що вона працює тільки тоді, коли передбачена змінна бінарна, всі предиктори не залежать один від одного, а також що набір даних повний.

Логістична регресія використовується для виявлення факторів ризику захворювання та планування профілактичних заходів, класифікації слів за частинами мови, прогнозування погоди. Хорошим прикладом логістичної регресії є приклад, коли компанії кредитних карт розробляють моделі, які вирішують, чи буде клієнт дефолтувати свої кредити чи ні.

Метод Баєсу. Цей алгоритм заснован на теоремі Баєсу з припущенням, що між кожною парою властивості(атрибуту) не існує зв'язку, тобто вони незалежні. Цей класифікатор добре вирішує багато реальних проблем, наприклад, фільтрація від спаму та класифікація документів.

Найбільшою перевагою цього методу – є те, що він не вимагає великої кількості вхідних даних для оцінки необхідних параметрів. Також цей метод набагато швидше, у порівнянні з іншими складними методами.

Недоліком цього алгоритму – є те, що він припускає, що властивості деякого класу незалежні. Через це знижується точність.

Метод Баєсу дозволяє автоматично класифікувати веб-сторінки, повідомлення на форумах, фрагменти блогу та твітів.

Метод k найближчих сусідів. Модель ледачого навчання, оскільки вона не намагається побудувати загальну внутрішню модель, а просто зберігає екземпляри навчальних даних. Класифікація обчислюється простою більшістю голосів до найближчих сусідів кожної точки.

Переваги. Алгоритм простий, а також надійність до шумних даних навчання, а також ефективність, коли дані навчання великі.

Недоліки. Необхідність визначення величини K . Витрати такого обчислення високі, оскільки вони потребує обчислення відстані кожного екземпляра до всіх навчальних вибірок.

Алгоритм k найближчих сусідів використовується в промислових додатках, коли користувач намагається знайти подібні елементи, ознаки порівняно з іншими. Також цей метод використовується в програмах для виявлення рукописного тексту та завдань розпізнавання зображень / відео.

Дерева рішень. Цей метод бере дані атрибутів з його класами, та буде деревоподібну послідовність правил, які можна використовувати для класифікації даних.

Переваги. Дерево прийняття рішень є простим для розуміння та візуалізації, потребує невеликої підготовки даних і може обробляти як числові, так і категоріальні дані.

Недоліки. Дерево прийняття рішень може створювати складні дерева, які не узагальнюються, і дерева рішень можуть бути нестабільними, оскільки невеликі зміни даних можуть призвести до створення абсолютно іншого дерева.

Застосування дерев рішень варіюється від дослідження даних, розпізнавання образів, ціноутворення опцій у фінансах і виявлення тенденцій хвороб і ризиків.

Метод випадкового лісу. Класифікатор випадкових лісів – це мета-оцінювач, який вписується в ряд дерев рішень на різних під-зразках наборів даних і використовує середнє значення для поліпшення прогнозної точності моделі та управління переналагодженням. Розмір зразка завжди є таким же, як і початковий розмір вибірки, але зразки відбираються з заміною.

Переваги. Класифікатор випадкових лісів більш точний, ніж дерева рішень у більшості випадків.

Недоліки. Повільне прогнозування в реальному часі, складне в реалізації та складний алгоритм.

Алгоритм випадкового лісу використовується в фінансових додатках, коли необхідно визначити, чи є заявник кредиту низько ризиковим або високо ризиковим. Також ця модель використовується для промисловості, прогножуючи відмову механічних запчастин.

Усі без винятку ML алгоритми мають проблему пов'язану перенавчання. При перенавчанні побудований алгоритм дуже добре працює на прикладах, що брали участь в навчанні, але досить погано працює на прикладах, які не брали участі в навчанні. Це пов'язано з тим, що при побудові такого алгоритму «в процесі навчання» в навчальній вибірці виявляються деякі випадкові закономірності, які відсутні в реальному домені.

Для уникнення перенавчання використовують додаткові методики машинного навчання:

– перехресну перевірку (при оцінці моделі наявні дані розбиваються на k частин. Потім на $k-1$ частинах даних проводиться навчання моделі, а частина даних, що залишилася використовується для тестування. Процедура повторюється k разів; в результаті кожна з k елементів даних використовується для тестування. В результаті виходить оцінка ефективності обраної моделі з найбільш рівномірним використанням наявних даних);

– регуляризацію (метод додавання деякої додаткової інформації до умови. Ця інформація часто має вигляд штрафу за складність моделі. Наприклад, це можуть бути обмеження гладкості результуючої функції або обмеження по нормі векторного простору);

– спрощення архітектури нейронної мережі, зменшення числа вхідних сигналів, зведення параметрів нейронної мережі до невеликої кількості виділених значень, зниження вимог до точності вхідних сигналів.

Слід зауважити, що на мобільних пристроях використовується кілька систем глибокого навчання: розпізнавання осіб, китайських ієрогліфів, кодових фраз "Hello, World" або "Hello, Siri". Варто враховувати, що всі ці програми нічому не навчаються у користувача. Найчастіше весь процес навчання відбувається на віддаленому сервері з великою кількістю GPU, розрахованих на велику кількість паралельного обчислення. Дані про запити користувачів надходять саме до серверу, які згодом обробляються системою.

Часом це займає дуже багато часу. Як приклад можна привести згортальну нейронну мережу, якій для навчання потрібно проаналізувати мільйони картинок. На потужному сервері дана операція займе декілька днів, на середньому комп'ютері – пару місяців і дуже великий час на мобільному телефоні.

Для додатків, оновлення яких відбувається нерегулярно, серверне навчання – це відмінна стратегія. На жаль, навчання серйозних систем на телефоні поки що неможливо, але, зауважуючи темпи прогресу нових технологій, це – лише питання часу.

Виходячи з вищеперерахованого можна прийти до висновку, що машинне навчання є дуже потужним підходом для реалізації інтелектуальних систем, але такий підхід є достатньо ресурсоємним для мобільних телефонів. Слід зазначити, що найбільш відповідними моделями для діагностики є логістична регресія та дерева рішень.

1.5.2 Огляд методів інженерії знань

Алгоритми прийняття рішень на основі методів інженерії знань та моделей знань, мають інший характер, на відміну від алгоритмів машинного навчання. Вони не потребують для роботи великих обчислювальних потужностей, проте для правильної роботи таких алгоритмів підбір правильних даних є критично важливим.

Інженерія знань являє собою сукупність моделей, методів і технічних прийомів, націлених на створення систем, які призначені для вирішення проблем з використанням знань.

Такі системи називаються експертними системами. Експертні системи – це система штучного інтелекту, що включає знання про певну слабо структурованої і важко формалізується вузькій предметній області і здатна пропонувати і пояснювати користувачеві розумні рішення. Експертна система складається з бази знань, механізму логічного висновку і підсистеми пояснень.

Фактично інженерія знань – це теорія та технології, які охоплюють методи видобутку, аналізу, представлення й обробки знань експертів.

Важливим питанням при створенні системи на основі моделей знань є вибір способу представлення знань. Мета подання знань - організація необхідної інформації в таку форму, щоб програма мала легкий доступ до неї для прийняття рішень, планування, впізнавання об'єктів і ситуацій, аналізу сцен, виведення висновків та інших когнітивних функцій.

На даний час розроблено різноманітні моделі представлення знань. Такі моделі можна умовно поділити на логічні та емпіричні.

В логічних моделях вся інформація представлена як набір тверджень і фактів, що їх зв'язують, та представлені у вигляді формул в деякій логіці. При цьому знання відображені набором подібних одне до одного тверджень, а побудова висновків і отримання нових знань зводиться до реалізації процедури логічного

висновку. Цей процес може бути строго формалізований, тому що в його основі лежить класичний апарат математичної логіки.

Інший підхід, що зветься емпіричним заснований на вивченні принципів організації людської пам'яті і моделюванні механізмів вирішення завдань людиною. На основі цього підходу в даний час розроблено велику кількість видів моделей, деякі з них ми розглянемо далі [18].

На основі емпіричного підходу на даний час розроблені і отримали найбільшу популярність наступні моделі: продукційна, мережева, фреймова та логічні модель.

Логічна модель – це така модель представлення знань, основною ідеєю якої є те що, вся інформація, необхідна для вирішення прикладних завдань, розглядається як сукупність фактів і тверджень, які представляються як формули в деякій логіці. Знання в такій моделі відображаються сукупністю формул логіки, а отримання нових знань зводиться до реалізації процедур логічного виводу.

Логічна модель використовує мову логіки предикатів що включає:

- предметні константи (імена індивідуумів: об'єктів, подій);
- предметні змінні. Їх значення – предметні константи. Кожній змінній відповідає певна область значень, котрі можуть призначенні певній змінними;
- предикатні константи;
- функціональні імена;
- логічні зв'язки, квантори;
- правила створення логічних формул.

В основі логічних моделей представлення знань лежить поняття формальної теорії, що задається кортежем:

$$S = \langle B, F, A, R \rangle \quad (1.1)$$

де B – множина базових символів (алфавіт), F – множина формул, A – множина аксіом, R – кінцева множина відносин між формулами (правила виведення).

Основою перевагою цієї моделі є використання класичного апарату математичної логіки, методи якої досить добре вивчені і формально обґрунтовані, тому представлення знань за допомогою цієї моделі є досить надійним. Також до переваг логічної моделі слід віднести, що вони дозволяють зберігати у базах знань безліч аксіом, а всі інші знання отримувати з них за правилами виведення, а також Дані, факти та інші відомості про людей, предмети, події та процеси [19].

Продукційна модель – це також модель представлення знань, ідеєю якої є представлення знань у вигляді продукцій – виразів вигляду: ЯКЩО <умова> ТО <дія> та ЯКЩО <умова> ТО <дія> ІНАКШЕ <дія>. Такий запис схожий на імплікацію (логічний вислів, стверджуючий, що, якщо ліва частина істина то істина і права частина), але продукція скоріше схожа на інструкцію аналогічну умовним операторам мов програмування. Тобто якщо ліва частина істина тоді необхідно виконати деяку дію.

У загальному випадку продукційну модель можна представити в наступному вигляді:

$$i = \langle S; L; A \rightarrow B; Q \rangle \quad (1.2)$$

де S – опис класу ситуацій, L – умова, при якому продукція активізується, $A \rightarrow B$ - ядро продукції, Q – післяумова продукційного правила.

Традиційно будь-яка продукційна модель включає в себе наступні компоненти:

1. Набір продукцій, що представляють базу знань продукційної системи.

2. Робочу пам'ять (база даних), в якій зберігаються вихідні знання, а також знання, сформовані з інших правил за допомогою механізму логічного висновку.

3. Механізм логічного виводу, що дозволяють з існуючих знань, згідно з правилами формування, формувати нові знання.

Кожне правило представляє частину бази знань продукційної системи та складається з умовної та заключної частини. В умовній частині правила знаходиться або одиночний факт, або кілька фактів. У заключній частині правила знаходяться процедури, які повинні бути виконані, якщо умовна частина правила є істинною [20].

Продукційна модель має багато переваг, наприклад: простота створення окремих правил, простота модифікації бази знань (набору продукцій), простота механізму логічного висновку та можливість паралельної і асинхронної обробки правил.

До недоліків продукційних моделей слід віднести наявність конфліктів між продукцій при великих об'ємах правил у база знань. Для вирішення конфліктних ситуацій існують набір підходів, що дозволяють прийти до правильного висновку.

Ще однією моделлю представлення знань є модель на основі фреймів. Фрейми – це структури даних, в яких представлені знання про властивості об'єкта. Нині концепція фреймів швидко розвивається і розширюється, завдяки розвитку методів об'єктно-орієнтованого програмування.

Інформація, що відноситься до фрейму, міститься в його слотах. Слоти – це деякі змінні фрейма, котрі, після заповнення даними, дозволять фрейму представляти ту чи іншу ситуацію, явище чи об'єкт предметної області.

Фрейми утворюють ієрархію, що породжує єдину багаторівневу структуру, описуючи або об'єкт, якщо слоти описують тільки властивості об'єкта, або процес, якщо окремі слоти є іменами процедур, приєднаних до фрейму.

Фреймова модель має значну перевагу так як дозволяє представляти складні об'єкти у вигляді єдиної сутності. Фреймову модель досить часто реалізують, використовуючи об'єктно-орієнтовані мови програмування, так як основний

концепт представлення знань у фреймових моделях схожий до особливостей представлення проблемних областей в об'єктно орієнтованому програмуванні.

Останньою моделлю представлення знань є модель на основі семантичної мережі. В основі семантичних моделей лежить орієнтований граф, утворений вершинами та ребрами. Вершини мережі представляють деякі поняття (абстрактні або конкретні об'єкти, процеси), а ребра – відносини між ними, які вони пов'язують (зв'язки типу «це», «має частиною», «належить»)

В залежності від типів відноси між вершинами та ребрами розрізняють такі види мереж:

- класифікаційні, в яких використовуються відносини (ціле, рід, вид, індивід), що описують структуру предметної області та дозволяє відобразити в базах знань різні ієрархічні відносини між інформаційними одиницями;
- функціональні (їх часто називають обчислювальними моделями), що дозволяють описувати процедури «обчислень» одних інформаційних одиниць через інші;
- змішані, що використовують різноманітні типи відносин.

Семантичні мережі можна класифікувати за:

- кількістю відношень (однорідні та неоднорідні);
- арністю (бінарні та n-арні);
- розміром (локальна, глобальна, домена).

Методи інженерії знань є фундаментом експертних систем, що використовуються при розробці систем заснованих на правилах для сучасних медичних застосувань.

Вибір моделі представлення знань залежить від типу проблем, що необхідно вирішити, а також від конкретних засобів, що будуть використані під час реалізації тієї чи іншої програми. Так, наприклад, для процедурної мови програмування С найбільш зручна буде продукційна модель, а для об'єктно орієнтованих мов програмування краща буде Java або С#. Всі 4 моделі мають свої переваги й

недоліки та являються потужними засобами створення програмних систем, котрі моделюють людський інтелект.

1.6 Постановка задачі

Під час огляду існуючих систем поліпшення безпеки руху було виявлено, що сьогодні не існує універсальної системи котра могла би працювати з різноманітними автомобілями та приймати до уваги як і біометричні дані водія так і дані про навколишнє середовище.

Програмне забезпечення такої системи повинно мати можливість чітко розуміти в якому стані знаходиться водій та генерувати поради для забезпечення безпеки руху, а у разі нещасного випадку допомогти врятувати життя водієві та пасажирам.

Невід'ємною частиною такої системи є частина програмного забезпечення, що займається аналізом та обробкою отриманих даних, виявляючи можливі ризики та проблеми. Тому головною метою даної роботи є дослідження методів аналізу даних та прийняття рішень на базі методів штучного інтелекту, що будуть використані у прототипі програмного додатку для забезпечення безпеки руху.

У попередньому опрацюванні літератури стало відомо, що алгоритми прийняття рішень базуються на методах штучного інтелекту. Найбільш придатними методами є методи машинного навчання та інженерії знань. У машинному навчанні було визначено 2 моделі, котрі вирішують суміжне завдання із завданням атестаційної роботи: ліс дерев та лінійна регресія. У інженерії знань найбільш придатними моделями виявились продукційні та фреймові.

Також під час опрацюванні існуючих додатків стало зрозуміло, що цільова платформа – мобільна. Тому під час дослідження методів слід звернути значну увагу до споживання ресурсів алгоритмів. Вони повинні споживати якомога менше

пам'яті та їх коректна робота повинна бути незалежною від наявності інтернет з'єднання.

Задачі атестаційної роботи:

– дослідити та визначити найбільш придатний метод штучного інтелекту для реалізації логіки аналізу даних про стан водія у мобільному додатку для поліпшення безпеки руху;

– реалізувати прототип мобільного додатку для поліпшення безпеки руху.

Для реалізації системи було обрано середовище розробки Microsoft Visual Studio Code. Для реалізації мобільної системи було обрано новий фреймворк flutter. Реалізований прототип буд доступний на платформах Android та iOS.

2 МЕТОД ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ЩОДО ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ РУХУ АВТОТРАНСПОРТУ

2.1 Модель проведення дослідів

Метою даного дослідження є порівняння та вибір найбільш оптимального методу штучного інтелекту за деякими критеріями для вирішення проблеми поліпшення безпеки руху. Сьогодні вирішенням подібних задач займається теорія прийняття рішень (ТПР).

Теорія прийняття рішень – це область досліджень, що використовує методи статистики, економіки, математики, психології та менеджменту для вивчення закономірностей вибору варіантів рішення різноманітних задач, а також визначення найбільш оптимальних варіантів із можливих рішень. Ця сфера досліджень застосовується в когнітивній психології, економіці, політології, а також для створення штучного інтелекту.

Альтернатива – це один із існуючих варіантів рішень, що може бути приманою. У випадку, коли множина альтернатив порожня, вибору не існує. Для вирішення задачі такого типу необхідно мати хоча б дві альтернативи. Задачі прийняття рішень можливо класифікувати наступним чином:

- класифікування альтернатив на упорядковані за якістю групи;
- впорядкування альтернатив (визначення порядку у множині альтернатив);
- вибір кращої альтернативи.

Альтернативи можуть бути попередньо заданими, конструйованими в процесі прийняття рішень, отриманими після вироблення правила прийняття рішень.

Критерій – це ознака, на підставі якої проводиться оцінка. Критерій – є способом вираження відмінностей між альтернативами. Критерії повинні відповідати наступним вимогам:

- повнота (критерії забезпечують адекватність оцінки досягнення мети рішення);
- операційність (критерії чітко сформульовані);
- декомпозиційність (критерії мають можливість бути структурованими);
- достатність (критерії не мають надмірності);
- мінімальність (набір критеріїв мінімально необхідний для здійснення оцінки);
- вимірність (критерії дають кількісну або якісну оцінку ступеня досягнення мети).

Процес прийняття рішення – це послідовність процедур, що приводять до знаходження рішення. Він складається наступної послідовності дій:

- визначення проблеми та постановка задач;
- описання властивостей зовнішнього середовища для визначення можливих альтернатив вибору;
- визначення характеристики рішення та формалізація характеристики до рівня критеріїв;
- вибору способу оцінку;
- оцінювання якості альтернатив, порівняння їх між собою, а також вибір однієї чи декількох альтернатив, що найбільш повно відповідають поставленій меті [22].

У разі необхідності якісного оцінювання сильних сторін альтернатив, то необхідно покладатися на множинні або парні порівняння, класифікацію та ранжування.

У контексті цієї роботи, основною задачею дослідження є пошук оптимального (найкращого) рішення при наявності різноманітних критеріїв

оптимальності. У випадку, коли поставлена задача може бути вирішена засобами математичного програмування, то такі задачі називають задачами багатокритеріальної оптимізації. Завдання подібного типу можуть носити нелінійний та лінійний характер. Схему прийняття рішень (див. рис. 2.1) можна описати наступними кроками: визначення мети дослідження, визначення альтернатив вибору, оцінка обраних альтернатив, визначення принципу порівняння та підсумок результатів.



Рисунок 2.1 – Схеми процесу прийняття рішень

Основна задача дослідження цієї роботи відповідає задачі прийняття оптимального рішення із заданою множиною альтернатив та визначеними принципами оптимальності. Подібна задача називається загальною задачею оптимізації. Ідея такої задачі полягає у визначенні набору альтернатив, що є "найкращими" для вирішення проблеми. Цей метод є особливо відповідним, коли принцип оптимальності задається скалярною функцією вибору і в такому випадку ми маємо звичайну задачу оптимізації.

У випадку, коли принцип оптимальності визначається набором критеріальних функцій, ми маємо багатокритеріальну задачу оптимізації. Задачу, що має відомий набір альтернатив та явно визначений принцип оптимальності називають задачею вибору.

Для розв'язання загальної оптимізаційної задачі прийняття рішень, беруть участь три групи людей: особи, які приймають рішення (ОПР), експерти (Е) та консультанти (К).

У ситуаціях, що є найпростішими, ОПР може виступати у трьох ролях, у складніших випадках ОПР має можливість поєднувати функції аналітика та звертатися до спеціалістів, що мають вузький профіль для вирішення особливих проблем. Загалом ОПР звертається для вирішення особливих питань звертаються до аналітиків-консультантів, котрі, у свою чергу, звертаються для консультації до експертів.

Експертом – людина, що є спеціалістом у своїй галузі та володіє спеціальною інформацією про задачу, не маючи прямої відповідальності за результати її розв'язання. Експерти допомагають ОПР на всіх стадіях постановки й розв'язання задачі прийняття рішення.

Консультантами називаються спеціалісти, із теорії прийняття рішень, які розробляють математичну модель задачі прийняття рішень, процедуру прийняття рішень, а також організують роботу експертів та ОПР.

У прикладних задачах, формалізація кожного етапу, процесу прийняття рішень мають деякі, часом складні, проблеми. По-перше, з'являється проблема правильної постановки мети та інструментів для її досягнення. Цілком можливо допустити помилку та поставити недосяжну мету. Особливий інтерес привертає формалізація загальної задачі прийняття рішення (ЗЗПР) та описання її, використовуючи мову математики, для моделювання прикладних ситуацій прийняття рішень. Цілком можливо, що описана математична модель покаже, що розв'язку поставлена задача не має. У такому випадку поставлена мета не досяжна.

Після етапу визначення мети та методів її досягнення, наступною проблемою, котру необхідно вирішити – є проблема визначення множини альтернатив – варіантів вибору, націлених на досягнення поставленої мети.

Перша проблема, котру необхідно вирішити під час другого етапу – побудова найбільш повного списку альтернатив. Бувають випадки, коли пропуск однієї з альтернатив, призводить до цілком не правильного висновку, а іноді навіть до неможливості розв'язання задачі.

Наступною проблемою є проблема оцінки альтернатив, визначення ймовірних результатів вибраної дії. Зазвичай, визначені оцінки альтернатив несуть лише особистий характер, так як вони формуються на базі обробленої інформації експерта. Навіть якщо є можливим оцінювати альтернативи, використовуючи деякі об'єктивні процедури, постає проблема визначення найважливіших аспектів оцінки кожної альтернативи. Неврахування хоча б єдиного критерію для оцінки альтернатив може призвести до негативних наслідків.

Останньою проблемою вирішення загальної задачі прийняття рішень є визначення принципу порівняння альтернатив та на його основі – принципу оптимальності. У попередньому етапі критерії були визначені числовими оцінками. Під час цього етапу вибір принципу оптимальності зводиться до вибору критеріїв оптимізації, які максимально відповідають меті ЗЗПР.

У випадку, коли мета ЗЗПР має декількома числових критеріїв задача зводиться до задачі багатокритеріальної оптимізації. У такому випадку, необхідно визначити принцип на основі якого будуть порівнюватися для визначення кращої альтернативи.

Деякі критерії можуть підсилювати один одного, інші суперечити, треті –не мають впливу один до одного. Процес вирішення багатокритеріальних задач оптимізації неминуче пов'язаний з експертними оцінками самих критеріїв та взаємин між ними. Задачі багатокритеріальної оптимізації мають наступний ряд методів:

- оптимізація одного визнаного найбільш важливим критерію;

- упорядкування множини критеріїв та послідовна оптимізація кожного з них;
- зведення багатьох критеріїв до одного шляхом введенням вагових коефіцієнтів для критеріїв. Чим важливіший критерій тим більш високу вагу отримує.

Розглянемо методи рішення задач прийняття рішень, основною ідеєю якої є зведення початкової багатокритеріальної задачі оптимізації до скалярної за допомогою введення деякого узагальненого критерію.

Всі методи рішення подібного типу вирішуються за наступною схемою:

- абсолютно всі обрані критерії нормують. Це означає, що їх приводять до порівнянного безрозмірного вигляду;
- після нормування критеріїв відбувається їх «згортання» в одну цільову функцію. Така функція має ім'я "узагальнений критерій". Під час згортання враховують відносну важливість критеріїв за допомогою вагових коефіцієнтів;

У результаті вищеперерахованих дій багатокритеріальна задача оптимізації зводиться до звичайної задачі оптимізації за одним критерієм.

Сьогодні найбільш поширеними згортками є:

- мультиплікативна згортка критеріїв. Мультиплікативна згортка базується на принципі справедливої сатисфакції відносних змін приватних критеріїв. Перевагою такого виду згортки критеріїв є спрощення задачі аналізу за рахунок зведення її до однокритеріальної задачі. Недоліком є невиправдана компенсація одних ефектів іншими та неможливість розділити та порівняти такі ефекти;
- узагальнення визначених критеріїв, використовуючи середнєзважену функцію. Серед таких критеріїв особливо виділяється лінійна згортка критеріїв. Лінійна згортка критеріїв є зручною у використанні, а також дозволяє зберігати лінійність результуючих функцій. Це означає, що у випадку коли вихідні критерії лінійні, ми отримуємо результуючий критерій, котрий також буде лінійним [21].

Розглянемо перший пункт детальніше. Мультиплікативною згортою є добуток критеріїв. Для такого випадку перед перемноження критеріїв можна звести їх до рівня тим більшого, чим більше важливість критерія.

Слід зазначити, що мультиплікативна згортка виправдана лише у тому випадку коли усе критерії невід'ємні. У зворотньому випадку властивість "зворотнього знаку" (мінус на мінус дає плюс) зіпсує обчислення, зробивши «позитивне» значення згортки з двох завідомо негативних критеріїв.

У тому випадку, коли лише один з критеріїв має від'ємне значення парадоксу подібного роду не буде. Це означає, що ми можемо користуватися методом мультиплікативної згортки.

Слід враховувати, випадок коли один з критеріїв дорівнює нулю. У такому разі мультиплікативна згортка дорівнює нулю теж. Для адитивної згортки таке правило не виконується.

Отже, ми маємо m альтернатив та n критеріїв. Побудуємо матрицю рішень $n \times m$ (див в таблицю 2.1).

Значення альтернатив заповнюються у стовпчику, що мають j індекс. Значення для критеріїв заносяться до стовпчиків, що мають i індекс.

Мультиплікативна згортка з нормалізуючими факторами описується наступним законом

$$Z = \max \prod_{i=1}^m a_i x_{ij} \quad (2.1)$$

де α_i – нормуючі множники, x_{ij} – значення критерію для кожної альтернативи в матриці рішень $n \times m$ (див. табл. 1).

Таблиця 2.1 – Матриця прийняття рішень

	Альтернатива 1	...	Альтернатива j
Критерій 1	x_{11}	...	x_{j1}

	Альтернатива 1	...	Альтернатива j
...
Критерій i	x_{1i}	...	x_{ji}

Базою мультиплікативної згортки є наступний постулат: «низька оцінка принаймні одного критерію тягне за собою низьке значення функції корисності».

Вагові коефіцієнти критеріїв – це параметр, який відображає значимість даного чинника або показника в порівнянні з іншими чинниками, що впливають на досліджуваний об'єкт або явище.

Зазвичай вага встановлюється на основі інтуїтивного ідеї відносної важливості критеріїв. Як правило, людина (експерт), що встановлює значення коефіцієнту не може безпосередньо надати правильну чисельну вагу критеріям.

2.2 Критерії вибору методу

Для визначення оптимальної альтернативи, котра є найбільш відповідною до вирішення поставленої задачі, спочатку, необхідно визначити ряд найважливіших критеріїв відповідно до котрих буде здійснюватися вибір методу для аналізу даних отриманих про стан водія під час руху транспортного засобу.

Під час огляду та аналізу предметної області стало зрозуміло, що цільовою платформою цієї системи – є мобільна платформа. Мобільна платформа значно відрізняється від звичайних персональних комп'ютерів.

Головні відмінності мобільних телефонів:

- обмежена продуктивність процесорів;
- обмежений об'єм оперативної пам'яті;
- джерело живлення має обмежений запас заряду.

Також слід звернути увагу наскільки легко та комфортно буде реалізовувати ту чи іншу альтернативу. Треба чітко розуміти наскільки вибрана альтернатива легка у підтримці, надійна та наскільки точно вона може визначати критичні випадки.

Відповідно до поставленої задачі та вищесказаного можна визначити наступні критерії:

- низькі вимоги до пам'яті;
- низькі вимоги до задіяння процесору;
- можливість коректно працювати без інтернету;
- легкість реалізації;
- легкість підтримки;
- надійність;
- точність передбачення.

Розглянемо визначені критерії більш детальноше.

Низькі вимоги до пам'яті. Мається на увазі оперативна пам'ять (RAM). Оперативна пам'ять — це швидкодійна пам'ять, призначена для запису, зберігання та читання інформації у процесі її обробки. Цей тип пам'яті є невід'ємною частиною будь якої обчислювальні системи. Сучасні мобільні телефони стрімко розвиваються і поступово наздоганяють персональні комп'ютери за потужністю, але все ще достатньо сильно відстають від потужностей звичайних комп'ютерів. Так, наприклад, сучасний смартфон Samsung S8 має лише 4096 мегабайти оперативної пам'яті. Об'єм оперативної пам'яті найсучасніших мобільних пристроїв досягає 8192 мегабайт, але такі пристрої ще занадто дорогі для масового користування.

Низькі вимоги до задіяння процесору. Процесор та екран мобільного телефону є основними споживачами електроенергії. Тривалість автономної роботи мобільного телефону дуже залежить від того наскільки інтенсивно користувач використовує його. Так, наприклад, мобільний телефон у режимі очікування може працювати до декількох діб, а у режимі перегляду відео та користування

інтернетом знижується до 6 годин. Тому вибраний метод повинен дуже дбайливо відноситись до ресурсів процесору, адже це може призвести до швидкого витрату заряду батареї, що зробить використання додатку набагато дискомфорним. Користувачу завжди доведеться носити з собою засоби поповнення енергії акумулятору.

Можливість коректно працювати без інтернету. Цей критерій означає, що вся логіка аналізу даних та прийняття рішень повинна знаходитись безпосередньо на мобільному пристрою. Водій може пересуватись у місцях з обмеженим покриттям зв'язку і навіть в таких випадках система повинна працювати коректно. Також постійний зв'язок із сервером є одним з джерел витрачання електричного заряду акумулятору.

Легкість реалізації. Деякі програмні методи складні у реалізації, а надмірна складність програмного забезпечення призводить до більше складного супроводу, а також призводить до підвищення ризику помилок у кодї. Одним з правил об'єктно орієнтованого програмування є принцип "KISS" (keep it simple stupid), що вимагає підтримувати простоту коду. Реалізація даної системи не є винятком, тому необхідно притримуватись цього правила.

Легкість підтримки. Розробка програмного забезпечення складається з деякого набору етапів, одними з котрих є етап реалізації системи, а наступним є етап підтримки. Обраний метод для аналізу даних не повинен ускладнювати процес підтримки. Окрім точного визначення критичних показників, обраний метод повинен забезпечувати легкість у підтримці програмного додатку.

Надійність. Надійність – це властивість системи працювати без збоїв, у встановлених межах значення всіх параметрів, що характеризують здатність виконувати необхідні функції в заданих умовах застосування, технічного обслуговування, зберігання і транспортування.

Точність передбачення. Різні методи мають різноманітні моделі, котрі мають прямий вплив на точність висновку. Головна задача методу аналізу даних показників стану водія, у будь-який момент часу, чітко та правильно давати

відповідь на питання: "Чи існують які-небудь загрози на даний момент для безпеки водія?"

2.3 Проведення порівнянь альтернатив

У попередньому розділі було проаналізовану наукову літературу та визначено найбільш придатні методи штучного інтелекту для аналізу даних про стан водія. До них відносяться: дерева рішень, продукційні моделі, фреймові моделі, логічна регресія.

Для кожного методу було розглянуто та проаналізовану літературу відповідно до визначеним атрибутами якості. У таблиці 2.2 описані висновки щодо дослідження усіх методів. Кожен стовпець таблиці 2.2 – назва методу аналізу даних, а кожен рядок – один з атрибутів якості програмного забезпечення. У таблиці представлено шкалу від 1 до 10, де 10 балів – це "Відмінно", а 1 бал "Погано". Такий підхід дозволив провести пряме порівняння між методами аналізу даних.

Необхідно підкреслити, що існують інші методи аналізу даних, можливо, навіть більш потужні, але слід пам'ятати, що цільовою платформою цього додатку є мобільна. Саме цьому серед усіх методів залишилося лише чотири, що описані вище. Альтернативи: дерева рішень (A1), логістична регресія(A2), продукційна модель (A3), модель на основі фреймів (A4). Дані таблиці були базуються на попереднього огляду літератури та на оцінках експертів.

Таблиця 2.2 – Матриця рішень для методів аналізу даних

Критерій	Альтернативи				Нормуючі коефіцієнти
	A ₁	A ₂	A ₃	A ₄	
Вимоги до пам'яті	3	4	10	9	1/26
Вимоги задіяння процесору	4	4	9	8	1/25
Коректність роботи без інтернету	7	7	9	9	1/32
Простота реалізації	5	6	8	10	1/29
Легкість підтримки	7	7	8	9	1/31
Надійність	8	8	7	8	1/31
Точність передбачення	7	7	8	9	1/31

Матриця рішень для методів аналізу даних дозволяє наглядно продемонструвати оцінки експертів, а також зручно підрахувати нормуючі коефіцієнти. У таблиці 3, нижче, відображені результат мультиплікативної згортки з нормуючими коефіцієнтами.

Таблиця 2.3 – Результат мультиплікативної згортки з використанням нормуючих коефіцієнтів

A ₁	A ₂	A ₃	A ₄
12.21 E-08	13.21 E-08	20.93 E-08	21.88 E-08

У результаті обчислення мультиплікативної згортки з нормалізуючими факторами стали відомо, що найкращий варіант – це A4. Це означає, що для реалізації системи необхідно використовувати модель на основі фреймів, що є фундаментом систем заснованих на правилах. Ця модель має перевагу перед продукційними моделями, так як знижує ризик наявності конфліктів у правилах.

2.4 Моделювання системи на основі правила та знань

Під час дослідження методів обробки даних для реалізації інтелектуальної системи стало відомо, що методи машинного навчання не підходять для реалізації автономного додатку на базі мобільних телефонів. Причинами несумісності є велика потреба ресурсів, проблема перенавчання алгоритмів, що може призвести до не правильної роботи системи, а також відсутність великої кількості даних для навчання системи.

Альтернативою машинному навчанню є моделі на основі правил, що дозволяють реалізувати систему засновану на правилах (rule-based system). Класичним прикладом системи, заснованої на правилах, є доменна специфічна експертна система, яка використовує набір існуючих правил у системі для розрахунків, виборів або класифікації.

Експертна система – комп'ютерна система, здатна частково замінити фахівців-експертів у вирішенні проблемної ситуації. Наприклад, експертна система може допомогти лікарю вибрати правильну діагностику на основі набору симптомів або допомогти гравцю вибрати тактичні ходи для гри.

Моделі представлення та опрацювання знань є фундаментом будь-якої програмної системи заснованої на правилах [23]. Такі системи мають чотири основні компоненти (див. рис. 2.2).

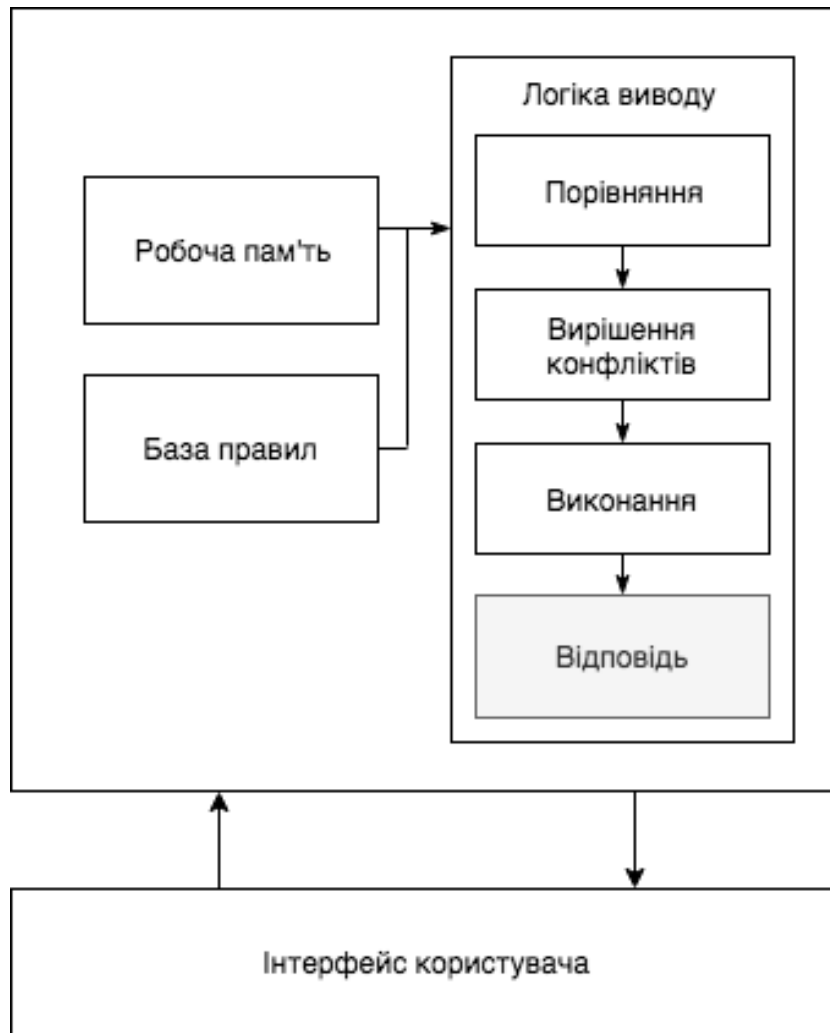


Рисунок 2.2 – Компоненти системи заснованої на правилах

1. список правил або бази правил, який є специфічним типом бази знань;
2. логіка виводу, яка виводить інформацію або вживає дії, засновані на взаємодії вхідних даних і бази правил. Також такий компонент ще називається інтерпретатором, що виконує програму системи;
3. тимчасова робоча пам'ять у якій знаходиться робочі екземпляри правил;
4. інтерфейс користувача або інше підключення до зовнішнього світу, через який надходять і надсилаються вхідні та вихідні сигнали.

Компонент логіки виводу працює за наступним циклом:

1. Порівняння. Під час виконання запиту, аргументи запиту проходять через набір правил для виявлення тих правил, що відповідають запиту. У результаті виконання цієї процедури може бути 2 випадки. Перший випадок, коли у результаті порівняння виявляється лише одне правило, котре може бути виконано. У цьому випадку виконання програми переходить до кроку 3. У другому випадку, коли існує декілька взаємовиключних правил виконання програми переходить до кроку 2 для вирішення конфліктів;

2. Вирішення конфліктів. Під час виконання кроку 2 всі конфліктуючі правила проходять через функції, що розв'язують їх. У результаті виконання цього кроку система отримує набір не взаємовиключних правил, що переходять до третього кроку.

3. Виконання. На цьому кроці система отримує вільні від конфліктів правила, що можуть бути виконані.

До переваг такої системи можна віднести легкість у реалізації, прозорий процес відладки, легкість контролю поведінки системи та її визначеність. Нажаль, такий підход має деякі недоліки. Наприклад, необхідно підтримувати правила, добавляти нові, досить тяжко покрити великі домени.

3 РОЗРОБКА ПРОТОТИПУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Загальні відомості про систему

Під час аналізу програмної системи для поліпшення безпеки руху було встановлено, що така система повинна повноцінно функціонувати, незалежно від зовнішніх сервісів. Реальна система повинна інтегруватись за допомогою протоколу безпроводної мережі Bluetooth з датчиками, котрі зчитують інформації про стан водія та стан руху. Створення таких датчиків виходить за рамки цієї роботи, тому було прийнято рішення реалізувати mock модулю передавача даних, що емулює роботу.

Діаграма розгорткування реалізованої системи (див. рис. 3.1) складається з одного компоненту. Цільовим пристроєм на якому розгортається реалізований артефакт – це мобільний пристрій.

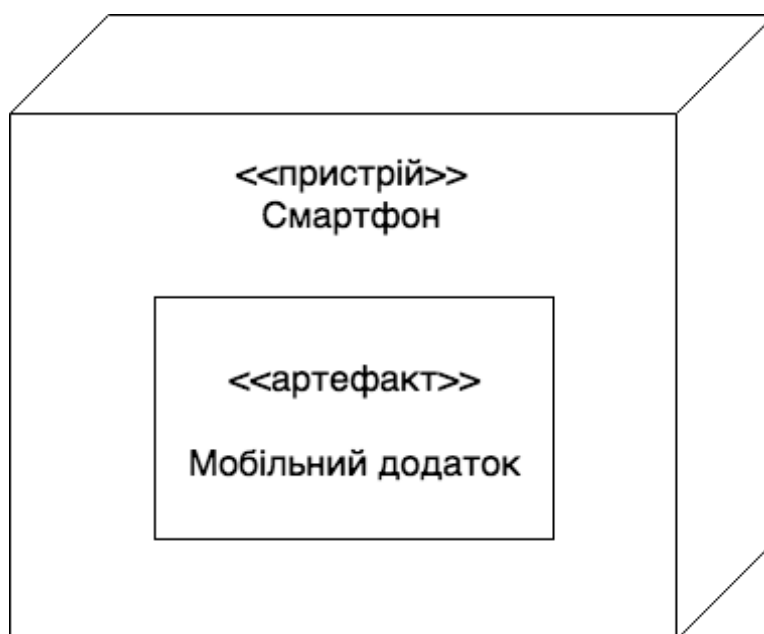


Рисунок 3.1 – Діаграма розгорткування реалізованої системи

Сам артефакт представляє з себе кросплатформений мобільний додаток написаний на новому фреймворку "Flutter" від компанії "Google".

Мобільний додаток складається (див. рис. 3.2) чотирьох основних компонентів: інтерфейс користувача, модуль аналізу даних та прийняття рішень, емулятору приймача даних та бази знань.

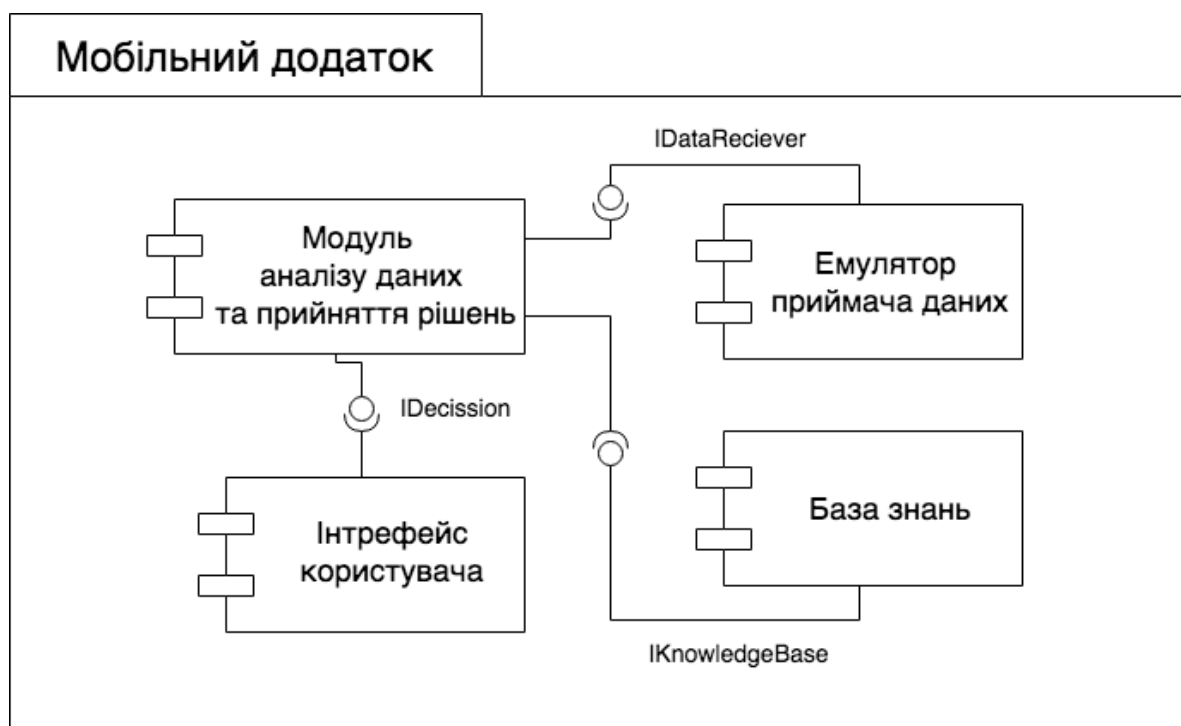


Рисунок 3.2 – Діаграма компонентів мобільного додатку

Інтерфейс користувача – це набір flutter віджетів, котрі представляють собою різноманітні UI засоби керування, забезпечуючи зручний засіб взаємодії користувача та реалізованої системи. До таких віджетів належать різноманітні кнопки, форми вводу, текстові бірки та інші.

База знань – це компонент, що використовується для зберігання складної структурованої та неструктурованої інформації, котра далі використовується програмною системою. У нашому випадку, база знань – це набір правил, котрі допомагають визначити у якому стані знаходиться водій та процес руху.

Емулятор приймача даних – це невеликий компонент, що дозволяє змоделювати роботу реальних сенсорів, що передають стрім даних до мобільного додатку. В реальній системі, цей компонент буде, замість імітування приймання даних, приймати справжні данні зі справжніх датчиків.

Останній компонент цієї системи – це модуль аналізу даних та прийняття рішень. Цей модуль зчитує правила з бази знань та отримує данні з приймача даних, після цього алгоритм виконує процедуру аналізу даних, визначає у якому стані знаходиться водій та передає результат до інтерфейсу користувача.

3.3 Інтерфейс та функціонал

Мета реалізованої програмної системи поліпшити безпеку руху транспортних засобів на вулицях міст. Додаток емулює з'єднання з зовнішніми датчиками, котрі зчитують данні про навколишнє середовище та передають до програмної системи.

Реалізований прототип надає наступні функціональні можливості (див. рис. 3.3):

- з'єднання з датчиками;
- відображення біометричних показників: температури тіла, тиску крові, пульсу;
- відображення даних про транспортний засіб. Наприклад: час руху та швидкість автомобілю;
- можливість заповнити персональні данні про водія;
- імітація виклику швидкої у разі екстреної ситуації;
- визначення поганого самопочуття водія на основі правил та отриманих даних. Наприклад: водій керує автомобілем протягом декількох годин в ночі та поступово його пульс сповільнюється, неначе водій засинає;

– генерування порад, щодо поліпшення безпеки керування транспортною засобу. Наприклад: система визначила, що водій засинає та видала гнучкий звуковий сигнал, для того щоб привернути увагу водія, а також запропонувала відпочити.

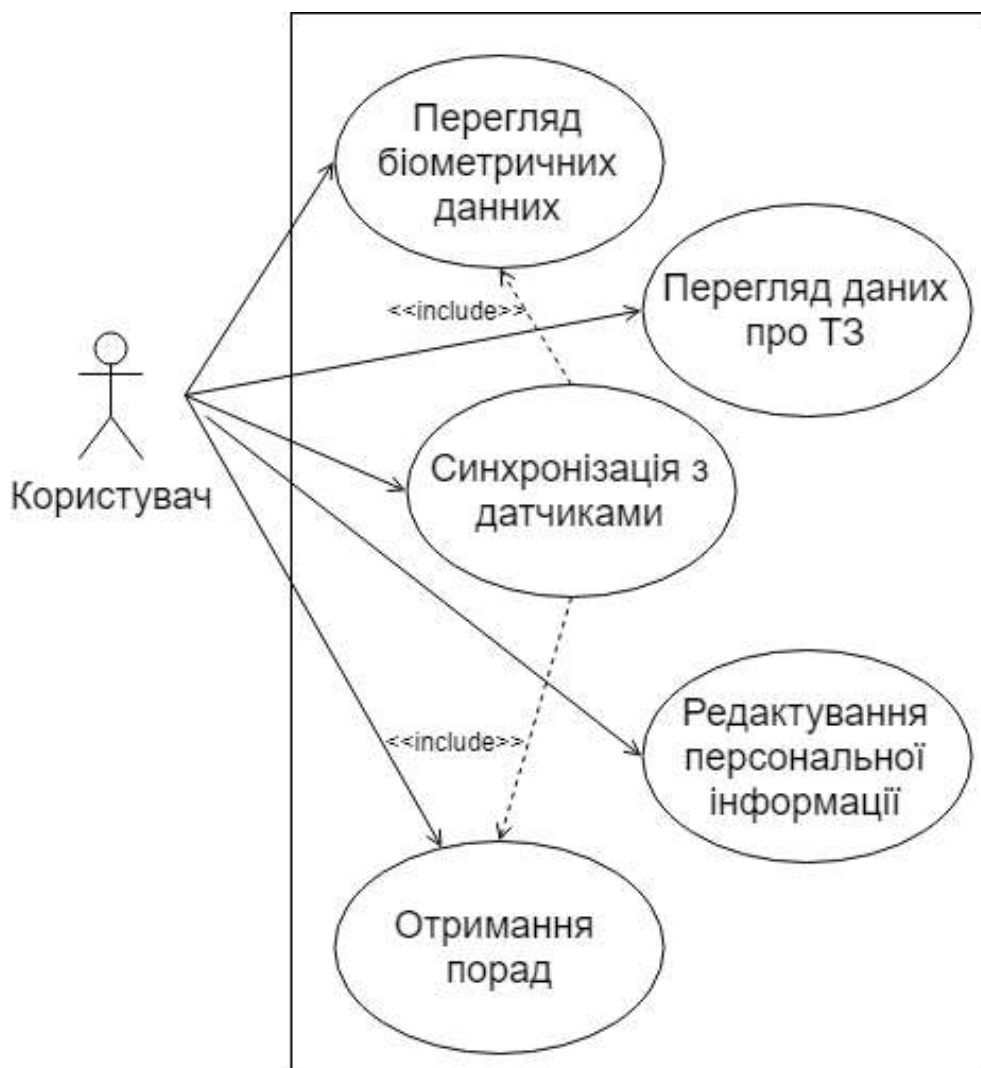


Рисунок 3.3 – Діаграма варіантів використання системи

Система поліпшення безпеки руху представляє собою мобільний додаток для смартфона водія, котрий завжди знаходиться поряд з водієм. Вибір цієї платформи дозволив зробити систему більш доступною для користувачей, так як сьогодні майже неможливо уявити людину, котра не має мобільного телефону.

Для того щоб розпочати використання системи, спочатку, користувачеві необхідно пройти процес авторизації за допомогою сторінки логіну (див. рис. 3.4). Сторінка логіну з інтуїтивно доступним інтерфейсом відкривається одразу після запуску додатку. Сторінка складається всього з трьох віджетів: двох для вводу тексту та однієї кнопки.



Рисунок 3.4 – Форма авторизації

Користувач повинен ввести адрес своєї пошти та пароль у відповідні поля. Після цього необхідно натиснути на кнопку "Вхід". У випадку правильно введених даних користувач отримує доступ до повного функціоналу системи. Якщо користувач помилився – то він побаче відповідну помилку.

Головна сторінка (див. рис. 3.5) складається з 4 секцій: інформація про користувача, стан водія, данні відповідаючи стану руху та кнопка виклику екстреної допомоги.

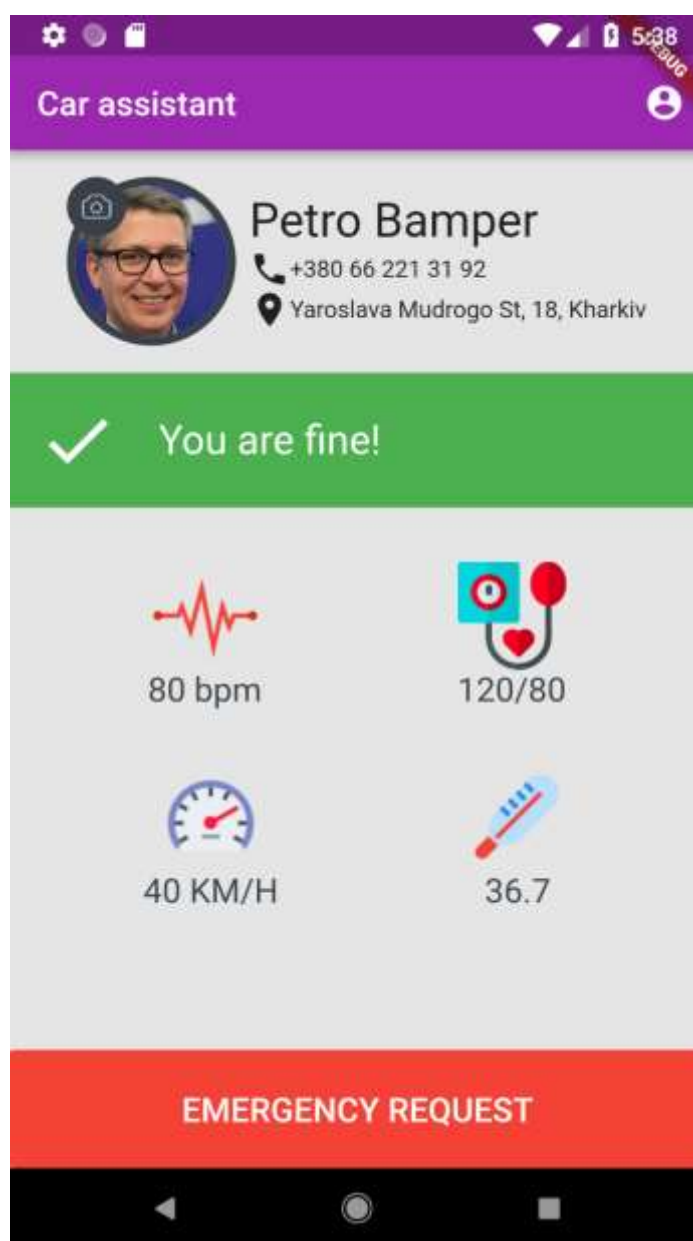


Рисунок 3.5 – Головна сторінка додатку

На першій секції відображено фото користувача, його ім'я, номер мобільного телефону та місце проживання. Ці данні можуть бути додані до повідомлення про надзвичайну ситуацію. Наступна секція відображає стан керування транспортним засобом. У разі виникнення проблем водій побаче (див. рис. 3.6) відповідне повідомлення.

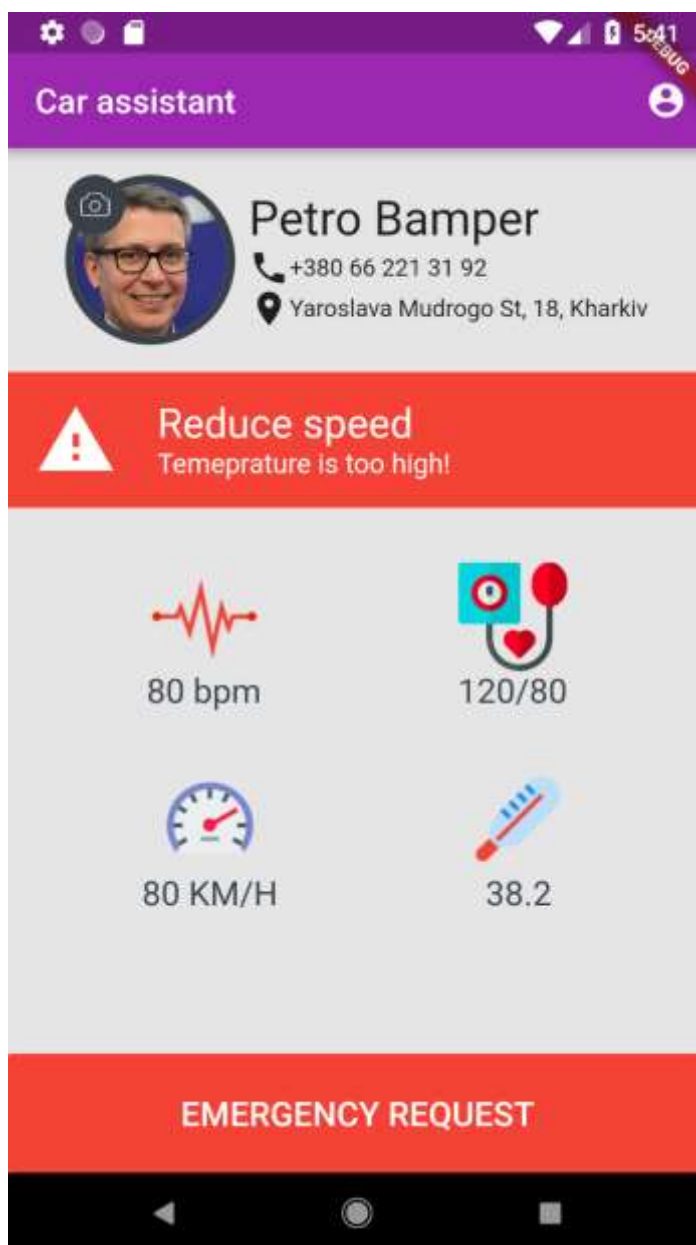


Рисунок 3.6 – Сторінка під час нештатної ситуації

Тобто, стан руху може бути безпечним або небезпечним. У випадку, коли стан руху безпечний користувач побачить зелений колір екрані свого телефону та відповідне повідомлення. У разі виникнення проблем, користувач побаче червоний колір та система запропонує дії, щодо вирішення проблемної ситуації, що виникла під час керування транспортним засобом.

Наступною секція – це секція відображення даних, про нинішній стану руху. Ця секція відображає дані отримані з емулятору приймача даних. Наприклад, у цій секції відображаються данні про швидкість транспортного засобу, температури тіла водія, кров'яного тиску та пульсу.

Остання секція головної сторінки – це секція виклику екстреної допомоги. Якщо водій відчує себе зле він може натиснути на велику червону кнопку та найостанніші данні відправляться до найближчої екстреної служби з повідомлення про надзвичайну ситуацію. Також система може самостійно відправити запит до екстрених служб у разі виникнення дорожньо-транспортної пригоди. Наприклад, водій рухався зі швидкістю 80 кілометрів на годину і в наступний момент данні про температуру тіла, пульсу та тиску перестають поступати до системи. У такому випадку алгоритм одразу визначе надзвичайну ситуацію та зроби запит до екстреної служби.

У разі виникнення екстреної ситуації, важливою інформацією є інформація про родичів. Реалізований додаток має сторінку редагування персональних даних (див. рис. 3.7) де користувач має можливість ввести додаткову інформацію про родичів, з котрими рятувальники зможуть зв'язатися та проінформувати. Також родичі можуть надати додаткову інформацію, що посприє надання допомоги у разі виникнення нещасного випадку.

Сторінка редагування даних складається з набору текстових засобів вводу, що дозволяють відредаткувати його ім'я, номер телефону, домашню адресу, додаткову інформацію про близьких.

Також сторінка дозволяє змінити пароль. Для цього користувачеві необхідно ввести його поточний пароль, новий пароль та підтвердження нового паролю. Така

схема дозволить переконатися у тому, що користувач не допустив помилки під час вводу нового паролю.

У майбутньому можна додати додаткові поля для вводу додаткової інформації про водія, наприклад про деякі його захворювання, що може дозволити більш точно контролювати його стан.



The image shows a screenshot of a mobile application interface titled "User info". The form consists of several text input fields stacked vertically. The fields are labeled: "Your name", "Your phone number", "Your address", "Your relative name", "Your relative phone number", "Current password", "New password", and "Repeat new password". The application is running on an Android device, as indicated by the navigation bar at the bottom. In the top right corner, there is a red banner with the word "Crash" written on it, indicating that the application has encountered an error. The status bar at the top shows the time as 2:58 and various system icons.

Рисунок 3.7 – Вигляд сторінки під час нештатної ситуації

Після завершення редагування користувач повинен натиснути на кнопку "Зберегти" у правому верхньому куті.

ВИСНОВКИ

У результаті виконання дипломного проекту було досягнуто поставлені цілі, а саме: досліджено та визначено найбільш придатний метод штучного інтелекту для реалізації логіки аналізу даних про стан водія у мобільному додатку для поліпшення безпеки руху, розроблено підхід для реалізації системи заснованої на правилах, а також реалізовано прототип такої системи на базі мобільного додатку, що дозволяє поліпшити безпеку руху.

У ході дослідження було виявлено, що найбільш придатною платформою для даної системи – є мобільна. Під час дослідження можливих альтернатив було визначено, що використання машинного навчання в мобільних додатках стикається з багатьма проблемами, наприклад: потужності процесора і пам'яті мобільного пристрою недостатньо для обробки та обчислень великих обсягів даних, що робляться під час навчання моделі. Такі обчислення будуть виконуватися досить довго, що потребує витрат енергії та може призводити до значного розігріву батареї пристрою, а також до її розрядки.

Тому для логіки аналізу даних системи було використано систему заснованої на правилах, а у якості моделі відображення даних було використано модель фреймів. Такий підхід може виконувати функції аналізу даних про стан водія з невеликими затратами ресурсів пристрою.

Визначення правильного методу аналізу даних, дозволило реалізувати мобільний додаток, що контролює стан водія та процес руху транспортного засобу. Подібний додаток повинен поліпшити безпеку водія та процес пересування транспортних засобів на вулицях міст. Реалізований прототип підходить для подальшого удосконалення та вводу у експлуатацію.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Статистика ДТП в Україні за 2017 рік / Патрульна поліція України. URL: <http://patrol.police.gov.ua/wp-content/uploads/2018/06/DTP-12-2017.xls> (дата звернення: 15.03.2019).
2. Road traffic injuries / World Health Organization. URL: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries> (дата звернення: 15.03.2019).
3. Під Києвом автобус з дітьми потрапив в ДТП - водій заснув за кермом / Відео-Новини. URL: <https://fraza.ua/video/270520-pod-kievom-avtobus-s-detmi-popal-v-dtp-voditel-usnul-za-rulem> (дата звернення: 15.03.2019).
4. Проблеми організації підготовки водіїв в навчальних закладах України / Збірник наукових робіт. URL: <https://www.sworld.com.ua/konfer31/49.pdf> (дата звернення: 17.03.2019).
5. Статистика / Патрульна поліція України. URL: <http://patrol.police.gov.ua/statystyka/> (дата звернення: 18.03.2019).
6. Motor vehicles per 1000 people: Countries Compared / International statistics. URL: <https://www.nationmaster.com/country-info/stats/Transport/Road/Motor-vehicles-per-1000-people/> (дата звернення: 18.03.2019).
7. How do WA road fatalities compare to Australia? / RAC WA. URL: https://rac.com.au/car-motoring/info/state_wa-fatalities (дата звернення: 18.03.2019).
8. Teen Drivers' Perceptions of Inattention and Cell Phone Use While Driving / The National Center for Biotechnology Information advances science. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4594631/> (дата звернення: 18.03.2019).
9. Мобільний додаток Drive Safe & Save / Apps on Google Play. URL: <https://play.google.com/store/apps/details?id=com.statefarm.dssm> (дата звернення: 20.03.2019).

10. Mercedes-Benz Digital Challenge—Winners, Projects and Prizes! / Medium. URL: <https://medium.com/high-mobility/mercedes-benz-challenge-winners-projects-and-prizes-7a743b4fbfdb> (дата звернення: 25.03.2019).
11. Safe Drive Systems: про компанію / LinkedIn. URL: <https://www.linkedin.com/company/safe-drive-systems/about/> (дата звернення: 25.03.2019).
12. Rajaraman A., Ullman J. D. Mining of Massive Datasets – Cambridge: Cambridge University Press, 2011. — 328 с.
13. Литвин В. В. Методи та засоби інженерії даних та знань – Магнолія, 2012. — 248 с.
14. Воронцов К. Е. Математические методы обучения по прецедентам (теория обучения машин) / Портал машинного обучения. URL: <http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf> (дата звернення: 25.03.2018).
15. Goodfellow I., Bengio Y., Courville A. Deep Learning. Adaptive Computation and Machine Learning series — The MIT Press, 2016. — p. 775.
16. Shalev-Shwartz S., Ben-David S. Understanding Machine Learning: From Theory to Algorithms — Cambridge: Cambridge University Press, 2014. — p. 449.
17. Девятков В. В. Системы искусственного интеллекта — М.: Изд-во МГТУ им. Н. Э. Баумана, 2001. — 352 с.
18. 7 Types of Classification Algorithms / Analytics India magazine. URL: <https://www.analyticsindiamag.com/7-types-classification-algorithms> (дата звернення: 25.04.2018).
19. F. Hayes-Roth, N. Jacobstein. The State of Enowledge-Based Systems — Communications of the ACM, 1994. — p. 200.
20. Гаврилова Т. А., Хорошевский В. Ф. Базы знаний интеллектуальных систем. Учебник. — СПб.: Питер, 2000. — 400 с.
21. Горюнов Ю.Ю., Горюнова Т.Ю., Дружинин Д.В. Теория и методы принятия решений. Учебное пособие. — Пенза: РГУИТП, 2010. — 50 с.

22. Орлов А. И. Теория принятия решений. — М.: Март, 2004. — 656 с.
23. Ligeza A., Logical Foundations for Rule-Based Systems. — Springer, 2012. —
p. 312.

ДОДАТОК А

Лістинг коду програми

```

import 'package:flutter/material.dart';
import 'package:mobile/colors.dart';
import 'package:mobile/screens/home/screen.dart';
import 'package:mobile/screens/login_screen.dart';
import 'package:mobile/screens/user_details.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Car assistant',
      theme: ThemeData(
        primarySwatch: AppColors.purple,
        backgroundColor: AppColors.lightGray,
      ),
      home: LoginScreen(),
      routes: <String, WidgetBuilder>{
        '/login': (context) => LoginScreen(),
        '/home': (context) => Home(),
        '/user-details': (context) => UserDetails(),
      },
    );
  }
}

import 'package:flutter/material.dart';
import 'package:mobile/colors.dart';
import 'package:mobile/widgets/label.dart';
import 'package:mobile/widgets/text_input.dart';

class LoginScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Theme.of(context).backgroundColor,
      body: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        crossAxisAlignment: CrossAxisAlignment.center,
        children: [
          TxtLabel(
            text: 'Login',
            fontSize: 28,
            margin: EdgeInsets.only(bottom: 24),
          ),
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
            hint: 'Email',
          ),
        ],
      ),
    );
  }
}

```

```

    ),
    TextInput(
      margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
      hint: 'Password',
    ),
    Align(
      alignment: Alignment.bottomRight,
      child: Padding(
        padding: EdgeInsets.only(right: 8),
        child: RaisedButton(
          color: AppColors.purple,
          onPressed: () {},
          child: TextLabel(
            text: 'Log in',
            color: AppColors.white,
            fontSize: 20,
          ),
        ),
      ),
    ),
  ],
),
);
}

import 'package:flutter/material.dart';
import 'package:mobile/screens/home/help_btn.dart';
import 'package:mobile/screens/home/user_bio_data.dart';
import 'package:mobile/screens/home/user_info.dart';
import 'package:mobile/screens/home/user_status.dart';

class Home extends StatefulWidget {
  Home({Key key}) : super(key: key);

  @override
  _HomeState createState() => _HomeState();
}

class _HomeState extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Theme.of(context).backgroundColor,
      appBar: AppBar(
        title: Text('Car assistant'),
        actions: [
          IconButton(
            icon: Icon(Icons.account_circle),
            onPressed: () {
              Navigator.of(context).pushNamed('/user-details');
            },
          ),
        ],
      ),
      body: Column(

```

```

mainAxisAlignment: MainAxisAlignment.spaceBetween,
children: [
  Column(
    children: [
      UserInfo(),
      UserStatus(),
      UserBioData(),
    ],
  ),
  HelpButton(),
],
),
);
}
}

import 'package:flutter/material.dart';
import 'package:mobile/widgets/full_screen_scroll_view.dart';
import 'package:mobile/widgets/text_input.dart';

class UserDetails extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('User info'),
        actions: [
          IconButton(
            icon: Icon(Icons.check),
            onPressed: () => Navigator.of(context).pushNamed('/user-
details'),
          ),
        ],
      ),
      backgroundColor: Theme.of(context).backgroundColor,
      body: FullScreenScrollView(
        children: [
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
            hint: 'Your name',
          ),
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
            hint: 'Your phone number',
          ),
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
            hint: 'Your address',
          ),
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 32, 8, 0),
            hint: 'Your relative name',
          ),
          TextInput(
            margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
            hint: 'Your relative phone number',
          ),
        ],
      ),
    );
  }
}

```

```
),
  TextInput(
    margin: EdgeInsets.fromLTRB(8, 32, 8, 0),
    hint: 'Current password',
  ),
  TextInput(
    margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
    hint: 'New password',
  ),
  TextInput(
    margin: EdgeInsets.fromLTRB(8, 8, 8, 0),
    hint: 'Repeat new password',
  ),
],
);
}
```

