

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління  
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)  
(рівень вищої освіти)

Методи нейромережевого розпізнавання тексту на зображеннях  
(тема)

Виконав: студент 2 курсу, групи СКСм-22-1

Соколова В. К.  
(прізвище, ініціали)

Спеціальність 123 Комп'ютерна інженерія  
(код і повна назва спеціальності)


Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма \_\_\_\_\_  
Спеціалізовані комп'ютерні системи  
(повна назва освітньої програми)

Керівник доц. Хаханова Г. В.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

  
(підпис)

Чумаченко С. В.  
(прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Комп'ютерної інженерії та управління \_\_\_\_\_  
Кафедра \_\_\_\_\_ Автоматизації проектування обчислювальної техніки \_\_\_\_\_  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
Спеціальність \_\_\_\_\_ 123 Комп'ютерна інженерія \_\_\_\_\_  
(шифр і назва)  
Тип програми \_\_\_\_\_ Освітньо-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)  
Освітня програма \_\_\_\_\_ Спеціалізовані комп'ютерні системи \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:  
Зав. кафедри \_\_\_\_\_  
(підпис)  
«03» вересня 2023 р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові \_\_\_\_\_ Соколовій Вікторії Костянтинівні \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ Методи неймережевого розпізнавання тексту на зображеннях \_\_\_\_\_

затверджена наказом по університету від \_\_\_\_\_ 03.11.2023 р. № \_\_\_\_\_ 1288СТ \_\_\_\_\_

2. Термін подання студентом роботи до екзаменаційної комісії \_\_\_\_\_ 24.12.2023 р.

3. Вихідні дані до роботи \_\_\_\_\_

\_\_\_\_\_ Зображення \_\_\_\_\_

\_\_\_\_\_ Розпізнавання тексту \_\_\_\_\_

\_\_\_\_\_ Платформа Android Studio \_\_\_\_\_

\_\_\_\_\_ Мова програмування Kotlin \_\_\_\_\_

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

\_\_\_\_\_ Штучний інтелект \_\_\_\_\_

\_\_\_\_\_ Нейронні мережі \_\_\_\_\_

\_\_\_\_\_ Створення мобільного застосунку \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 16 слайдів

---



---



---



---



---



---


6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п. 1 )


| Найменування розділу | Консультант (посада, прізвище, ім'я, по батькові) | Позначка консультанта про виконання розділу |      |
|----------------------|---|---|------|
|                      |   | підпис                                      | дата |
|                      |   |   |      |
|                      |   |   |      |
|                      |   |   |      |

7. Дата видачі завдання 02.09.2023

### КАЛЕНДАРНИЙ ПЛАН

| №  | Назва етапів роботи                         | Терміни виконання етапів роботи | Примітка |
|----|---|---------------------------------|----------|
| 1  | Отримання завдання на кваліфікаційну роботу | 02.09.2023-06.09.2023           |          |
| 2  | Аналіз літератури за темою                  | 06.09.2023-30.09.2023           |          |
| 3  | Суть технічної проблеми                     | 30.09.2023-08.10.2023           |          |
| 4  | Постановка задачі                           | 08.10.2023-12.10.2023           |          |
| 5  | Існуючі методи для вирішення задачі         | 12.10.2023-20.10.2023           |          |
| 6  | Розробка алгоритму застосунку               | 20.10.2023-25.10.2023           |          |
| 7  | Програмна реалізація застосунку             | 25.10.2023-20.11.2023           |          |
| 8  | Тестування застосунку                       | 20.11.2023-30.11.2023           |          |
| 9  | Оформлення пояснювальної записки            | 30.11.2023-17.12.2023           |          |
| 10 | Оформлення графічної частини                | 17.12.2023-24.12.2023           |          |

Студент   
(підпис)

Керівник роботи  доц. Хаханова Г. В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи містить 63 сторінки, 41 рисунок, 28 джерел за переліком посилань.

МАШИННЕ НАВЧАННЯ, НЕЙРОН, НЕЙРОННІ МЕРЕЖІ, РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ, ШТУЧНИЙ ІНТЕЛЕКТ, ANDROID-РОЗРОБКА.

У кваліфікаційній роботі розглянуті основні можливості штучного інтелекту та нейронних мереж. Проведено дослідження методів та способів вирішення задач розпізнавання тексту з зображень. Проаналізовано існуючі інструменти для розробки. Написано мобільний застосунок на Android мовою Kotlin.

## ABSTRACT

The explanatory note to the qualification work contains 63 pages, 41 figures, 28 sources according to the list of references.

MACHINE LEARNING, NEURON, NEURAL NETWORKS, IMAGE RECOGNITION, ARTIFICIAL INTELLIGENCE, ANDROID DEVELOPMENT.

The qualification paper examines the main capabilities of artificial intelligence and neural networks. A study of methods and ways of solving problems of text recognition from images was conducted. The existing development tools were analyzed. A mobile application on Android was written in the Kotlin language.

## ЗМІСТ

|  |    |
|--|----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....                     | 7  |
| ВСТУП.....   | 8  |
| 1 АНАЛІЗ ЗАВДАННЯ.....   | 10 |
| 1.1 Суть технічної проблеми .....  | 10 |
| 1.2 Поняття, концепція і можливості штучного інтелекту .....                                 | 11 |
| 1.3 Поняття, концепція і можливості нейронних мереж.....                                     | 16 |
| 1.3.1 Біологічний нейрон .....   | 17 |
| 1.3.2 Штучний нейрон .....   | 19 |
| 1.3.3 Види нейронних мереж.....  | 21 |
| 1.3.4 Навчання нейронних мереж.....  | 26 |
| 1.4 Загальний алгоритм розпізнавання зображень .....   | 27 |
| 2 ПОСТАНОВКА ЗАДАЧІ ТА АНАЛІЗ ОБРАНИХ ЗАСОБІВ РОЗРОБКИ .                                     | 30 |
| 2.1 Постанова задачі.....  | 30 |
| 2.2 Методи для вирішення задачі .....  | 31 |
| 2.2.1 Розпізнавання за шаблоном .....  | 31 |
| 2.2.2 Розпізнавання за допомогою метрик.....   | 31 |
| 2.2.3 Розпізнавання за допомогою нейронних мереж .....                                       | 32 |
| 2.3 Програмна частина.....   | 32 |
| 2.4 Огляд існуючих рішень.....   | 33 |
| 2.5 Аналіз відомих мобільних операційних систем .....  | 38 |
| 2.6 Аналіз програмних засобів для розробки застосунків для операційної системи Android ..... | 39 |
| 2.7 Загальні поняття Android-розробки.....   | 40 |
| 3 РОЗРОБКА ПРОГРАМНОЇ ЧАСТИНИ .....  | 42 |
| 3.1 Загальна архітектура .....   | 42 |
| 3.2 Опис застосунку .....  | 43 |
| 4 ТЕСТУВАННЯ.....  | 48 |
| ВИСНОВКИ .....   | 60 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....  | 61 |

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

БД – база даних;

ОС – операційна система;

ПЗ – програмне забезпечення;

ШІ – штучний інтелект;

CNN – конволюційна нейронна мережа (англ. Convolutional Neural Network);

GAN – генеральна змагальна мережа (англ. Generative Adversarial Network);

IOS – мобільна операційна система від Apple (англ. iPhone OS);

LSTM – довга короткострокова пам'ять (англ. Long Short-term Memory);

OCR – оптичне розпізнавання символів (англ. Optical Character Recognition);

PDF – універсальний формат файлів, у якому зберігаються елементи вихідних документів, (англ. Portable Document Format);

RNN – повторна нейронна мережа (англ. Recurrent Neural Network);

TXT – стандартне розширення для текстових файлів.

## ВСТУП

Сьогодні сучасні технології все більше впроваджуються у наше життя. З кожним днем автоматизуються все більше галузей людської діяльності: навчання, дослідження, управління, проектування, виробництво. Тенденції розвитку сучасних технологій, мають неабиякий вплив на повсякденне життя людей.

Впровадження пристроїв з сучасними технологіями у життя людей стає все більш поширеним явищем. Сучасні технології не є чимось незвичним, а навпаки простими і зрозумілими.

Яскравим прикладом результату розвитку сучасних технологій є штучний інтелект. Штучний інтелект був розроблений з метою полегшення праці, часткової або ж повної заміни людини у трудомісткій роботі. Використання штучного інтелекту має значні переваги в плані ефективності, точності та конкурентоспроможності. Однак, варто враховувати обмеження та ризики, планувати заходи щодо запобігання потенційних ризиків.

Штучний інтелект представляє собою розумові функції, які зазвичай пов'язані з людським інтелектом, але вбудовані в комп'ютерні системи або програми. Одним з завдань, які може вирішити штучний інтелект є розпізнавання тексту з зображень. За допомогою цієї технології, комп'ютер або ж телефон зможе аналізувати та інтерпретувати текст, який подається у вигляді зображення. Розпізнавання тексту з зображень можна використовувати у багатьох сферах, таких як реклама, освіта, оцифрування документів, юридична галузь, безпека, фінанси, медицина тощо.

Зазвичай людина групує об'єкти чи явища на основі їхніх спільних характеристик або властивостей. За деякими характеристиками людина заносить в певну групу подібні предмети, які можуть все ж значно відрізнятися один від одного. Органи чуттів і досвід відіграють важливу роль у класифікації, оскільки людина використовує свої знання та раніше

отриманий досвід для розрізнення об'єктів і призначення їм певних категорій. На основі характеристик, що є у її досвіді, виділяє кожен символ окремо, а вже потім об'єднує в слова та згодом в тексти.

Штучний інтелект, розпізнає текст так: спочатку кожен символ переводиться в код, щоб потім на основі бази даних класифікувати його. Але потрібно звісно спочатку занести в базу даних інформацію для класифікації, щоб штучний інтелект міг проводити аналіз.

Звісно штучний інтелект при розпізнаванні помилятиметься більше ніж людина, бо має лише обмежену частину інформації, на основі якої й проводить аналіз. Штучний інтелект може зробити не визначити весь текст зображення, особливо при наявності складних чи нестандартних шрифтів, або ж в умовах обмеженої якості зображення. Проте із розвитком сучасних технологій та вдосконаленням алгоритмів, точність систем розпізнавання тексту постійно покращується.

В кваліфікаційній роботі пропонується розглянути актуальну тему розпізнавання тексту на зображеннях. Нейромережеві підходи обробки зображень та розпізнавання тексту в них в останні роки зазнають значного розвитку, а дослідження в цьому напрямку мають широкі практичні застосування.

# 1 АНАЛІЗ ЗАВДАННЯ

## 1.1 Суть технічної проблеми

Завдання розпізнавання образів супроводжує людину щодня з самого народження. З органів відчуттів надходить інформація. Очі збирають світлові подразники і перетворюють їх на нервові сигнали. Вуха отримують звукові хвилі, перетворюють їх на електричні сигнали. Потім мозок обробляє отриману інформацію, приймає рішення, потім передає сигнал, наприклад, органам руху, для реалізації дій. Цей неперервний процес є основою функціонування організму, дозволяє адаптуватися до змін в середовищі та виконувати необхідні завдання.

За допомогою штучного інтелекту та інформаційних технологій, завдання розпізнавання образів буде реалізовано ефективніше та швидше, чим зробіть життя людей простішим.

Розпізнавання образів є важливою галуззю штучного інтелекту, яка зосереджена на розробці та застосуванні методів класифікації об'єктів на основі їхніх властивостей чи характеристик. Об'єкт, який піддається класифікації називають образом. Образ може бути представлений зображенням, символом або голосовим записом мови.

Розпізнавання образів можна використовувати в багатьох сферах, а саме робототехніка, медицина, безпека, автомобільна промисловість. За допомогою розпізнавання образів можна вирішувати такі завдання: розпізнавання зображень, класифікація зображень за характеристиками, розпізнавання мовлення, розпізнавання написаного тексту або текстових символів на зображеннях. Це широкий термін, який застосовується до різних задач аналізу даних, включаючи зображення.

Розпізнавання образів є широким терміном, що охоплює багато завдань аналізу даних, але особливо популярним є обробка зображень.

Розпізнавання зображень є підтипом розпізнавання образів, який

спеціалізується на виявленні та класифікації об'єктів на зображеннях. Завдання розпізнавання зображень полягає у визначенні, які об'єкти або класи присутні на зображенні або в певній області зображення. Головна мета полягає в розробці алгоритмів та моделей, які дозволяють комп'ютерам розпізнавати об'єкти на зображеннях. За допомогою розпізнавання зображень, можна аналізувати та класифікувати велику кількість візуальної інформації, що дозволяє комп'ютерам або телефонам виконувати швидко та ефективно завдання, які раніше були можливо виконати лише людині.

Розпізнавання образів і розпізнавання зображень є двома пов'язаними, але різними завданнями в області комп'ютерного зору та штучного інтелекту.

## 1.2 Поняття, концепція і можливості штучного інтелекту

Штучний інтелект (ШІ, англ. Artificial intelligence) – галузь науки та технологій, що займається розробкою інтелектуальних машин, здатних виконувати завдання, які зазвичай потребують людського інтелекту. Штучний інтелект зосереджений на розумінні, розпізнаванні, вирішенні проблем, самоосвіті та прийнятті рішень, подібних до тих, які здатні виконувати люди. Штучний інтелект досліджує методи розв'язання таких завдань, які вимагають людського розуміння, що передбачає розвиток способів їх розв'язання з використанням методів аналогії, дедукції та індукції, накопичення базових знань і вміння їх використовувати [1].

Штучний інтелект досліджує методи розв'язання завдань, для яких не існує інших способів розв'язання або розв'язання іншими засобами, оскільки вони надто складні. Також штучний інтелект являє собою системи, які можуть навчатися та спроможні замінити в майбутньому інтелектуальними системами людей.

Отже, узагальнено під поняттям штучний інтелект можна розуміти розділ комп'ютерної лінгвістики та інформатики, котрий вивчає найефективніші алгоритми самостійного пошуку та прийняття рішень

шляхом формалізації проблем та завдань, подібних до тих, які виконує людина та тих, які вона не спроможна виконати [2].

Класифікація алгоритмів систем штучного інтелекту:

- простий ШІ включає базові алгоритми та системи, які здатні виконувати конкретні завдання або розв'язувати проблеми, вони можуть мати обмежені можливості та не взаємодіяти з іншими системами, або отримувати обмежені дані;
- складний ШІ представляє більш розвинені системи, які можуть вирішувати більш широкий спектр завдань та взаємодіяти з різноманітними даними, вони можуть використовувати техніки машинного навчання, нейронні мережі та інші методи для вирішення складних проблем;
- суперінтелект відноситься до систем, які перевершують інтелектуальні здібності навіть людей, суперінтелект може мати абсолютну кмітливість та можливості розв'язувати будь-яке завдання, навіть які не під силу людині [3].

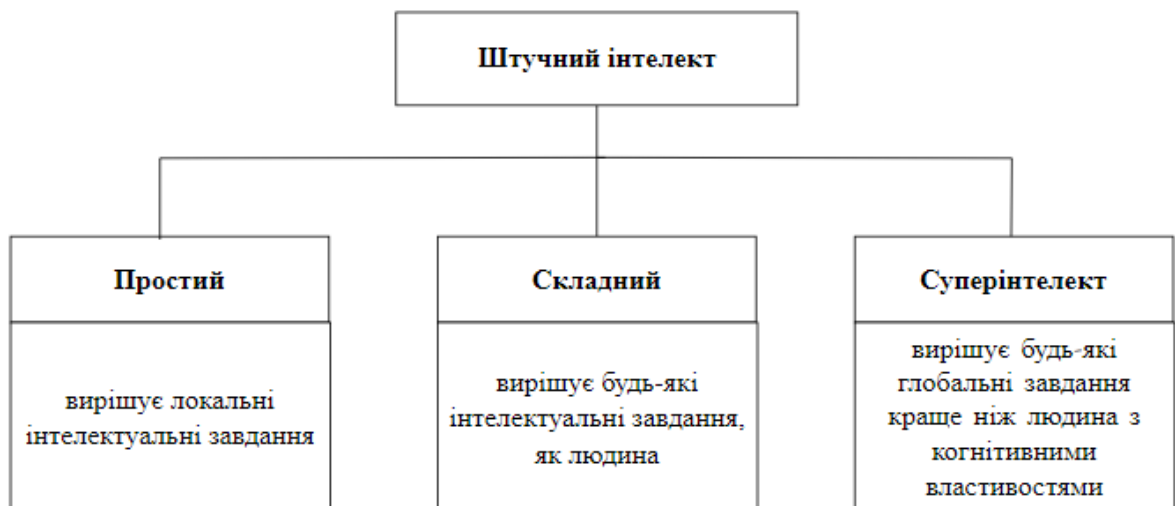


Рисунок 1.1 – Класифікація алгоритмів систем штучного інтелекту

Основні складові штучного інтелекту:

- data science (з англ. – наука про дані, наука аналізу) – процес аналізу інформації для прийняття рішень, що включає в себе методи збору та обробки інформації, оцінку ризиків, моделювання та прогнозування за допомогою інформаційних і телекомунікаційних технологій, методи та підходи до роботи з великими обсягами даних, служать основою для тренування та оптимізації інтелектуальних моделей;

- machine learning (з англ. – машинне навчання) – це складова штучного інтелекту, спрямована на використання статистичних методів уможливлення комп'ютером здатності навчатися, покращувати продуктивність під час виконання завдань, розроблені алгоритми, дозволяють системам самостійно навчатися на основі даних;

- deep learning (з англ. – глибинне навчання) ґрунтується на навчанні ознак даних з метою заміни ручної роботи алгоритмами автоматичного або напівавтоматичного навчання, використовує глибокі нейронні мережі для вирішення завдань розпізнавання образів та узагальнення інформації;

- computer vision (з англ. – комп'ютерний зір) – підгалузь штучного інтелекту, яка охоплює теорію, методи та технології створення комп'ютерних машин, які можуть проводити виявлення та стеження за об'єктами.

Складові штучного інтелекту взаємодіють між собою, щоб створювати інтелектуальні системи, які зможуть ефективно аналізувати, розуміти та взаємодіяти з різними типами інформації. Вони становлять основу для розвитку штучного інтелекту та його застосувань у різних галузях [4].

Основні складові штучного інтелекту та приклади їх застосування зображені на рисунку 1.2.

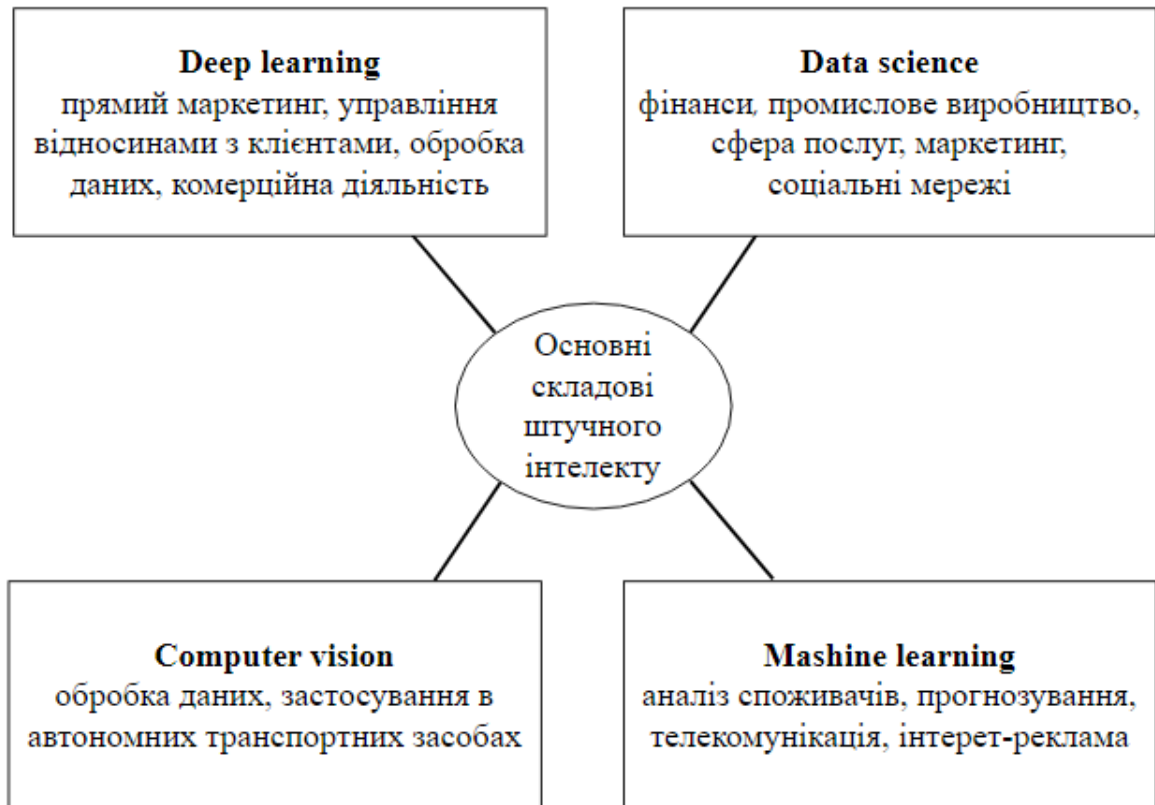


Рисунок 1.2 – Основні складові штучного інтелекту та приклади застосування

Також штучний інтелект може мати різний рівень інтелектуальної здатності, від простих систем, які виконують конкретні завдання, до загального штучного інтелекту, який може вирішувати різноманітні ресурсозатратні завдання.

Штучний інтелект є концепцією, яка виникла з бажання створити комп'ютерні системи і програми, здатні виконувати завдання, які зазвичай потребують інтелектуальних здібностей людини. Ця концепція базується на ідеї, що можна розробити системи, які здатні сприймати оточуючий світ, аналізувати інформацію, навчатися, вирішувати проблеми, розуміти мову та взаємодіяти з людьми.

Концепція штучного інтелекту також включає в себе ідею розробки сильного штучного інтелекту, який був би рівноцінним інтелекту людини.

Такий рівень ШІ мав би здатність вирішувати широкий спектр завдань, володіти загальним розумінням та самосвідомістю.

Типи штучного інтелекту

Розрізняють 4 основні типи штучного інтелекту:

- реактивні машини;
- обмежена пам'ять;
- теорія розуму;
- самосвідомість.

Реактивні машини.

Реактивні машини – це перший тип штучного інтелекту із мінімальними технологічними можливостями. Вони здатні лише імітувати людський мозок та можливість реагувати на деякі види симуляцій.

Такі системи не мають вбудованої функції пам'яті, тому не можуть використовувати попередній досвід чи вивчати нову інформацію для виконання дій. Їх використовують лише у випадках, коли потрібно швидко реагування на стандартні вхідні дані, або ж коли потрібно створити алгоритм дій, що повторюються.

Обмежена пам'ять.

Однією з перших типів штучного інтелекту, яка могла навчатися на власному досвіді, була система штучного інтелекту з обмеженою пам'яттю.

Штучний інтелект з обмеженою пам'яттю використовує попередній досвід для прийняття майбутніх рішень. Цей тип має здатність реагувати на минулі події та навчатися на їхньому прикладі. За допомогою набору навчальних даних, система штучного інтелекту зможе аналізувати те, що відбувається у режимі реального часу.

Теорія розуму.

Теорія штучного інтелекту лише на рівні розуму вказує на розробку системи, яка може моделювати та розуміти людський розум, потреби, логіку та вирішення проблем.

Самосвідомість.

Самосвідомі системи штучного інтелекту вказують на здатність системи до усвідомлення свого власного існування, стану та можливостей. Ця концепція включає в себе вищий рівень свідомості та самосвідомості, схожий на те, що мають люди.

### 1.3 Поняття, концепція і можливості нейронних мереж

Нейронна мережа – це математичні моделі, які імітують роботу людського мозку та нейронної системи для виконання завдань штучного інтелекту, з метою вирішення завдань, таких як розпізнавання, прогнозування та класифікація [6].

Нейронна мережа складається з вхідних шарів, прихованих шарів та вихідних шарів. Дані подаються на вхідний шар, потім інформація обробляється у прихованих шарах, і на вихідних шарах буде отримано результат. Структуру нейронної мережі представлено на рисунку 1.3.

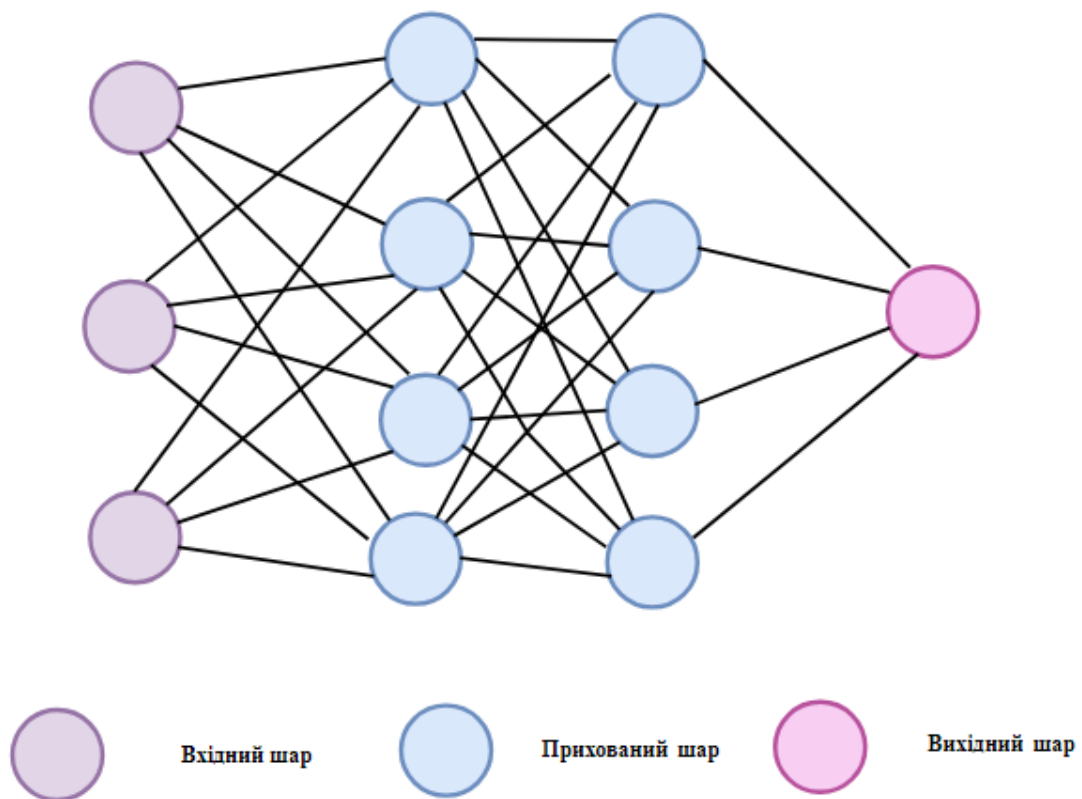


Рисунок 1.3 – Структура нейронної мережі

Структура нейронної була створена за прототипом біологічного нейрону. Система за таким прототипом може розпізнавати, класифікувати, зберігати та відтворювати інформацію для майбутніх дій.

Концепція нейронних мереж ґрунтується на ідеї моделювання обчислювальних процесів, що відбуваються у біологічних нейронах, з метою вирішення завдань штучного інтелекту.

В нейронній мережі обчислювальною одиницею штучний нейрон. Цей нейрон виконує різні функції для обробки та передачі інформації у мережі [7].

### 1.3.1 Біологічний нейрон

Біологічний нейрон є основною структурною та функціональною одиницею нервової системи у біологічних організмах. Він відіграє ключову роль у передачі та обробці інформації в нервовій системі. Основні елементи біологічного нейрона включають:

Майже всі нейрони виконують основні функції необхідні для нормального функціонування всіх клітин організму:

- отримують сигнали від інших нейронів через синапси;
- при отриманні вхідних сигналів нейрони обробляють їх у тілі клітини;
- внутрішні процеси визначають, що потрібно передати сигнал, аксон нейрона відправляє його через синапси іншим клітинам.

Біологічний нейрон може мати різну форму та розмір залежно від його функції та розташування. Усі нейрони складаються з:

- дендрити отримують повідомлення від інших нейронів і дозволяють передавати повідомлення до тіла клітини;
- сома – представляє собою центральну частину клітини, де знаходиться ядро і багато інших клітинних органел, сома обробляє вхідні

сигнали та виробляє електричні імпульси, які потім передаються вздовж аксону для подальшої передачі інформації в нервовій системі;

- аксон переносить електричний імпульс від тіла клітини до терміналів аксона, які передають імпульс до іншого нейрона;
- синапс – це з'єднання між закінченням одного нейрона та дендритами іншого нейрона [8].

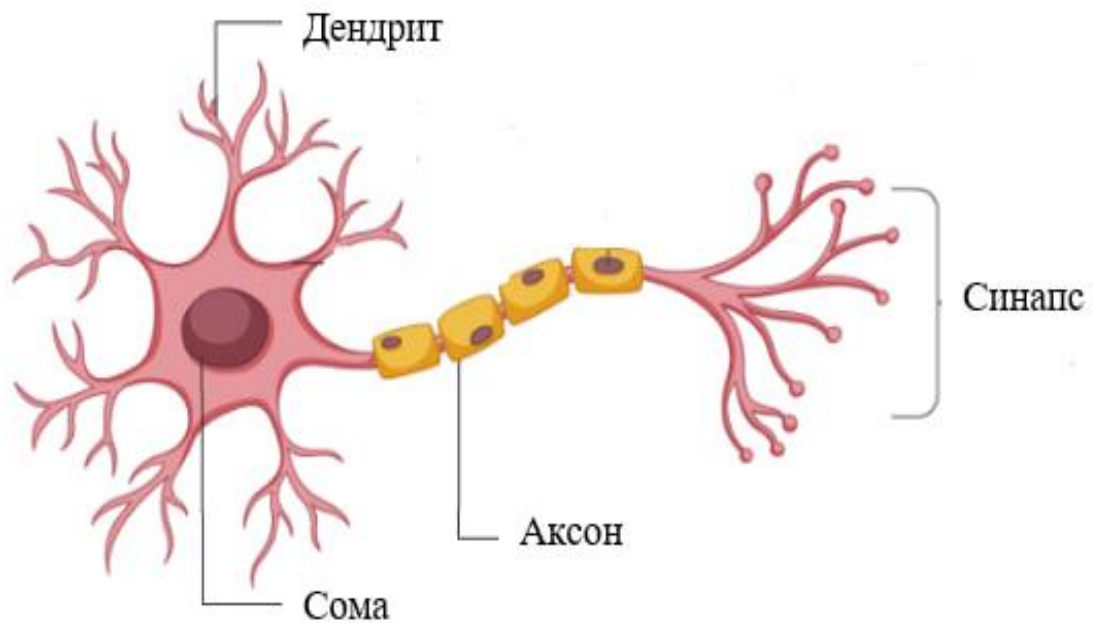


Рисунок 1.4 – Структура біологічного нейрону

Обробка отриманої вхідної інформації відбувається в тілі клітини, а виконується дендритами. Вхідні сигнали можуть бути:

- збудливими – вони стимулюють нейрон для генерації електричного імпульсу;
- гальмівними – вони не дають нейрону генерувати імпульс.

Кожен нейрон має дендритне дерево, де може отримувати значну кількість вхідних сигналів.

### 1.3.2 Штучний нейрон

Основною одиницею нейронної мережі є штучний нейрон. Штучний нейрон називають перцептроном. Перцептрон – це математична функція, яка заснована на моделі біологічних нейронів.

Перцептрон виконує такі основні функції:

- отримує вхідні сигнали від інших нейронів;
- кожен вхідний сигнал множиться на їх відповідні ваги, потім зважені значення додаються для отримання зваженої суми;
- щоб визначити поріг активації, до зваженої суми можуть додавати біас;
- передає суму через активаційну функцію для отримання результату.

Перцептрон складається з 3 шарів рис 1.5:

- S-елементи (сенсорні елементи);
- А-елементи (асоціативні елементи);
- R-елемент (реагуючі елементи).

Спочатку сигнали потрапляють на S-елементи. Вхідними даними є значення 0, якщо сигнал у стані спокою або значення 1, якщо сигнал у збудженому стані. Потім сигнали передаються А-елементам за допомогою S-А зв'язку. А-елементи приймають сигнал та аналізують його. Якщо сума вхідних сигналів не перевищує поріг, то буде значення 0, якщо ж сума перевищує поріг, то значення буде 1. Потім сигнали передаються R-елементам за допомогою А-R зв'язку. R-елементи приймають сигнал і на основі нього буде значення, яке буде подано на вихід. Якщо сума переданих сигналів не перевищує поріг, то буде значення 0, якщо ж сума перевищує поріг, то значення буде 1 на виході.

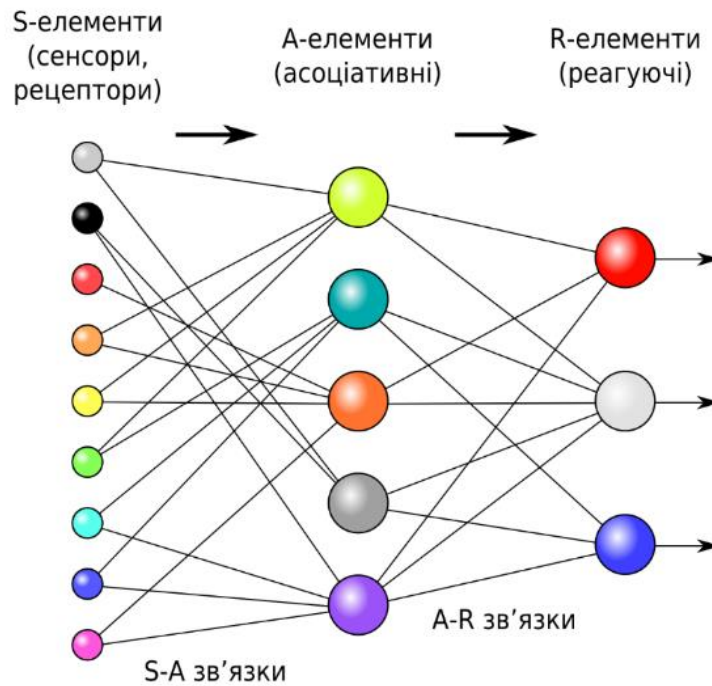


Рисунок 1.5 – Структура перцептрона

Існує 5 класів перцептронів:

- перцептрони, обмежені за діаметром – цей клас має обмеження на розмір фігур, які може розпізнати, величина розміру не перевищує за діаметром деяку фіксовану величину;
- перцептрони обмеженого порядку – цей клас перцептронів вказує на те, що кожна умова залежить від обмеженої кількості точок в просторі;
- перцептрони Гамба – у цьому класі перцептронів усі умови повинні бути лінійними пороговими функціями;
- випадкові перцептрони – у цьому класі умова повинна бути випадково вибраною булевою функцією;
- обмежені перцептрони – для цього класу перцептронів, характерно те, що множина умов є нескінченною, тому такі перцептрони можуть використовувати широкий спектр умов для визначення свого вихідного сигналу на основі вхідних даних [9].

### 1.3.3 Види нейронних мереж

#### 1. Дрібні нейронні мережі (спільна фільтрація).

Дрібні нейронні мережі мають лише вхідні, приховані та вихідні шари. Дрібні нейронні мережі мають малу кількість параметрів, що робить їх менш вимогливими до обчислювальних ресурсів та більш придатними для використання в обмежених середовищах. Дрібні нейронні мережі навчаються швидко, оскільки їм потрібно менше часу для обчислень та тренування. Спільна фільтрація є одним із поширених прикладів дрібних нейронних мереж. В прихованому шарі перцептрона, відбувається навчання представляти схожість між сутностями з метою формування рекомендацій.

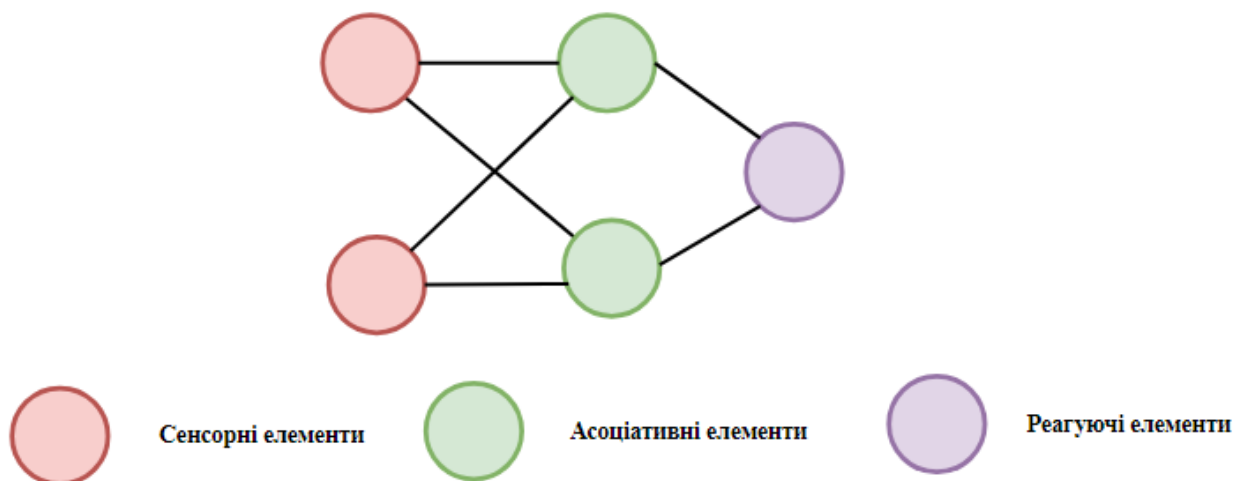


Рисунок 1.6 – Дрібні нейронні мережі

#### 2. Багатошаровий перцептор (глибокі нейронні мережі).

Глибокі нейронні мережі мають більше ніж один прихований шар. За допомогою великої кількості прихованих шарів, глибока нейронна мережа може вирішити будь-яке завдання. Глибина мережі визначається кількістю прихованих шарів. Чим більше прихованих шарів, тим більше глибина мережі.

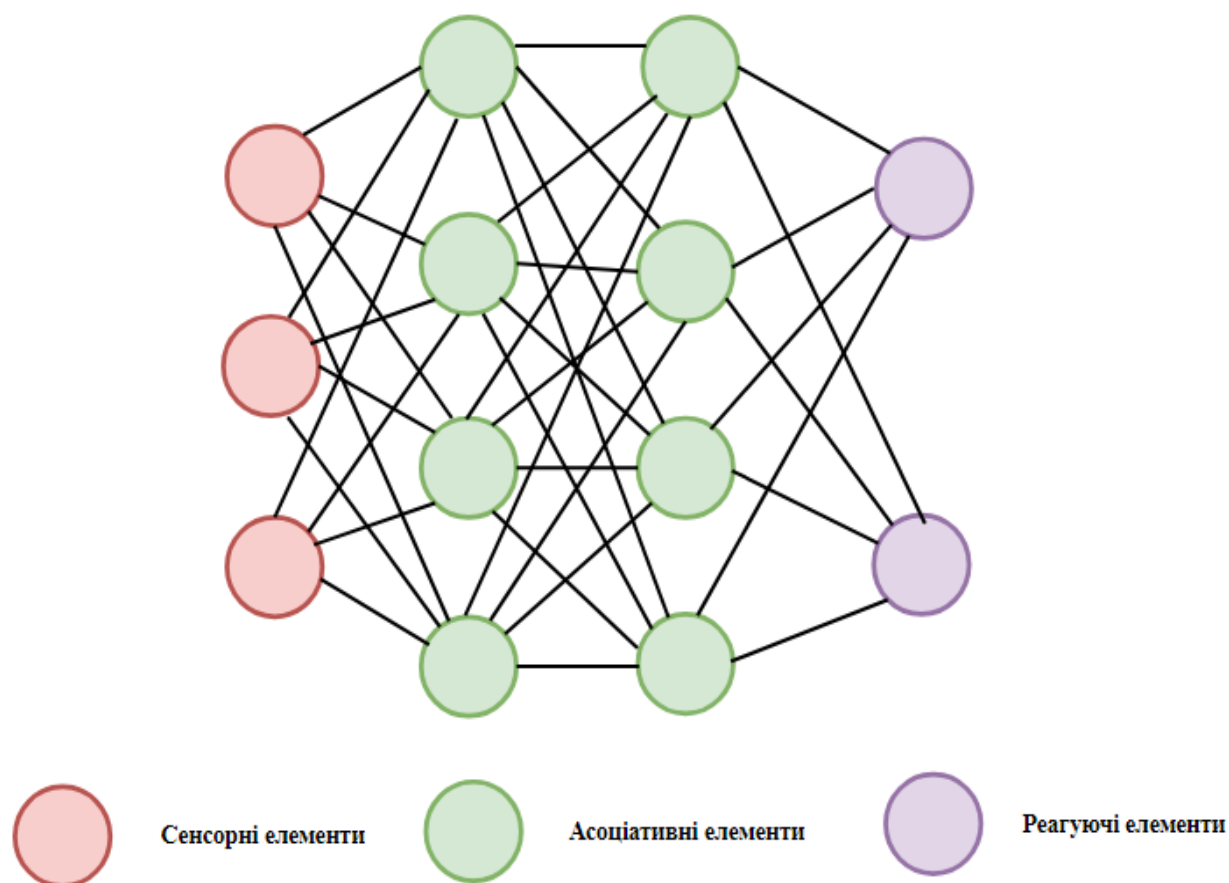


Рисунок 1.7 – Багатошаровий перцептор

### 3. Конволюційна нейронна мережа (CNN).

Конволюційна нейронна мережа була спеціально розроблена для обробки структурованих сіткових даних, таких як зображення. Конволюційна нейронна мережа у наборі даних сканує кожен піксель окремо. Конволюційні шари в мережі використовують для виявлення особливостей вхідних даних, таких як розмір (помічає контури фігур) та форму.

Конволюційна нейронна мережа за рахунок застосування згортки зменшує кількість інформації, що зберігається в пам'яті інформації та може краще справляється з класифікацією зображень [10].

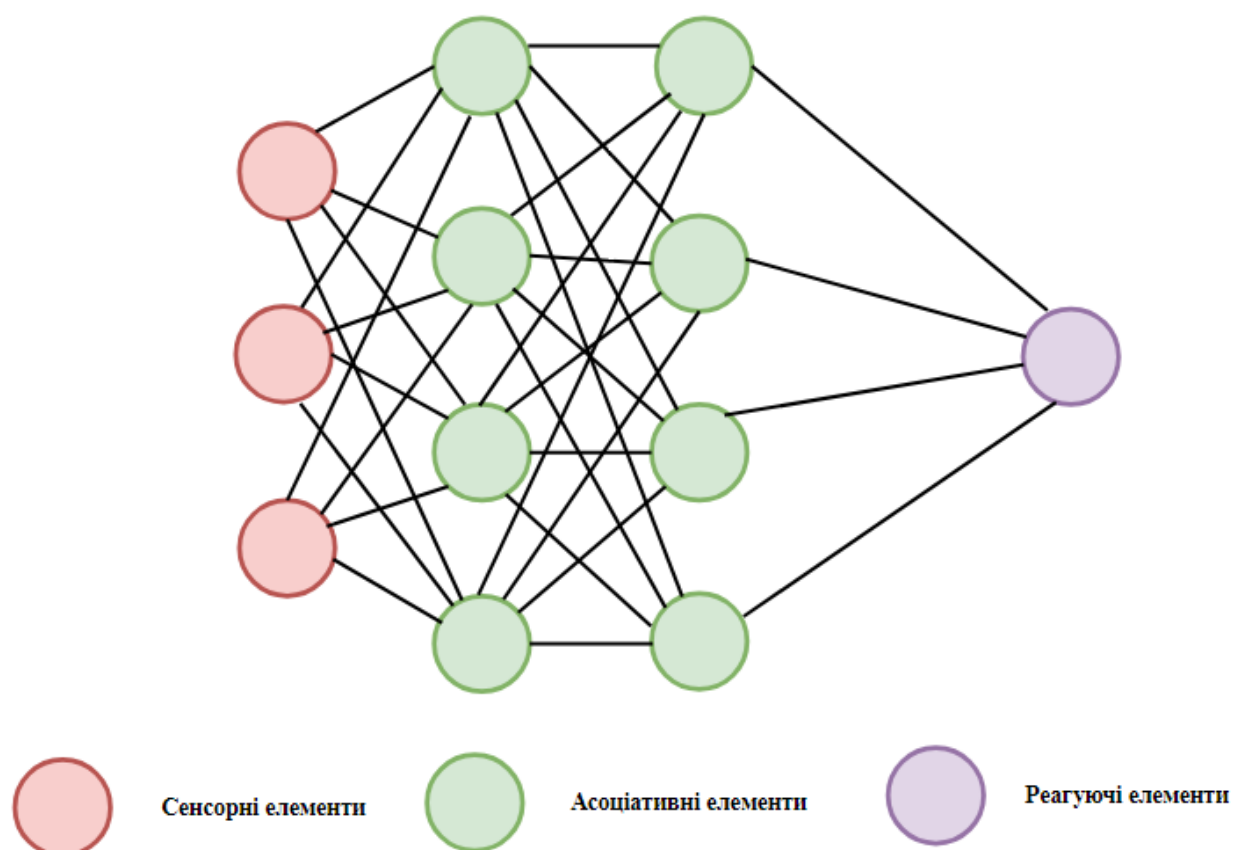


Рисунок 1.8 – Конволюційна нейронна мережа

#### 4. Повторна нейронна мережа (RNN).

Повторні нейронні мережі призначені для роботи з послідовними даними, такими як аудіо чи текст. На кожному кроці мережа приймає значення на вході та попередньому виході, потім обчислює новий вихід і оновлює значення внутрішнього стану. Повторна нейронна мережа зберігає попередні входні дані та використовує їх інформацію для обробки даних.

Повторні нейронні мережі подають на вихід декількох прихованих шарів назад до входного шару для агрегації та передачі наближення до наступної ітерації входного набору даних. Це також допомагає моделі самостійно вчитися та коригувати прогнози.

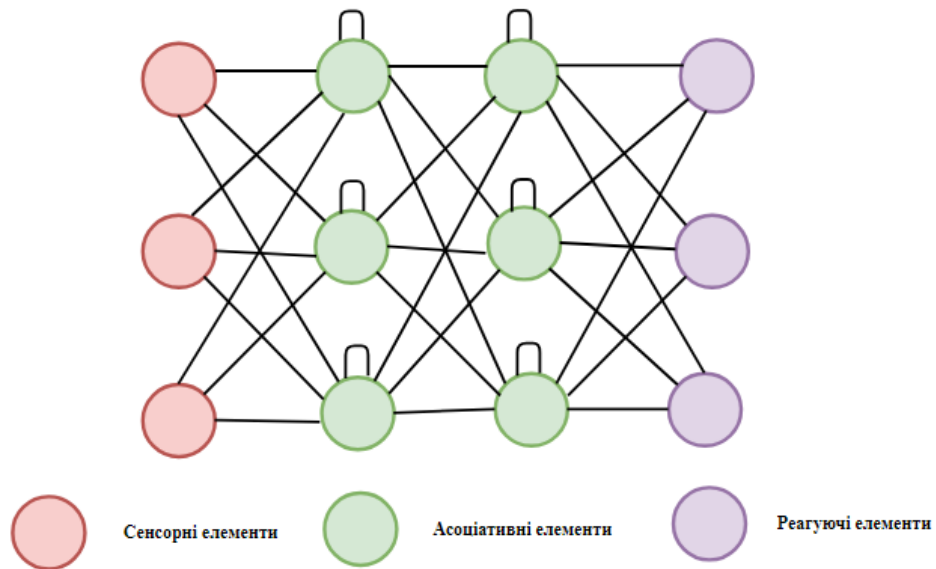


Рисунок 1.9 – Повторна нейронна мережа

### 5. Довга короткострокова пам'ять (LSTM).

Довга короткострокова пам'ять здатна управляти та зберігати довгострокові залежності, що дозволяє ефективно вирішувати завдання, де важлива структура послідовності. В прихованому шарі мережа визначає, яку інформацію потрібно додати до клітинного стану, а яку забути [11].

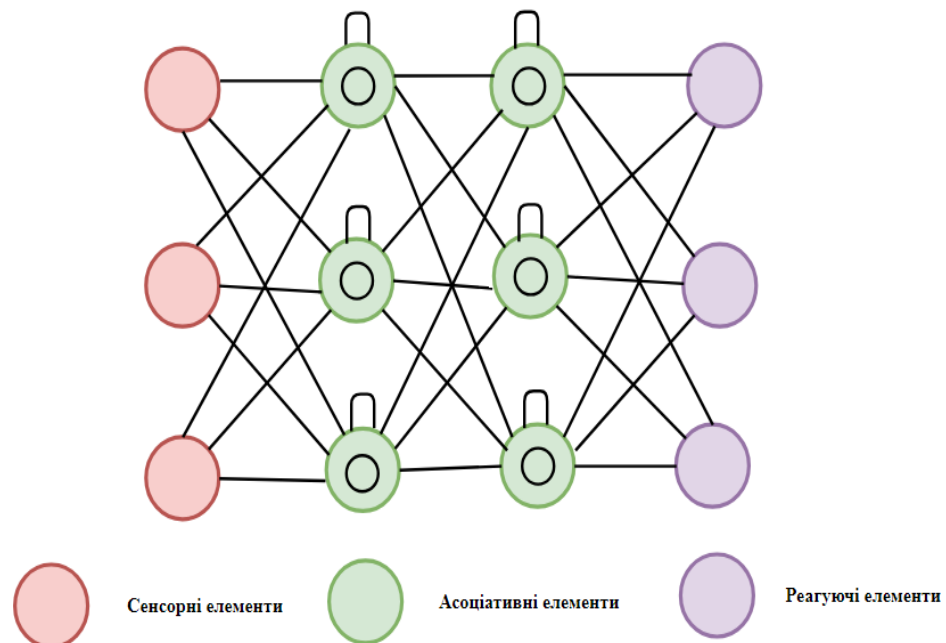


Рисунок 1.10 – Довга короткострокова пам'ять

## 6. Генеральна змагальна мережа (GAN).

Генеральна змагальна мережа складається з двох компонентів: генератора і дискримінатора. Головна мета генератора на основі навчальних даних створити нові приклади. Дискримінатор визначає, які дані є навчальними, а які згенеровані генератором. Дискримінатор отримує вхідні дані та намагається розрізнити між ними. В генеральній змагальній мережі генератор та дискримінатор змагаються один з одним. Генератор намагається створити дані реалістичними, а дискримінатор намагається відрізнити справжні та створені зразки. Змагальний процес буде тривати, коли генератор не зможе створювати реалістичні зразки, або зразки не зможе відрізнити дискримінатор [12].

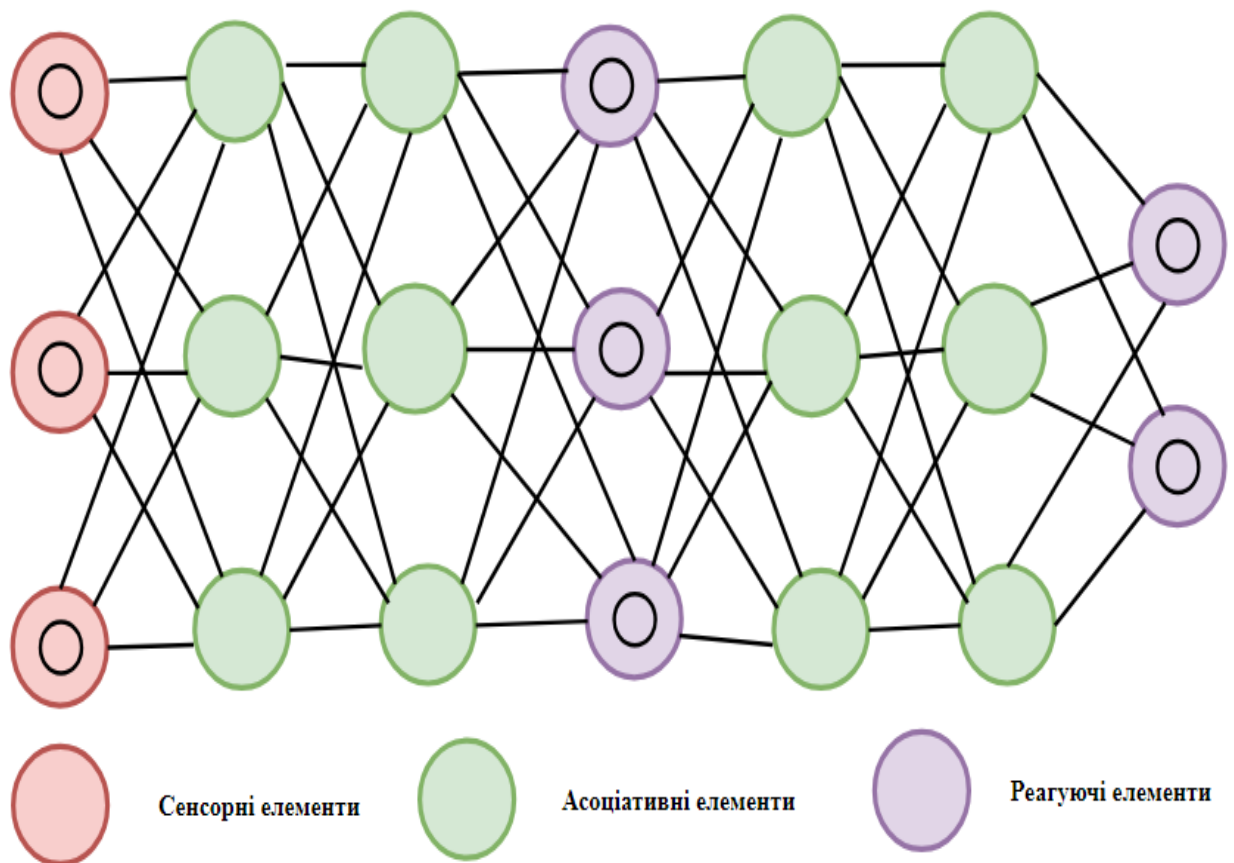


Рисунок 1.11 – Генеральна змагальна мережа

### 1.3.4 Навчання нейронних мереж

Однією з основних особливостей інформаційних технологій є здатність проводити узагальнення та класифікацію, ґрунтуючись на вже наявних і нових даних, або іншими словами – їх здатність до навчання.

Навчання – це процес, під час якого за допомогою моделювання середовища параметри нейронної мережі налаштовуються згідно мережі. Сьогодні не існує універсального алгоритму навчання, який підходив би для всіх нейронних мереж, тому згідно параметрів обирається тип навчання.

Існують три підходи навчання нейронних мереж:

- навчання з вчителем (supervised learning);
- навчання без вчителя (unsupervised);
- навчання з підкріпленням (reinforcement learning).

При підході навчання з учителем, нейронна мережа має значення вхідних та вихідних значень, і згідно них буде відбуватися навчання. Для навчання потрібно ще додати бажані результати. Цей підхід використовують, коли потрібно класифікувати значення на основі вхідних даних.

При підході навчання без учителя, нейронна мережа має значення вхідних, але не має вихідних значень, згідно яких буде відбуватися навчання. Тому нейронна мережа самостійно в процесі навчання має визначити значення на виході. Навчання без вчителя застосовується для візуалізації даних та виявлення помилок, де необхідно передбачити вихідні значення на основі вхідних даних.

Навчання з підкріпленням є проміжним підходом між навчанням з вчителем і навчанням без вчителя. При застосуванні цього підходу є можливість оцінювати коректність роботи мережі та зазначати бажані результати на виході, це і називають підкріпленням.

Кожен тип навчання має особливості застосування, а вибір типу залежить від завдання. Для вирішення складних завдань можна комбінувати підходи, тим самим покращувати ефективність виконання завдань [13].

## 1.4 Загальний алгоритм розпізнавання зображень

Загальний алгоритм розпізнавання зображень складається з декількох основних етапів.

Етапи:

1. завантаження зображення;
2. підготовка зображення;
3. класифікація;
4. обробка результатів;
5. виведення результатів.

Загальний алгоритм розпізнавання зображень зображений на рис.1.12.

1. Завантаження зображення.

Зображення може бути завантажене з пам'яті телефону або за допомогою камери. При отриманні зображення, система приймає його як вхідні дані для подальшої обробки та аналізу.

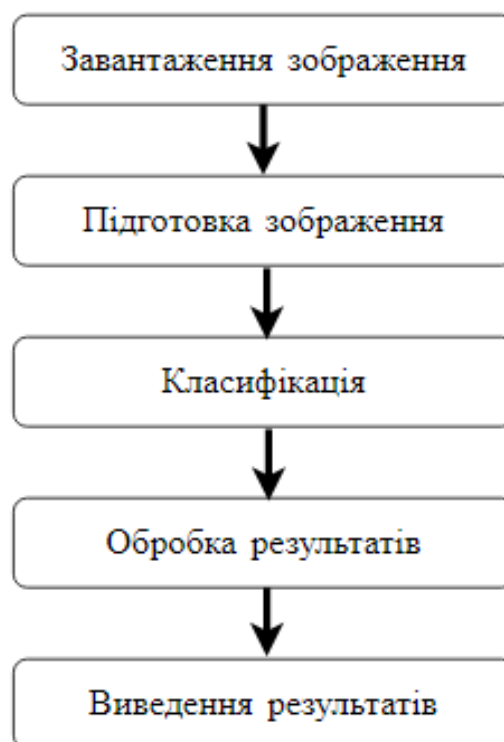


Рисунок 1.12 – Загальний алгоритм розпізнавання зображень

## 2. Підготовка зображення.

Підготовка зображення складається з певних операцій, що виконуються із зображенням. Послідовність операцій, які виконуються над зображенням показані на рис.1.13.

Першим етапом за необхідності потрібно змінити розмір зображення. Потім потрібно застосувати операції по регулюванню кольору, яскравості та інших параметрів для досягнення оптимального візуального вигляду. Далі для поліпшення якості потрібно зменшити шум на зображенні. Наступним етапом є перетворення зображення в двійковий формат. На етапі сегментації зображення поділяють на сегменти. Розпізнавання кожного сегменту відбувається окремо. Далі відбувається виділення основних ознак та зберігання їх. Останнім етапом є збереження обробленого зображення для подальшого використання.

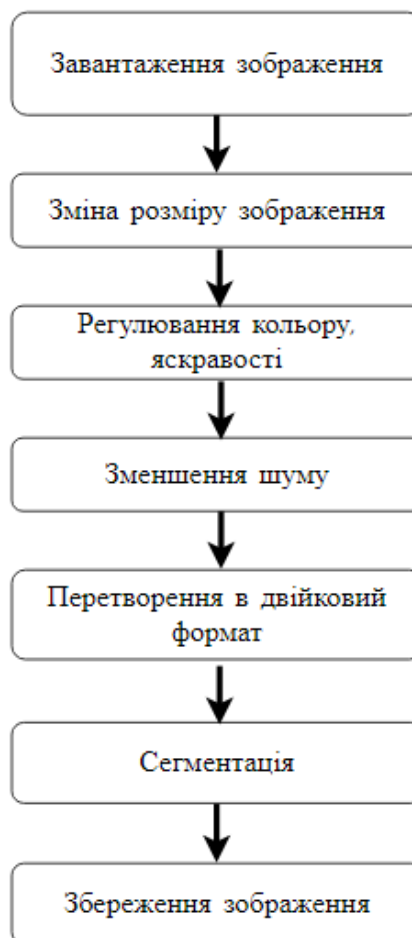


Рисунок 1.13 – Алгоритм підготовки зображення

### 3. Класифікація.

На цьому етапі відбувається класифікація об'єктів на основі отриманих признаков. На вхід подається сегментоване зображення, що було отримане на минулому етапі.

### 4. Обробка результатів.

На цьому етапі система об'єднує результати та виправляє можливі помилки.

### 5. Виведення результатів.

Завершальним етапом є виведення результатів. На цьому етапі система розпізнані символи виводить файл, у якому представлено результат у текстовому вигляді. За необхідності результат можна зберегти для подальшого використання.

## 2 ПОСТАНОВКА ЗАДАЧІ ТА АНАЛІЗ ОБРАНИХ ЗАСОБІВ РОЗРОБКИ

### 2.1 Постанова задачі

Об'єкт дослідження у роботі – розпізнавання тексту на зображеннях з використанням нейронних мереж.

Предмет дослідження – методи нейромережевого розпізнавання тексту на зображеннях.

Мета роботи – розробка мобільного застосунку для розпізнавання тексту на зображеннях.

Задачі, які потрібно вирішити для поставленої мети:

- аналіз методів розпізнавання тексту з зображень;
- аналіз мобільних операційних систем;
- розробка повноцінного застосунку, для розпізнавання тексту з зображень.

Основними вимогами до застосунку є:

- система повинна розпізнавати текст, що присутній у зображенні;
- можливість взаємодії з результатами, а саме копіювання тексту;
- система повинна показувати повідомлення про помилку користувачеві, коли зображення не в потрібному форматі;
- система повинна забезпечувати якість розпізнавання та зберігання тексту.

Основним функціоналом застосунку є:

- завантаження зображення для розпізнавання за допомогою камери або з пам'яті телефону;
- отриманий результат можна закріпити на головному екрані або видалити;
- результат тестування у вигляді тексту, що можна скопіювати та зберегти.

## 2.2 Методи для вирішення задачі

### 2.2.1 Розпізнавання за шаблоном

OCR (англ. Optical Character Recognition) – це використання технології для автоматичного визначення та розпізнавання тексту на зображеннях. Основний процес OCR передбачає розпізнавання тексту документа та переклад символів у код, який потім буде оброблено.

У технології OCR поєднано апаратне та програмне забезпечення. Апаратним обладнанням, яке буде як сканер, використовується мобільний телефон, для завантаження зображень. Програмне забезпечення написано на базі Android з використанням штучного інтелекту для розпізнавання тексту з зображень. Після отримання тексту з зображень, користувачі можуть копіювати та зберігати його.

Першим етапом технології OCR є використання камери телефону для завантаження зображення. Потім завантажене зображення аналізується за загальним алгоритмом розпізнавання зображень. Далі на основі програмного забезпечення текст розпізнається з зображення та виводиться текстом користувачу. Отриманий текст користувач можна скопіювати та зберегти на телефон.

Програмне забезпечення розпізнає в тексті кожен символ окремо. Розпізнавання засновано на використанні ознак та характеристик символів, що були завчасно завантажені в систему, щоб на основі них і було розпізнавання [14].

### 2.2.2 Розпізнавання за допомогою метрик

Метрика – це деяке умовне значення функції, що визначає відстань між об'єктами в просторі. Метрика відображає ступінь схожості між об'єктами, об'єкти, які розташовані близько один до одного, матимуть відносно невеликі значення відстані. А якщо об'єкти розташовані далеко один від одного, то значення метрики буде великим, що вказує на велику відмінність між ними.

Розпізнавання за допомогою метрик полягає у порівнянні метрики з усіма шаблонами в системі. І метрика якого шаблону буде найменша і буде відповіддю. Оскільки деякі літери схожі одна на одну, тому для кращого результату, потрібно робити декілька розрахунків метрик [15].

### 2.2.3 Розпізнавання за допомогою нейронних мереж

Нейронна мережа – це математична модель, яка намагається відтворити функціональні аспекти роботи нервової системи людини. Для початку мережу потрібно навчити, внести значення для кожної літери окремо. В мережі будуть зберігатися навчальні набори, які потім будуть використані.

Кількість входів дорівнює кількості пікселів зображення. Кожен піксель окремо подається на вхід. Кожен вхід в нейронну мережу враховується з ваговим коефіцієнтом, і під час розпізнавання на кожному нейроні накопичується заряд, заряд є більшим за нейрон, тому видаватиме імпульс.

Також мережі необхідно налаштувати вагові коефіцієнти, що використовуються під час навчання. Навчальний модуль використовують для налаштування цих коефіцієнтів. На основі навчальних наборів відбувається порівняння значень [16].

## 2.3 Програмна частина

Мобільний застосунок, для розпізнавання тексту на зображеннях, було створено в середовищі розробки для мобільних застосунків на Android, платформі Android Studio. В якості мови розробки для Android застосунку обрано Kotlin, те, що Kotlin простий та гнучкий, має можливості, які полегшують розробку, такі як розширення функціонального програмування, удосконалені системи типів тощо.

## 2.4 Огляд існуючих рішень

Сьогодні існує велика різноманітність мобільних застосунків, для розпізнавання тексту на зображеннях. Ці застосунки використовують технології штучного інтелекту та використання нейронних мереж для аналізу зображень і виявлення на них тексту.

Google Keep є популярним мобільним застосунком для збереження нотаток, списків справ і нагадувань. Він також має функцію розпізнавання тексту на зображеннях. За допомогою цієї функції можна фотографувати або завантажувати зображення, і застосунок автоматично розпізнає цей текст і перетворює його на редагований текстовий формат. Це дозволяє легко зберігати та організувати текстову інформацію з фотографій у вигляді нотаток в Google Keep.

Google Keep є доступним для Android та iOS, а також можна використовувати його через веб-інтерфейс на комп'ютерах. Застосунок синхронізує нотатки між усіма пристроями, де є обліковий запис Google [17].



Рисунок 2.1 – Застосунок Google Keep

Microsoft Office Lens є мобільним застосунком, розробленим компанією Microsoft, який надає можливість сканувати документи, розпізнавати текст на зображеннях та зберігати їх у різних форматах. Застосунок Office Lens доступний для пристроїв на базі операційних систем Android і iOS.

За допомогою Microsoft Office Lens можна зробити фотографію документа або іншого зображення з текстом, а застосунок автоматично розпізнає цей текст і перетворює його на редаговану нотатку або текстовий документ. Текст можна зберегти у форматах PDF, Word або PowerPoint, щоб потім поділитися даними.

Microsoft Office Lens також має ряд інших корисних функцій, таких як покращення якості зображень, вирівнювання перспективи, видалення тіней та автоматичне розподілення документів на сторінки.

Microsoft Office Lens працює у синхронізації з іншими продуктами Microsoft Office, такими як OneNote, OneDrive, Word і PowerPoint. Це дозволяє зручно та швидко імпортувати скановані документи в інші програми Microsoft Office для подальшої обробки та редагування [18].

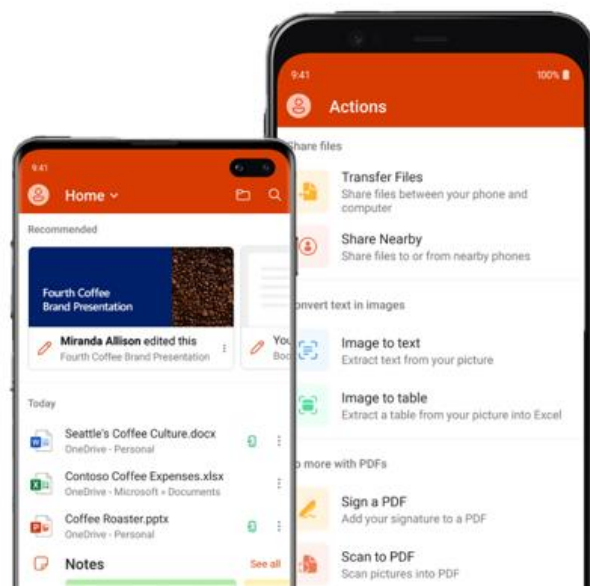


Рисунок 2.2 – Застосунок Microsoft Office Lens

Adobe Scan є мобільним застосунком, розробленим компанією Adobe, який дозволяє сканувати документи та розпізнавати текст на них. Цей застосунок дозволяє використовувати камеру вашого мобільного пристрою для створення високоякісних сканів документів та автоматичного розпізнавання тексту на цих зображеннях.

За допомогою Adobe Scan можна фотографувати будь-який документ, а застосунок розпізнає цей текст і створить редаговану копію документа. Потім можна зберегти копію у форматі PDF або зберегти текст, для подальшого редагування.

Adobe Scan також має додаткові функції, такі як автоматична обрізка та покращення якості зображень, можливість об'єднання сканованих сторінок в один документ, створення множинних сторінок зображень та можливість надіслати скановані документи електронною поштою або іншими способами.

Adobe Scan доступний для пристроїв на базі операційних систем Android і iOS. Застосунок також може інтегруватися з іншими сервісами Adobe, такими як Adobe Acrobat і Adobe Document Cloud, для зручного управління та обробки сканованих документів [19].

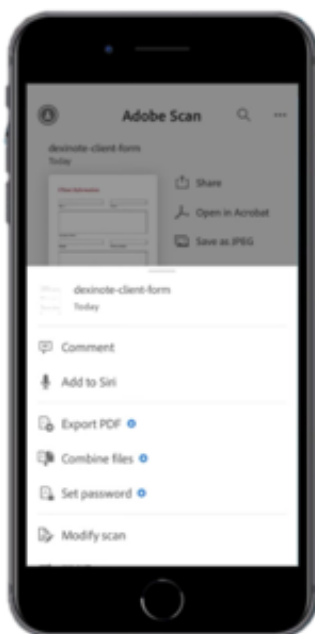


Рисунок 2.3 – Застосунок Adobe Scan

Evernote є мобільним застосунком для збереження нотаток, списків справ та інших типів інформації. Крім цього, Evernote також має функцію розпізнавання тексту на зображеннях.

Evernote дозволяє фотографувати зображення, які містять текст, та автоматично розпізнавати цей текст. Потім розпізнаний текст можна редагувати, додавати до нього додаткову інформацію та зберігати.

Evernote має й інші корисні функції, які допомагають організувати інформацію:

- нотатки та дані автоматично синхронізуються, що дозволяє отримати доступ до них з будь-якого пристрою;
- проста організація, можливість створення розділів, міток та тегів для швидкого пошуку;
- можливість швидко знаходити потрібну інформацію, а також розпізнавати текст на зображеннях.

Evernote доступний для мобільних пристроїв на базі операційних систем Android і iOS, а також має веб-версію, яку можна використовувати на комп'ютерах [20].

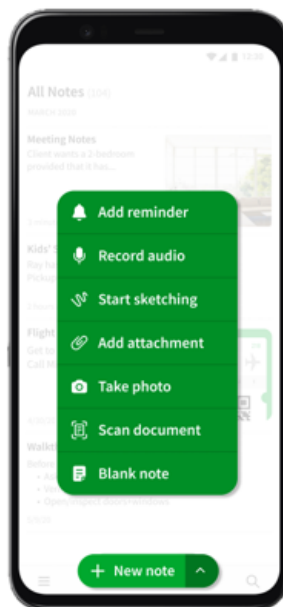


Рисунок 2.4 – Застосунок Evernote

CamScanner є мобільним застосунком, який дозволяє сканувати документи, картки, рахунки та інші зображення з текстом. Він також має функцію розпізнавання тексту на зображеннях.

За допомогою CamScanner можна робити фотографію документа або іншого зображення з текстом, а застосунок автоматично розпізнає текст на ньому. Є можливість редагування розпізнаного тексту, з вибором шрифтів та кольорів, а також зберігання тексту у різних форматах.

CamScanner також має інші корисні функції, такі як автоматичне обрізання та покращення якості зображень, можливість об'єднання сканованих сторінок в один документ, створення паролів для захисту документів, а також можливість надіслати скановані документи по електронній пошті.

CamScanner доступний для пристроїв на базі операційних систем Android і iOS. Застосунок також має платну версію з додатковими функціями, такими як безмежне сховище у хмарі, видалення водяних знаків[21].

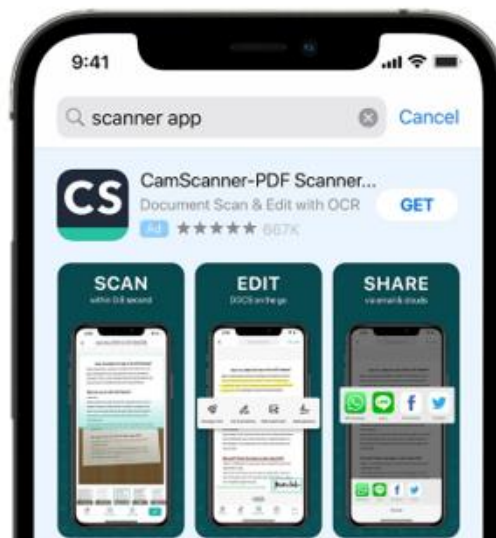


Рисунок 2.5 – Застосунок CamScanner

## 2.5 Аналіз відомих мобільних операційних систем

Сьогодні все частіше інформаційні технології входять в наше життя, тому комп'ютер чи смартфон не є чимось незвичним. Як і в будь-якому комп'ютері, у мобільному пристрою є своя операційна система. Операційні системи для мобільних пристроїв грають ключову роль у їхньому функціонуванні, популярними операційними системами є Android і iOS.

Android, розроблена компанією Google, вийшла на ринок у 2008 році. З тих пір вона здобула велику популярність і наразі є лідером серед мобільних операційних систем.

iOS, розроблена компанією Apple, є другою за популярністю мобільною операційною системою. Вона запущена разом з першим iPhone у 2007 році. Операційна система iOS інтегрована із пристроями від Apple, такими як iPhone та iPad [22].

Переваги ОС Android:

- можливість допрацювати або змінювати код;
- інтеграція з різними сервісами Google;
- просте оновлення через Інтернет;
- широкі можливості налаштувань;
- різноманітність застосунків на Android;
- присутність файлової системи;
- безпроблемна одночасна робота кількох застосунків;
- простота встановлення застосунків із різних ресурсів, навіть створених самостійно.

Недоліки ОС Android:

- не завжди є NFC;
- вразливість до вірусів;
- потрібна реєстрація в Google;
- безпека, через відкритий код;
- короткий час роботи акумулятора через багатозадачність [23].

#### Переваги ОС IOS:

- безпека та надійність;
- висока якість застосунків;
- зручність використання;
- швидке оновлення;
- велика різноманітність застосунків;
- інтеграція з іншими продуктами Apple;
- тривала підтримка старих пристроїв;
- в App Store є сімейний доступ для покупок [24].

#### Недоліки ОС IOS:

- дорога вартість;
- велика кількість платних застосунків;
- обмежена можливість персоналізації інтерфейсу;
- закрита операційна система;
- деякі аксесуари може випускати тільки Apple;
- якщо немає підключення до інтернету, то деякі функції не працюють;
- застосунок створений самостійно встановити дуже важко, можна завантажувати тільки з AppStore [25].

Отже, порівнявши все вище описане, в якості мобільної операційної системи майбутнього застосунку для розпізнавання тексту з зображень було обрано Android.

## 2.6 Аналіз програмних засобів для розробки застосунків для операційної системи Android

Компанія Google випустила Android Studio, для розробки програмного забезпечення для платформи Android. Android Studio — це інтегроване середовище розробки (IDE), призначене спеціально для створення додатків під управлінням Android. Ця платформа має широкий набір інструментів,

включаючи візуальний редактор для інтерфейсу користувача, емулятор для тестування додатків на різних пристроях, розширені можливості по збірці та налагодженню коду.

Основними особливостями Android Studio є:

- рефакторинг коду;
- розширений редактор макетів;
- різноманітність збірок та генерація декількох .apk файлів;
- шаблони простих компонентів та макетів;
- контроль сумісності версій;
- підтримка інтеграції з сервісами Google.

## 2.7 Загальні поняття Android-розробки

Компоненти застосунку Android складаються з блоків. Маніфест застосунку `AndroidManifest.xml` описує компоненти програми та взаємодію компонентів між собою.

Існує чотири компоненти, які використовують в застосунку Android:

- операції (Activities), елементи призначені для взаємодії користувача інтерфейсу та екрану гаджету;
- служби (Services), працює у фоновому режимі, незалежно від того, чи взаємодіє користувач з застосунком чи ні, використовується для виконання довготривалих операцій, таких як завантаження файлів чи оновлення даних в реальному часі;
- приймачі ширококомовних повідомлень (Broadcast receivers), такі компоненти, що моніторять повідомлення ОС та передають системі керування;
- постачальники контенту (Content providers), такі компоненти, що відповідають за взаємодією застосунку з базою даних, реагують на системні або власні події, такі як зміна стану батареї чи отримання повідомлення.

Крім того існують і додаткові компоненти, які потрібні для побудови зв'язку між типовими компонентів та зовнішніми елементами. Додаткові типи компонентів:

- види (Views) – елементи, які відображаються на екрані, такі як кнопка, текстове поле чи зображення;
- ресурси (Resources) – зображення або константи;
- фрагменти (Fragments) – частини операцій;
- макети (Layouts) – відповідають за структуру та розташування елементів інтерфейсу;
- наміри (Intents) – механізм для комунікації між різними компонентами;
- маніфест (Manifest) – конфігураційний файл програми [28].

## 3 РОЗРОБКА ПРОГРАМНОЇ ЧАСТИНИ

### 3.1 Загальна архітектура

Мобільний застосунок матиме декілька екранів, а саме головний екран та екран результату розпізнавання. На головному екрані буде зображено перелік результатів та кнопки для початку нового розпізнавання (за допомогою камери та завантаження з пам'яті телефону). Екран результатів містить заголовок та розпізнаний текст. Також на цьому екрані є кнопки закріплення результату, видалення, копіювання та можливість надіслати. Загальна архітектура застосунку зображена на рис.3.1.

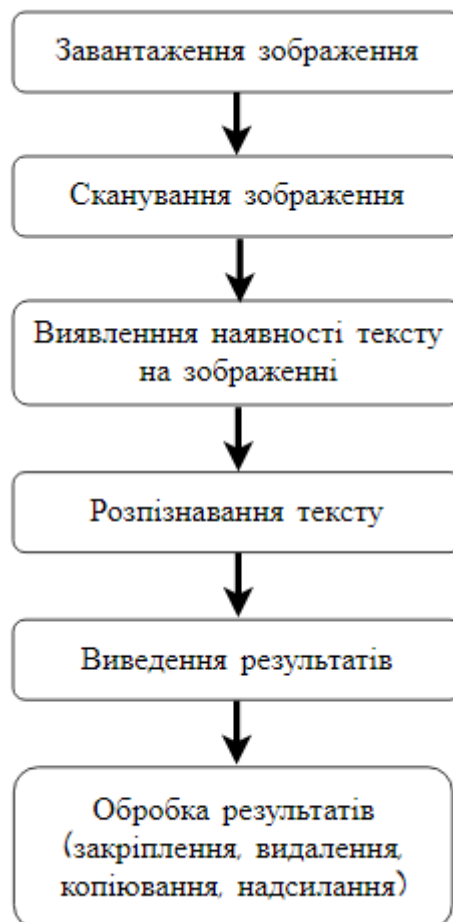


Рисунок 3.1 – Загальна архітектура застосунку

### 3.2 Опис застосунку

Після створення Android Studio було відкрито файл `build.gradle` та додано залежність служб Google Play для `dependencies` блокування:

```
compile'com.google.android.gms:play-services-vision:1.8.0'.
```

Потім потрібно додати залежності у `build.gradle`, залежності описано у лістингу 3.1.

#### Лістинг 3.1 – Фрагмент коду опису залежностей у `build.gradle`

```
dependencies {
    classpath 'com.android.tools.build:gradle:7.2.2'
    classpath"org.jetbrains.kotlin:kotlin-gradle-plugin:
    $kotlin_version"
    def nav_version = "2.5.3"
    classpath "androidx.navigation:navigation-safe-args-gradle-
    plugin:$nav_version"
}
```

Оскільки потрібно буде використовувати камеру телефону для додавання зображень, то потрібно надати дозвіл доступу до камери у файлі `AndroidManifest.xml`, дозвіл описано у лістингу 3.2.

#### Лістинг 3.2 – Фрагмент коду дозволу доступу до камери `AndroidManifest.xml`

```
<uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission
android:name="com.android.vending.BILLING"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher"
        android:supportsRtl="true"
        android:theme="@style/Theme.StudyTextScan"
```

```

        android:name=".MyApp">
        <activity android:name=".MainActivity"
            android:windowSoftInputMode="adjustResize"
            android:screenOrientation="portrait"
            android:exported="true">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>

<intent-filter>
<action android:name="android.intent.action.SEND" />

<category android:name="android.intent.category.DEFAULT" />
    <data android:mimeType="image/*" />

</intent-filter>
</activity>

```

Для попереднього перегляду зображень, знятих на камеру телефону було використано клас `CoordinatorLayout` та клас `FragmentContainerView` для відображення вмісту розпізнаного тексту, класи описано у лістингу 3.3.

### Лістинг 3.3 – Фрагмент коду `Activity_main.xml`

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout

xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:clipToPadding="false"
    android:background="@color/background"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

<androidx.fragment.app.FragmentContainerView
    android:id="@+id/nav_host_frag"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    android:name="androidx.navigation.fragment.NavHostFragment"
        app:defaultNavHost="true"
        app:navGraph="@navigation/nav_graph"/>
</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

Для розпізнавання тексту на зображенні використовуємо OCR та бібліотеку ML Kit, клас описано у лістингу 3.4.

#### Лістинг 3.4 – Фрагмент коду ScanTextFromImageUseCase.kt

```
class ScanTextFromImageUseCase(
    private val dispatcher: CoroutineDispatcher = Dispatchers.IO) {
    private val recognizers = listOf(
        TextRecognition.getClient(TextRecognizerOptions.DEFAULT_OPTIONS,
            TextRecognition.getClient(DevanagariTextRecognizerOptions.Builder()
                .build()),
            TextRecognition.getClient(JapaneseTextRecognizerOptions.Builder()
                .build()),
            TextRecognition.getClient(KoreanTextRecognizerOptions.Builder()
                .build()),
            TextRecognition.getClient(ChineseTextRecognizerOptions.Builder()
                .build()),
        )

    suspend operator fun invoke(image: InputImage) =
        withContext(dispatcher) {
            kotlin.runCatching {
                val detectedText = mutableSetOf<String>()

                recognizers.forEach { textRecognizer ->

                    val result = textRecognizer.process(image).await()
                    if (result.text.isNotBlank()) {
                        detectedText.add(result.text)
                    }

                    val completeText = buildString {
                        detectedText.forEach { text -> append(text) }
                    }
                    completeText}
            }
        }
    }
```

## Функції для відображення розпізнаного тексту описано у лістингу 3.5

## Лістинг 3.5 – Фрагмент коду ScannerTextField.kt

```

@Composable
fun ScannerDateText(
    modifier: Modifier = Modifier,
    text: String) {
    Text(
        modifier = Modifier
            .padding(horizontal = 16.dp, vertical = 2.dp)
            .then(modifier),
        text = text,
        fontSize = 14.sp,
        color = TextColorGray)
}

@Composable
fun ScannerTextField(
    modifier: Modifier = Modifier,
    fontSize: TextUnit = 22.sp,
    maxLines: Int = Int.MAX_VALUE,
    text: String,
    onTextChanged: (String) -> Unit,) {

    OutlinedTextField(
        modifier = Modifier.then(modifier),
        value = text,
        onValueChange = onTextChanged,
        colors = TextFieldDefaults.outlinedTextFieldColors(
            cursorColor = MaterialTheme.colors.primaryVariant,
            textColor = MaterialTheme.colors.primaryVariant,
            disabledBorderColor = Color.Transparent,
            errorBorderColor = Color.Transparent,
            focusedBorderColor = Color.Transparent,
            unfocusedBorderColor = Color.Transparent),

        placeholder = {
            Text(
                text = "Enter Title...",
                color = Color.LightGray,
                fontSize = fontSize)
        },
        textStyle = TextStyle(fontSize = fontSize),
        keyboardOptions = KeyboardOptions(keyboardType =
KeyboardType.Text),
        maxLines = maxLines, )
}

```

Функція для копіювання текстового вмісту в буфер обміну описано у лістингу 3.6.

### Лістинг 3.6 – Фрагмент коду DetailScanFragment.kt

```
private fun copyToClipboard(input: String?)=input?.let{ text ->
    val clipboardManager =
requireActivity().getSystemService(Context.CLIPBOARD_SERVICE) as
ClipboardManager
    val clip = ClipData.newPlainText("raw_data", text)
clipboardManager.setPrimaryClip(clip)
lifecycleScope.launch{snackbarHostState.showSnackbar(message=get
String(R.string.copied_clip)) }
}
```

Функція для надсилання текстового вмісту описано у лістингу 3.7.

### Лістинг 3.7 – Фрагмент коду DetailScanFragment.kt

```
private fun share(input: String?) = input?.let { text ->
    val shareIntent = Intent().apply {
        action = Intent.ACTION_SEND
        putExtra(Intent.EXTRA_TEXT, text)
        type = "text/plain"
    }
    val intent = Intent.createChooser(shareIntent, null)
    startActivity(intent)
}
```

## 4 ТЕСТУВАННЯ

На головному екрані зберігаються результати розпізнавання та можливість почати розпізнавання (рисунок 4.1).

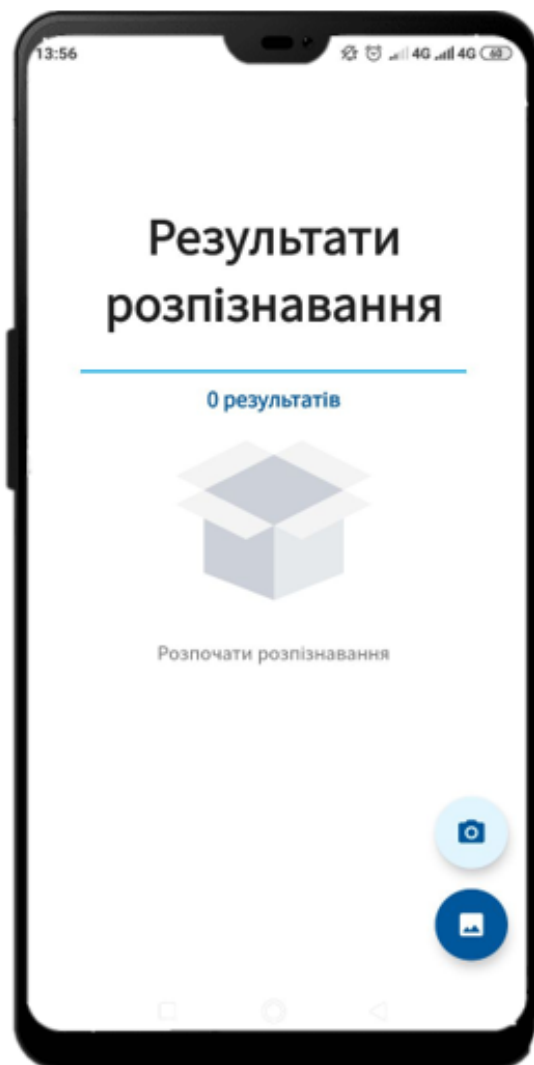


Рисунок 4.1 – Головний екран

Спочатку було проведено тестування для завантаження фото з пам'яті телефону. Було обрано зображення з галереї. Обране зображення можна обрізати потрібного розміру.

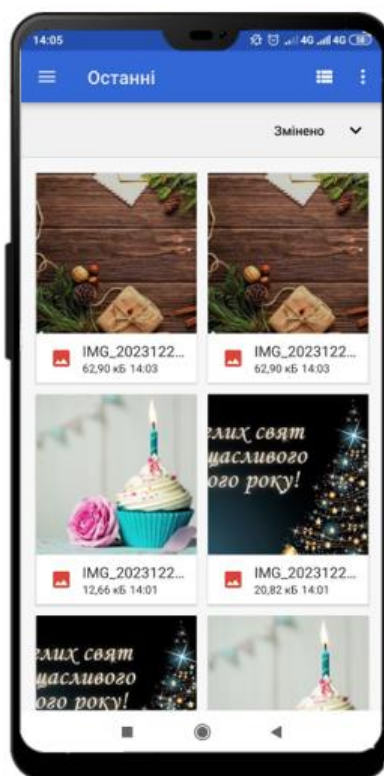


Рисунок 4.2 – Вибір зображення з галереї



Рисунок 4.3 – Вибір зображення з галереї



Рисунок 4.4 – Можливість обрізати зображення з галереї

Після завантаження зображення розпочинається процес розпізнавання.

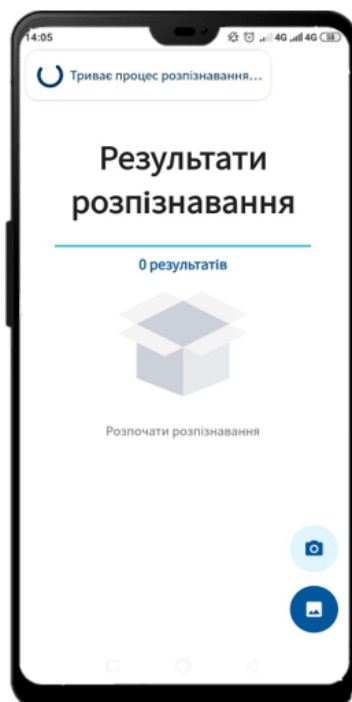


Рисунок 4.5 – Процес розпізнавання

На екрані з'явилося повідомлення, що не знайдено текст на зображенні.

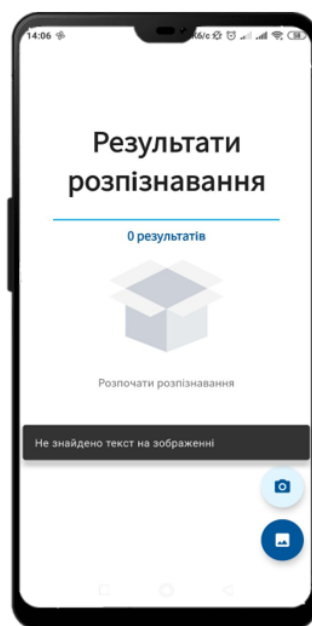


Рисунок 4.6 – Повідомлення про помилку

Також можна додавати зображення за допомогою камери.



Рисунок 4.7 – Фотографування за допомогою камери



Рисунок 4.8 – Завантаження зображення за допомогою камери



Рисунок 4.9 – Процес розпізнавання

Результат розпізнавання буде виведено на екран.



Рисунок 4.10 – Результат розпізнавання

Результату розпізнавання можна задати заголовок.



Рисунок 4.11 – Заголовок результату розпізнавання

Результат розпізнавання збережено на головний екран.

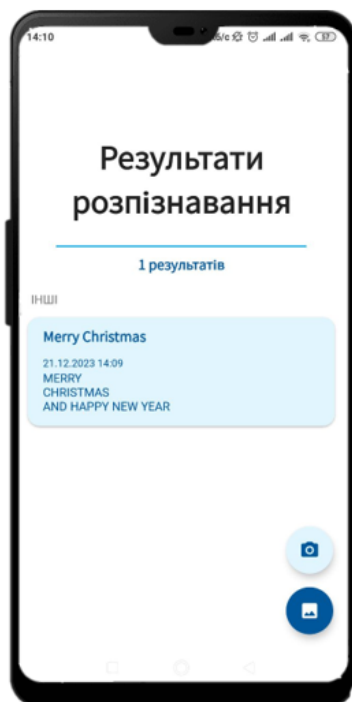


Рисунок 4.12 – Результат розпізнавання на головному екрані

Також результат можна закріпити на головному екрані.



Рисунок 4.13 – Закріплення результату

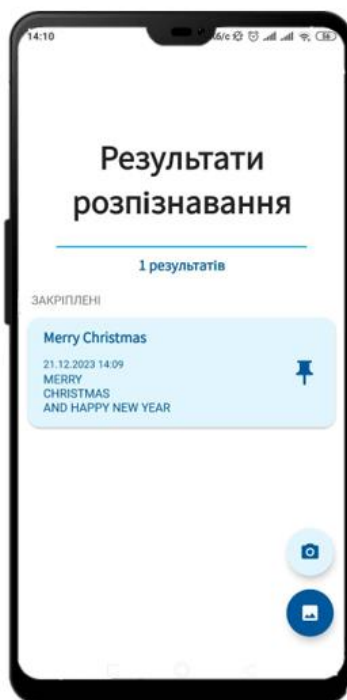


Рисунок 4.14 – Результат закріплено на головному екрані

Також є можливість видалити результат. При натисканні на кнопку видалення з'явиться попереджувальне повідомлення.

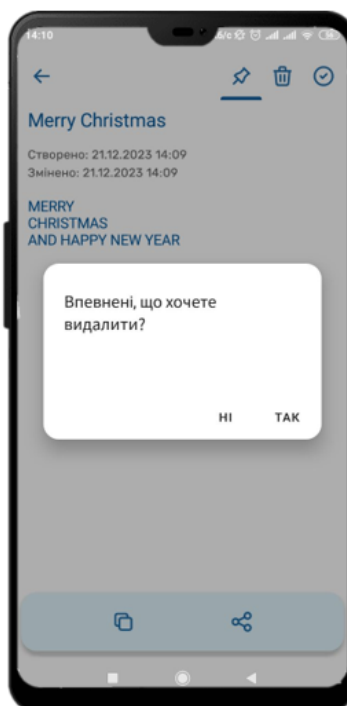


Рисунок 4.15 – Видалення результату

Також є можливість скопіювати результат. Після копіювання з'явиться повідомлення, що текст скопійовано.



Рисунок 4.16 – Копіювання результату

Також отриманий результат можна надіслати.

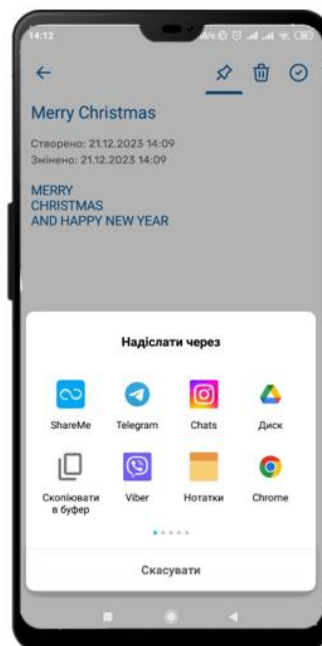


Рисунок 4.17 – Надсилання результату

Спробуємо ще додати зображення з галереї.



Рисунок 4.18 – Додавання зображення



Рисунок 4.19 – Результат розпізнавання

Якщо не заповнювати поле заголовку, то буде створено заголовок автоматично, по першому рядку тексту.

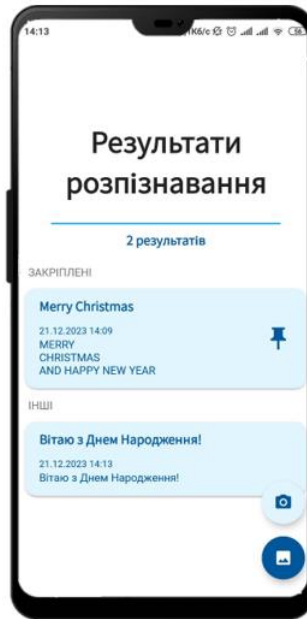


Рисунок 4.20 – Автоматичний заголовок результату

Заголовок можна записати самостійно.



Рисунок 4.21 – Процес запису заголовку

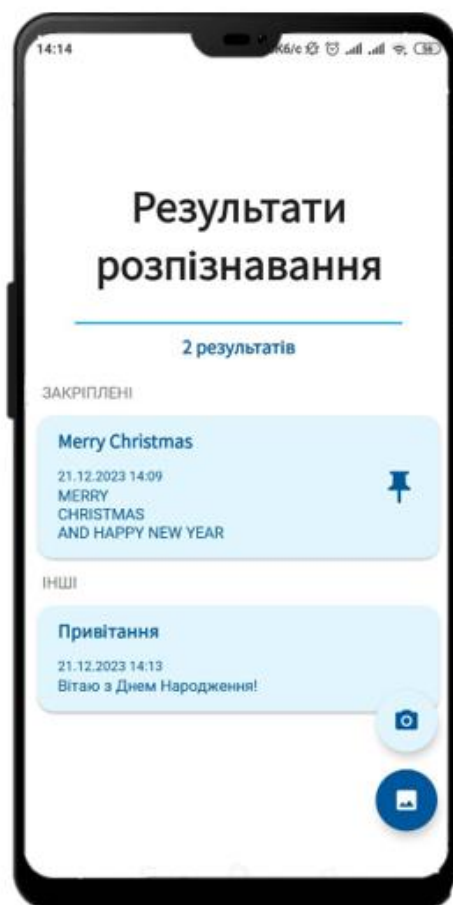


Рисунок 4.22 – Самостійно записаний заголовок результату

Результати розпізнавання на головному екрані поділені на закріплені результати та інші.

## ВИСНОВКИ

В результаті виконання кваліфікаційної роботи розроблено модель розпізнавання тексту з зображень та створено мобільний застосунок на Android. В роботі описано можливості штучного інтелекту та нейронних мереж. Розглянуто існуючі методи навчання та види нейронних мереж. Проаналізувавши особливості нейронних мереж було обрано конволюційну нейронну мережа.

В роботі було проведено порівняльний аналіз різних методів розпізнавання зображень. Отримані результати підтвердили важливість обробки та підготовки даних у роботі систем з OCR. Методом для вирішення поставленої задачі було обрано OCR. Було проаналізовано відомі мобільні операційні системи, їх переваги та недоліки. Порівнявши переваги та недоліки, було обрано ОС Android. Для застосунку було описано загальну архітектуру. На основі архітектури було написано застосунок. В роботі описано основні функції застосунку та надано лістинги до них. Потім було проведено тестування застосунку, на перевірку правильності функціонування.

Результати кваліфікаційної роботи можуть бути використані для розгортання системи розпізнавання тексту з зображень. У подальшому планується розробити систему використання для розпізнавання не лише пристроїв з Android, а й персональних комп'ютерів.

Отже, використання нейромережевих методів у розпізнаванні тексту на зображеннях є перспективним напрямком досліджень, що дозволяє автоматизувати процес розпізнавання тексту та поліпшити результати.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Штучний інтелект [Електронний ресурс]. – Режим доступу : [www/URL:https://uk.wikipedia.org/wiki/%D0%A8%D1%82%D1%83%D1%87%D0%BD%D0%B8%D0%B9\\_%D1%96%D0%BD%D1%82%D0%B5%D0%BB%D0%B5%D0%BA%D1%82](https://uk.wikipedia.org/wiki/%D0%A8%D1%82%D1%83%D1%87%D0%BD%D0%B8%D0%B9_%D1%96%D0%BD%D1%82%D0%B5%D0%BB%D0%B5%D0%BA%D1%82) – 21.12.2023 р. – Загл. з екрану.
2. Глибовець М. М., Олецький О.В. Системи штучного інтелекту [Електронний ресурс]. – Режим доступу : [www/URL:http://kist.ntu.edu.ua/textPhD/ArtificIntell.pdf](http://kist.ntu.edu.ua/textPhD/ArtificIntell.pdf) – 21.12.2023 р. – Загл. з екрану.
3. Три стадії штучного інтелекту [Електронний ресурс]. – Режим доступу : [www/URL:https://www.bbc.com/ukrainian/features-65728291](http://www.bbc.com/ukrainian/features-65728291) – 21.12.2023 р. – Загл. з екрану.
4. Що таке штучний інтелект: історія, види та складові [Електронний ресурс]. – Режим доступу : [www/URL:https://gigacloud.ua/blog/navchannja/scho-take-shtuchnij-intelekt-istorija-vidi-ta-skladovi](https://gigacloud.ua/blog/navchannja/scho-take-shtuchnij-intelekt-istorija-vidi-ta-skladovi) – 21.12.2023 р. – Загл. з екрану.
5. Штучний інтелект: сьогодні та майбутнє [Електронний ресурс]. – Режим доступу : [www/URL:https://ula.lantec.ua/statti/shtuchnij-intelekt-sogodennya-ta-majbutne](https://ula.lantec.ua/statti/shtuchnij-intelekt-sogodennya-ta-majbutne) – 21.12.2023 р. – Загл. з екрану.
6. Що таке нейронні мережі та де їх використовують? [Електронний ресурс]. – Режим доступу : [www/URL:https://incrypted.com/ua/shcho-take-nejromerezhi/](https://incrypted.com/ua/shcho-take-nejromerezhi/) – 21.12.2023 р. – Загл. з екрану.
7. Що таке нейронні мережі та як вони працюють? Класифікація штучних нейромереж [Електронний ресурс]. – Режим доступу : [www/URL:https://livingfo.com/shcho-take-nejronni-merezhi-ta-iak-vony-pratsiuiut/](https://livingfo.com/shcho-take-nejronni-merezhi-ta-iak-vony-pratsiuiut/) – 21.12.2023 р. – Загл. з екрану.
8. Що таке нейрон? [Електронний ресурс]. – Режим доступу : [www/URL:https://dovidka.biz.ua/shho-take-neyron/](https://dovidka.biz.ua/shho-take-neyron/) – 21.12.2023 р. – Загл. з екрану.
9. Перцептрон [Електронний ресурс]. – Режим доступу : [www/URL:https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%86%D0%B5%D0%BF%D1%82%D1%80%D0%BE%D0%BD](https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D1%86%D0%B5%D0%BF%D1%82%D1%80%D0%BE%D0%BD) – 21.12.2023 р. – Загл. з екрану.
10. CNN [Електронний ресурс]. – Режим доступу : [www/URL:https://www.v7labs.com/blog/convolutional-neural-networks-guide](https://www.v7labs.com/blog/convolutional-neural-networks-guide) – 21.12.2023 р. – Загл. з екрану.
11. LSTM [Електронний ресурс]. – Режим доступу : [www/URL:https://medium.com/mlearning-ai/building-a-neural-network-zoo-from-scratch-the-long-short-term-memory-network-1cec5cf31b7](https://medium.com/mlearning-ai/building-a-neural-network-zoo-from-scratch-the-long-short-term-memory-network-1cec5cf31b7) – 21.12.2023 р. – Загл. з екрану.
12. Класифікація нейронної мережі - Різні типи основних нейронних мереж [Електронний ресурс]. – Режим доступу : [www/URL:https://uk.education-wiki.com/4761295-classification-of-neural-network](https://uk.education-wiki.com/4761295-classification-of-neural-network) – 21.12.2023 р. – Загл. з екрану.

13. Методи нейромережевого розпізнавання тексту на зображеннях [Текст]: 27-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті». Зб. матеріалів форуму. Т. 5. – Харків: ХНУРЕ. 2023. – 178 с.
14. OCR [Електронний ресурс]. – Режим доступу : [www/ URL: https://www.techtarget.com/searchcontentmanagement/definition/OCR-optical-character-recognition](http://www.techtarget.com/searchcontentmanagement/definition/OCR-optical-character-recognition) – 21.12.2023 р. – Загл. з екрану.
15. Метрика [Електронний ресурс]. – Режим доступу : [www/ URL: https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D1%80%D0%B8%D0%BA%D0%B0\\_%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0](http://www.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D1%80%D0%B8%D0%BA%D0%B0_%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0) – 21.12.2023 р. – Загл. з екрану.
16. Нейронні мережі в задачах обробки зображень [Електронний ресурс]. – Режим доступу : [www/ URL: https://web.posibnyky.vntu.edu.ua/fksa/2kvetnyj\\_komp'yuterne\\_modelyuvannya\\_system\\_procesiv/t2/24..htm](http://www.web.posibnyky.vntu.edu.ua/fksa/2kvetnyj_komp'yuterne_modelyuvannya_system_procesiv/t2/24..htm) – 21.12.2023 р. – Загл. з екрану.
17. Google Keep [Електронний ресурс]. – Режим доступу : [www/ URL: https://id.nl/huis-en-entertainment/mobiel/smartphone-en-tablet/aan-de-slag-met-google-keep-50668](http://www.id.nl/huis-en-entertainment/mobiel/smartphone-en-tablet/aan-de-slag-met-google-keep-50668) – 22.12.2023 р. – Загл. з екрану.
18. Microsoft Office Lens [Електронний ресурс]. – Режим доступу : [www/ URL: https://support.microsoft.com/uk-ua/office/microsoft-lens-%D0%B4%D0%BB%D1%8F-android-ec124207-0049-4201-afaf-b5874a8e6f2b](http://www.support.microsoft.com/uk-ua/office/microsoft-lens-%D0%B4%D0%BB%D1%8F-android-ec124207-0049-4201-afaf-b5874a8e6f2b) – 22.12.2023 р. – Загл. з екрану.
19. Для ПК і смартфона: 5 безкоштовних програм для сканування документів [Електронний ресурс]. – Режим доступу : [www/ URL: https://www.unian.ua/techno/programi-dlya-skanuvannya-dokumentiv-bezkoshtovni-skaneri-na-pk-i-smartfoni-11671114.html](http://www.unian.ua/techno/programi-dlya-skanuvannya-dokumentiv-bezkoshtovni-skaneri-na-pk-i-smartfoni-11671114.html) – 22.12.2023 р. – Загл. з екрану.
20. Evernote [Електронний ресурс]. – Режим доступу : [www/ URL: https://evernote.com/intl/en](http://www.evernote.com/intl/en) – 22.12.2023 р. – Загл. з екрану.
21. 5 найкращих безкоштовних програм для сканування для iPhone та Android [Електронний ресурс]. – Режим доступу : [www/ URL: https://cybercalm.org/novyny/5-najkrashhih-bezkoshtovnih-program-dlya-skanuvannya-dlya-iphone-ta-android/](http://www.cybercalm.org/novyny/5-najkrashhih-bezkoshtovnih-program-dlya-skanuvannya-dlya-iphone-ta-android/) – 22.12.2023 р. – Загл. з екрану.
22. Мобільна операційна систем [Електронний ресурс]. – Режим доступу : [www/ URL: https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B1%D1%96%D0%BB%D1%8C%D0%BD%D0%B0\\_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D0%B0\\_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0](http://www.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B1%D1%96%D0%BB%D1%8C%D0%BD%D0%B0_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) – 22.12.2023 р. – Загл. з екрану.
23. Переваги та недоліки сучасних смартфонів на базі Андроїд [Електронний ресурс]. – Режим доступу : [www/ URL: https://cybercalm.org/novyny/5-najkrashhih-bezkoshtovnih-program-dlya-skanuvannya-dlya-iphone-ta-android/](http://www.cybercalm.org/novyny/5-najkrashhih-bezkoshtovnih-program-dlya-skanuvannya-dlya-iphone-ta-android/) – 22.12.2023 р. – Загл. з екрану.
24. iOS проти Android: 10 переваг операційної системи apple [Електронний ресурс]. – Режим доступу : [www/ URL:](http://www)

<https://cybercalm.org/novyny/5-najkrashhih-bezkoshtovnih-program-dlya-skanuvannya-dlya-iphone-ta-android/> – 22.12.2023 р. – Загл. з екрану.

25. Операційної система iOS [Електронний ресурс]. – Режим доступу : [www/](http://www/) URL: <https://zz.te.ua/operatsiy-na-systema-ios-plusy-minusy-kryterii-vyboru-ayfona/> – 22.12.2023 р. – Загл. з екрану.

26. Android Studio - офіційне інтегроване середовище розробки Android [Електронний ресурс]. – Режим доступу : [www/](http://www/) URL: <https://blog.desdelinux.net/android-studio-el-entorno-de-desarrollo-integrado-oficial-de-android/> – 22.12.2023 р. – Загл. з екрану.

27. Android Studio [Електронний ресурс]. – Режим доступу : [www/](http://www/) URL: [https://uk.wikipedia.org/wiki/Android\\_Studio](https://uk.wikipedia.org/wiki/Android_Studio) – 22.12.2023 р. – Загл. з екрану.

28. Програмування під Android/ Структура Android програми [Електронний ресурс]. – Режим доступу : [www/](http://www/) URL: [https://uk.wikibooks.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F\\_%D0%BF%D1%96%D0%B4\\_Android/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0\\_Android\\_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8](https://uk.wikibooks.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%BF%D1%96%D0%B4_Android/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_Android_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8) – 22.12.2023 р. – Загл. з екрану.