

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до магістерської роботи
на ступінь вищої освіти магістр

на тему: **«РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РОЗУМНОГО
БУДИНКУ ДЛЯ РОЗПІЗНАВАННЯ ДОМАШНІХ ТВАРИН ЗА
ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ»**

Виконав: студент 6 курсу, групи ПДМ–61
спеціальності

121 Інженерія програмного забезпечення
(шифр і назва спеціальності/спеціалізації)

_____ Горелов А.І.

(прізвище та ініціали)

Керівник _____ Трінтіна Н.А.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

Київ – 2022

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення
Ступінь вищої освіти -«Магістр»
Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Інженерії програмного
забезпечення

Негоденко О.В.
“ ____ ” _____ 2022 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТА

ГОРЕЛОВУ АРТЕМУ ІГОРОВИЧУ

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка інтелектуальної системи розумного будинку для розпізнавання домашніх тварин за допомогою нейронних мереж »

Керівник роботи: Трінтіна Н.А., к.т.н., доцент

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

Затверджені наказом вищого навчального закладу від «13» жовтня 2021 року №230.

2. Строк подання студентом роботи «24» грудня 2021 року

3. Вхідні дані до роботи

Науково-технічна література з питань, пов'язаних з методами прогнозування часових _____ рядів;

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

4.1 Розпізнавання образів на зображеннях як об'єкт дослідження;

4.2 Методи машинного навчання для розпізнавання образів на зображеннях;
4.3 Розширення функціоналу систем розумного будинку на основі побудованих моделей;

5. Перелік демонстраційного матеріалу (назва основних слайдів)

1. Актуальність проблеми
2. Існуюче програмне забезпечення та методи розпізнавання
3. Принцип роботи згорткової нейронної мережі
4. Результат задач розпізнавання домашніх тварин
5. Архітектура бази даних
6. Логічна діаграма компонентів архітектури програмного забезпечення

6. Дата видачі завдання «02» листопада 2021

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	02.11-07.11	Виконано
2	Вимоги до системи	08.11-10.11	Виконано
3	Створення моделі для розпізнавання образів на зображенні	11.11-18.11	Виконано
4	Проектування та розробка програмного забезпечення	19.11-28.11	Виконано
5	Оформлення розділів дипломної роботи	29.11-04.12	Виконано
6	Вступ, висновки, реферат	05.12-12.12	Виконано
7	Розробка обов'язкових демонстраційних матеріалів	13.11-15.12	Виконано
8	Перевірка на антиплагіат	15.12	
9	Попередній захист роботи	21.12-23.12	
10	Проходження нормконтролю	24.12	
11	Здача роботи		

Студент _____ Горелов А.І.
(підпис) (прізвище та ініціали)

Керівник роботи _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи с. 61, рис. 28, джерел 26.

ШТУЧНИЙ ІНТЕЛЕКТ, МАШИННЕ НАВЧАННЯ, НЕЙРОННА МЕРЕЖА, РОЗУМНИЙ БУДИНОК, РОЗПІЗНАВАННЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА

Об'єкт дослідження – системи розпізнавання об'єктів, образів на зображеннях, що використовують для цього нейронні мережі.

Предмет дослідження – різні типи нейронних мереж, їх архітектур, що дадуть змогу розпізнавати образи на зображеннях.

Мета роботи – побудова додатку з використанням нейронної мережі з функціоналом розпізнавання об'єктів, образів на зображеннях, для його інтеграції в системи розумного будинку.

З останнім часом штучний інтелект захопив чималу низку рішень, що допомагають нам в повсякденному житті. На сьогодні системи розумного будинку зайняли значуще місце серед розробок у сфері інформаційних технологій. Величезні обсяги даних, що генеруються в них та різноманітність форматів цих даних не дає змогу створити універсальних механізм для їхньої продуктивної обробки. Тому, інтеграція нейронних мереж, що найкраще підходить для вирішення окремих задач, забезпечить нас високими показниками ефективності систем розумного будинку з мінімально-допустимими похибками прийнятих рішень. Рік у рік зростає інтерес до вирішення все більш складних завдань розпізнавання об'єктів, у зв'язку з потребами в автоматизації формованого спілкування в інтелектуальних системах. Тому вдосконалення реалізації розпізнавання комп'ютерними системи зображення є актуальним. Для вирішення цієї задачі запропоновано використання штучних нейронних мереж і нейрокомп'ютерів як найбільш прогресивних щодо проблем класифікації завдань на розпізнавання образів. Аналіз із запропонованих рішень показує, що досі немає такої моделі, яка б була найкращою серед усіх отриманих параметрів продуктивності. У наш час пропонуються велика кількість архітектур нейронних

мереж для застосування при розпізнаванні об'єктів. У роботі проведено аналіз найбільш популярних архітектур, визначено їхні переваги та недоліки в рамках поставленого їм завдання, а саме – розпізнавання об'єктів та істот.

ABSTRACT

The work is presented on 61 pages, 28 pictures, 26 references.

ARTIFICIAL INTELLIGENCE, MACHINE LEARNING, NEURAL NETWORK, RECOGNITION, SMART HOUSE, CONVOLUTIONAL NETWORK.

Object of research – systems for recognizing objects, images in images that use neural networks.

The subject of research – the analysis of different types of neural networks, their architectures, which will allow to recognize images in images.

The purpose of the work is to build an application using a neural network with the functionality of recognizing objects in images, for its integration into smart home systems.

Recently, artificial intelligence has captured a number of solutions that help us in everyday life. Today, smart home systems have taken a significant place among the developments in the field of information technology. The huge amount of data generated in them and the variety of formats of this data does not allow to create a universal mechanism for their productive processing. Therefore, the integration of neural networks, which is best suited for individual tasks, will provide us with high efficiency of smart home systems with minimal errors in decisions. Year by year, there is a growing interest in solving more and more complex tasks of object recognition, due to the need to automate the formed communication in intelligent systems. Therefore, improving the implementation of computer image recognition is important. To solve this problem, it is proposed to use artificial neural networks and neurocomputers as the most progressive in terms of problems of classification of image recognition tasks. The analysis of the proposed solutions shows that there is still no model that would be the best among all the obtained performance parameters. We now offer a large number of neural network architectures for use in object recognition. The paper analyzes the most popular architectures, identifies their advantages and disadvantages in the framework of their task, namely – the recognition of objects and creatures.

ЗМІСТ

ЗМІСТ	10
ВСТУП	13
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	15
1.1 Машинне навчання	15
1.2 Нейронні мережі. Історія розвитку	16
1.3 Поняття штучної нейронної мережі	18
1.4 Біологічний та штучний нейрон	19
1.5 Штучні інтелектуальні системи	22
1.6 Поняття інтелектуальної інформаційної технології	28
1.7 Аналіз існуючих рішень	31
1.7.1 Google Lens	31
1.7.2 Image Recognizer	32
1.7.3 CamFind	33
2 АНАЛІЗ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ	37
2.1 Одношарові нейронні мережі	37
2.2 Багатошарові нейронні мережі	38
2.3 Рекурентні нейронні мережі	39
2.4 Нейронна мережа прямого поширення та перцептрон	40
2.5 Нейронна мережа Гопфілда	42
2.6 Ланцюги Маркова	43
2.7 Машина Больцмана	43
2.8 Згорткові нейронні мережі	44
2.9 Нейронна мережа для розпізнавання тварин	45
2.10 Алгоритм роботи ЗНМ	46
2.11 Види навчання згорткових нейронних мереж	48
2.12 Архітектура для згорткової нейронної мережі	51
2.12.1 LeNet	51
2.12.2 AlexNet	51
2.12.3 VGG	52
2.12.4 Network-in-network	53
2.12.5 ResNet	53

	11
2.12.6 Вибір архітектури для ЗНМ	54
3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	59
3.1 Вибір платформи та архітектури	59
3.1.1 Частина бази даних	59
3.1.2 Серверна частина.....	59
3.2 Реалізація серверної частини додатку	62
ВИСНОВКИ	66
ПЕРЕЛІК ПОСИЛАНЬ	67

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

ІІ – штучний інтелект;

ІІ – нейронна мережа;

ІІ – machine learning;

ІІІ – згортова нейронна мережа;

ІІ (Boltzman machines) – машини Больцмана;

ІІ (Markov chains) – ланцюги Маркова;

ІІ – операційна система;

ІІ – база даних;

ІІІ – система управління базами даних;

ІІ – програмне забезпечення;

ІІ – structured query language;

ІІ – entity framework;

ІІ – language integrated query;

ІІ – application programming interface;

ІІ – model-view-controller;

ІІ – data access layer;

ІІ – business logic layer;

ВСТУП

В сьогоднішній штучний інтелект, в саме нейронні мережі, вже не є таким екзотичним явищем як декілька десятиріч тому. Зараз вони присутні в кожному смартфоні і виконують повсякденні функції для покращення користувацького досвіду. Це зобумовлено їхньою продуктивністю та гнучкістю, а саме:

- здатність ефективно обробляти великі масиви неоднотипних даних;
- відсутність потреби розробляти алгоритм для кожного варіанту перебігу подій;
- змога нейронних мереж самоаналізувати себе та навчатись для покращення тих чи інших показників.

Нейронні мережі знайшли застосування в різних сферах діяльності людини: економіка – допомога для більш точного прогнозування курсів валют на основі історичних даних, промисловість – пошук та ідентифікація неполадок то пошкоджень на виробництві, медицина – винесення діагнозу на основі інформації, отриманої з рентгенівських знімків, аналізів, тощо. Нейронні мережі вже навчилися генерувати зображення на основі опису, генерувати цікаві оповідання на основі поданої теми та ключових слів, навіть генерувати логотипи для торгових марок на основі побажань замовника. Вони допомагають нам швидко та ефективно перекладати тексти з однієї мови на іншу, знаходити пісні, і на основі існуючої медіатеки пропонувати нам нові, знаходити фото в альбомі на основі введеного опису і т.д. Без використання нейронних мереж все було б або неможливим, або б потребувала гігантських зусиль розробників.

Нейронні мережі також використовують в системах розумного будинку. Вони допомагають автоматизувати процеси, які цього потребують та значно підвищити функціонал системи вцілому:

- контроль опалення в будинку;
- контроль вентиляції;

- контроль сигналізацій та безліч інших способів інтеграції нейронних мереж.

Є можливість розширити функціонал для тих мешканців, які мають домашнього улюбленця, або декілька. Система автоматичної подачі корму в залежності від того, який самий улюбленець цього потребує, адже нікому не потрібно годувати kota собачим кормом. Система деактивації зовнішньої сигналізації, якщо улюбленець захотів погратись у дворі. Система допуску до приміщень, коли він захотів повернутись.

Ціллю данної роботи є проектування та розробка програмного забезпечення, що дозволить розумному будинку приймати рішення на основі процесу розпізнавання тварин за допомогою інтеграції технологій машинного навчання та нейронних мереж.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Машинне навчання

Машинне навчання – один з методів функціонування штучного інтелекту, а саме – практичної реалізації його можливостей шляхом створення алгоритмів для виявлення закономірностей під час аналізу великих даних, та їх подальше використання для самонавчання. Останнє – головна особливість і пріоритетне завдання: не вирішити одну конкретну задачу напругу, а навчитися в процесі застосування рішень виконувати інші подібні завдання. Для цього використовуються математичний та статистичний аналізи, оптимізація та інші техніки опрацювання даних. Машинні алгоритми навчання використовують обчислювальні методи, щоб «вчитись» безпосередньо на основі отриманих даних, а не покладатися на заздалегідь підготовлене рівняння як модель. Тобто вчитися на досвіді – на своїх помилках та перемогах і не вірити сліпо у те, що їм сказали до цього. Машинні алгоритми адаптивно покращують свою продуктивність та ефективність у міру того, як кількість їх досвіду зростає. Алгоритми машинного навчання шукають (ще поки невідомі) природні зв'язки між даними, аналізують їх та передбачають найкращі рішення. При цьому до цих рішень машина приходиться самостійно і про їхнє можливе існування до початку роботи вона не знала. Алгоритми навчання працюють на основі того, що стратегії, алгоритми та висновки, які добре працювали в минулому, імовірно, будуть добре працювати й у майбутньому. Ці висновки можуть бути очевидними, наприклад, «оскільки сонце сходило щоранку протягом останніх 10 000 днів, воно, ймовірно, зійде і завтра вранці». Програми машинного навчання можуть виконувати завдання, не будучи явно запрограмованими для цього. Це включає в себе навчання комп'ютерів із наданих даних, щоб вони виконували певні завдання. Для простих завдань, покладених на комп'ютери, можна запрограмувати алгоритми, які повідомляють машині, як виконати всі кроки, необхідні для вирішення проблеми; з боку комп'ютера навчання не потрібне. Для більш складних завдань людині може бути

складно створити вручну необхідні алгоритми. На практиці може виявитися ефективнішим допомогти машині розробити власний алгоритм, аніж програмісти, що вказують кожен необхідний крок [10]. Для того, аби навчити машину, нам необхідні 3 речі:

- Данні. Це – фундамент, на якому тримається машинне навчання. Медичний діагноз буде сформований на основі аналізу сотень або тисяч схожих випадків. Повідомлення, щодо виявленого спаму на пошті буде сформовано на основі опрацювання великої кількості листів, а ціна на біржі буде спрогнозована на основі обробки попередніх змін на ринку. Чим більше даних надано, тим точніший результат буде отримано. Займатись збором даних також можна по-різному: вручну – це довгий процес. В результаті часу буде витрачено більше, даних буде менше, але всі вони будуть релевантні, без помилок та однотипні. Збір даних в автоматичному режимі є швидшим, та більш масштабним. Аля в якості ці данні програють тим, що зібрані вручну. Таким чином, проблему «неякісних» даних вирішують їхньою кількістю.

- Ознаки. Це – властивості, характеристики, якими може бути будь-яка необхідна інформація. Машині необхідно знати, що саме шукати, наприклад – слово в тексті, елемент в зображенні і т.д. У випадку, коли ознак надто багато – машина буде працювати повільно та непродуктивно. Вибір коректних ознак є однією з найважливіших задач при навчанні системи.

- Алгоритм. Як відомо, до результату можна прийти різними шляхами з вибором тих чи інших способів. Від вибору методу буде залежати швидкість, ефективність та точність моделі.

1.2 Нейронні мережі. Історія розвитку

Поняття «штучна нейронна мережа» (ШНМ) було вперше сформовано у 1943 році Уореном Маккалоком та Уолтером Пітсом в фундаментальній статті про логічні вичислення. Тоді були запропоновані вакуумні лампи в якості ідеального інструменту реалізації еквівалентів нейронних мереж[2]. У 1949 Доннальд Хебб

запропонував новітній алгоритм. Він припустив, що навчання, перше за все полягає в зміні сили синаптичних зв'язків. Його теорія – типовий процес самонавчання, де випробувана система вчиться виконувати бажану задачу без втручання тих, хто члїдкує за цим. Свій внесок зробив і Марвін Мінський, завдяки дослідженню низки задач, в тому числі відомої задачі виключної диз'юнкції [3]. У 1958 Френк Розенблатт винаходить одношаровий перцептрон і демонструє його спроможність вирішувати завдання розбиття на класи за ознаками. Перцептрон став дуже популярним – його використовують для розпізнавання об'єктів, прогнозування цїн і т.д. [4]. Після періоду затишшя в сферах розробки НМ, що тривав в 60-их – 80-их роках ХХ столїття, нові, більш продуктивні персональні комп'ютери дали їм друге дихання [5]. Чималий внесок в розвиток зробив Джон Хопфїлд. Він продемонстрував, що НМ з зворотніми зв'язками може імітувати систему, що мінімізує енергію. Так з'явилася мережа Хопфїлда. Хопфїлд, ґрунтуючись на правилах навчання Гебба, показав, що задачі з нейронами можуть бути зведені до ряду моделей з спільними ознаками, розроблених на той час у фізиці неупорядкованих систем [6]. У 1986 році Девїд Еммеретт Румельхарт, Джефрі Хїнтон та Рональд Уїльямс суттєво розвинули метод зворотнього розповсюдження похибки (метод обчислення градієнту, що використовується про оновленні ваги багатшарового перцептрону) [7]. Джефрі Хїнтон у 2007 році створив алгоритми глибокого навчання нейронних мереж. При навчанні нижніх слоїв мережі, Хїнтон використовував обмежену машину Больцмана, яка являє собою стохастичну рекурентну нейронну мережу. Сам процес навчання був повільним. Необхідно проаналїзувати великі об'єми даних аби видати результат з найменшою похибкою. Після навчання мережі, отриманий застосунок мїг швидко виконувати поставлену задачу (наприклад, пошук лиць на фото). Дану функцію вмонтовано у всі цифрові фотоапарати. Подїбна технологія використовується пошуковими системами в інтернеті для класифїкації картинок.

1.3 Поняття штучної нейронної мережі

Поняття ШНМ трактують по-різному:

- з точки зору машинного навчання, НМ є окремим випадком методів розпізнавання об'єктів, дискримінантного аналізу, методів розбиття на кластери за ознаками тощо;
- з математичної точки зору навчання НМ — це завдання з великою кількістю параметрів нелінійної оптимізації;
- з погляду кібернетики, НМ використовується у завданнях адаптивного управління та як алгоритми для робототехніки.
- з погляду розвитку обчислювальної техніки та програмування, НМ – спосіб ефективно вирішувати паралельні задачі;
- З погляду штучного інтелекту, ШНМ є основою філософської течії коннективізму та основним напрямом у структурному підході щодо вивчення можливості побудови (моделювання) природного інтелекту за допомогою комп'ютерних алгоритмів [8].

Беручи до уваги вищесказане, ШНМ (ANN – artificial neural network) є обчислювальною архітектурою для обробки складних даних за допомогою безлічі пов'язаних між собою процесорів і обчислювальних шляхів. Штучні нейронні мережі, створені за образом людського мозку, здатні навчатися та аналізувати великі та складні набори даних, які за допомогою більш лінійних алгоритмів обробити вкрай складно. Традиційний цифровий комп'ютер здатний успішно вирішувати багато різних завдань. Він робить це досить швидко і точно відповідно до вказівок користувача. На жаль, він безсилий у ситуаціях, коли сам користувач не до кінця розуміє проблему, яку він має вирішити. Гірше того, стандартні алгоритми не можуть працювати з неповними даними, при тому, що в реальному житті найчастіше доводиться мати справу саме з такою інформацією [9]. З математичної точки зору НМ являють собою спосіб вирішення нелінійних задач оптимізації. У кібернетиці використовується теорія НМ для вирішення задач адаптивного управління, побудови алгоритмів для робототехніки [10].

НМ використовуються для вирішення складних завдань, що вимагають аналітичних обчислень, подібних до того, що робить людський мозок. Найбільш поширеними програмами нейронних мереж є:

- класифікація – розподіл даних за параметрами. Наприклад, на вхід надається набір однотипних даних, і за певними заданими ознаками слід кластеризувати їх. Цю роботу може зробити нейронна мережа.

- Передбачення – можливість передбачити наступний результат. Проаналізувавши низку цільових даних, НМ може передбачити наступний результат, опираючись на ті чи інші параметри;

- Розпізнавання – нині найширше застосування НМ [11].

З назви «нейронна мережа» стає зрозуміло, що саме нейрон є елементарним логічним елементом мережі.

1.4 Біологічний та штучний нейрон

З точки зору біології, нейрон – це фундаментальний елемент в мозку та нервовій системі живих організмів. Вони з'єднані між собою нервовими волокнами. По ним передаються імпульси між нейронами. Абсолютно кожна думка, дія, будь-яке мислення чи бажання, спогади, емоції і т.д. – все це результат передачі електричних імпульсів між нейронами. Чим більше між певними нейронами передається імпульс, тим сильніший між ними зв'язок. Будь-який процес вивчення нового для людини – це процес зміцнення зв'язків між нейронами, що відповідають за цю інформацію [12].

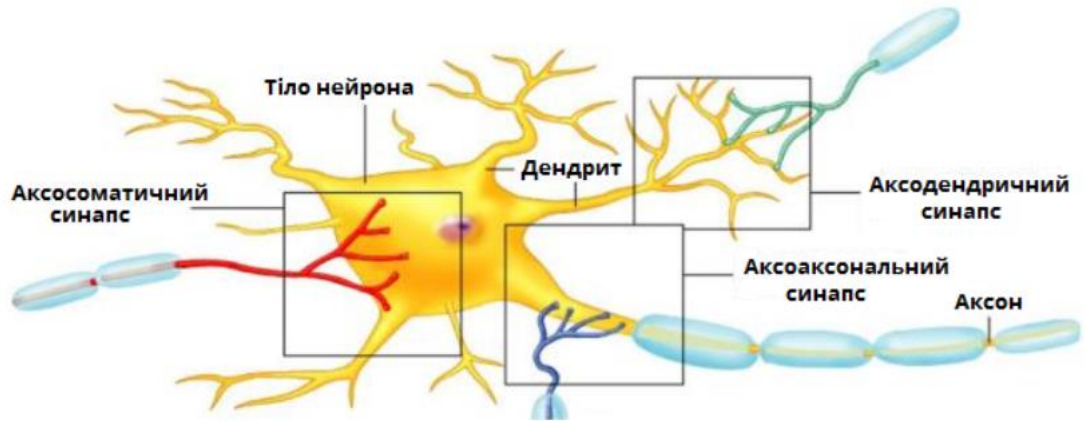


Рисунок 1.1 – Будова біологічного нейрона

Синапс – це таке утворення, що контролює силу імпульсу, для забезпечення контакту між аксоном та дендритом.

Аксон – передає імпульс нейрону.

Дендрит – приймає імпульси.

Вага синапса – це процес проходження імпульсу через синапс, в результаті якого його сила кратно збільшується. Отримавши імпульси за декількома дендритами, нейрон їх сумує. Після цього нейрон збуджується, і генерує свій власний імпульс. Він надішле його аксону якщо сумарним імпульсом буде перевищено поріг. Вага синапсів не статична і може змінювати своє значення впливаючи цим на поведінку нейрону[12]. Модель цього процесу представлена нижче:

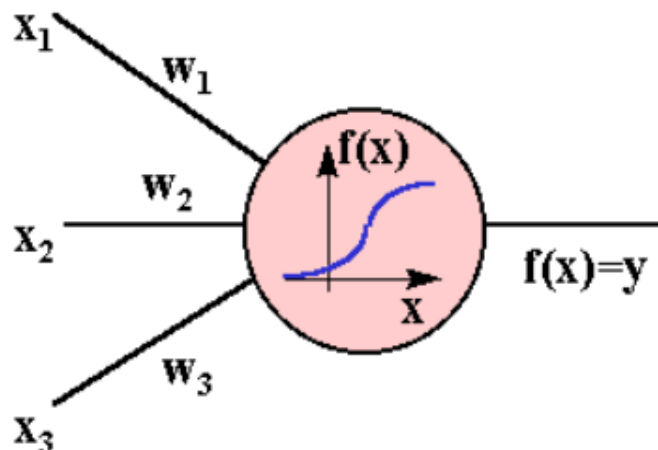


Рисунок 1.2 – Модель передачі імпульсів між нейронами

На малюнку x_1, x_2, x_3 – сили, що надходять до входів (дендритів). А w_1, w_2, w_3 – ваги синапсів. Сам нейрон отримує імпульси x_1w_1, x_2w_2, x_3w_3 . Формула отриманого сумарного імпульсу матеми вигляд (1.1).

$$x = x_1w_1 + x_2w_2 + x_3w_3 \quad (1.1)$$

Відповідно, сила вихідного імпульсу дорівнює (1.2)

$$f(x) = f(x_1w_1 + x_2w_2 + x_3w_3) \quad (1.2)$$

Тож, таким чином, основним критерієм при описі нейрону є його вага та передатня функція. Числовий ряд x_k що символізує входи нейрону буде результатом. Відповідно до цього нейрон генерує певне число y на вихід [13].

В свою чергу, штучний нейрон отримує на вхід велику кількість сигналів, кожен з яких є виходом іншого нейрону. Після цього відбувається перемноження входів на відповідну вагу. Після цього отримані результати сумуються. Так визначається рівень активації нейрону [12].

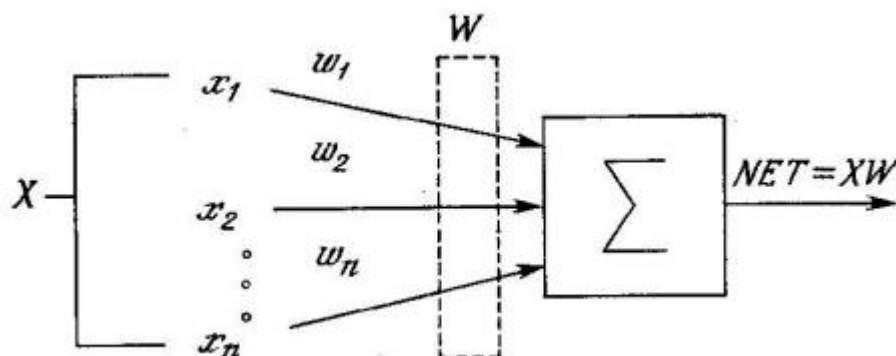


Рисунок 1.3 – Модель штучного нейрону

Тут нейрон отримує на вході велику кількість сигналів x_1, x_2, \dots, x_n . Вони представляють собою вектор x . Дані сигнали концептуально не відрізняються від тих, що отримує на входи біологічний нейрон. Після, кожен сигнал перемножується з відповідною вагою w_1, w_2, \dots, w_n . В кінці ряд добутків сумується формуючи блок Σ . Кожна з ваг дорівнює силі одного синаптичного зв'язку в біологічному нейроні. Вихід – NET – створюється з підсумовуючого блоку, де зважені елементи сумуються алгебраїчно. Активаційна функція формується

шляхом перетворення виходу лінійною функцією. Дана функція позначається F . Її вихідний сигнал = $OUT(1.3)$.

$$OUT = K (NET) \quad (1.3)$$

де K – постійна, граничної функції,

$OUT = 1$, якщо $NET > T$,

$OUT = 0$ в інших випадках,

T – постійна гранична величина, яка точніше моделює нейронну мережу [12].

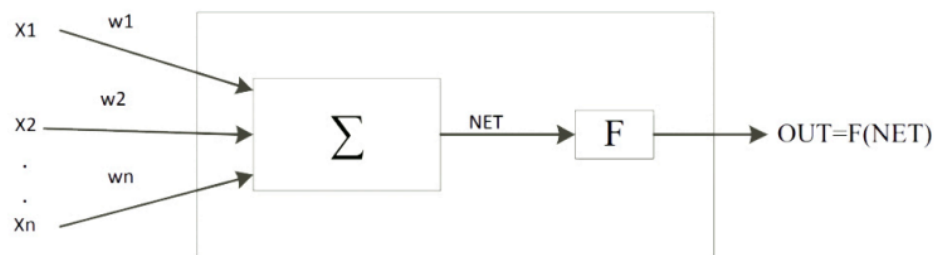


Рисунок 1.4 – Схема активаційної функції нейрону

Модель ШН не звертає увагу на більшість властивостей та параметрів біологічного нейрону. Однією такою властивістю є затримки в часі, які впливають на динаміку всієї системи. В ШН вихідний сигнал формується відразу після вхідного. Крім того, штучний нейрон не враховує впливу синхронізуючої функції біологічного нейрону [14].

1.5 Штучні інтелектуальні системи

Інтелектуальна система – це технічна або програмна система, яка може вирішувати задачі, що належать до конкретної предметної області, знання про яку зберігаються в пам'яті такої системи. Структура інтелектуальної системи включає 3 основних блока:

- базу знань;
- механізм виводу рішень;
- інтелектуальний інтерфейс [15].

Штучні інтелектуальні системи вирізняються своїми характерними особливостями:

1. Високий рівень комунікативних здібностей. Штучна інтелектуальна система має змогу опрацьовувати довільні, формальні та неформальні запити користувачів, а сформована відповідь максимально наближена до природньої мови спілкування.

2. Здатність вирішувати погано формалізовані задачі. Дуже зручно, коли система має оброблені, чіткі вхідні данні, алгоритми для врахування навколишніх чинників та конкретне, сформоване рішення. Штучна інтелектуальна система має змогу вирішувати задачі без всього цього та надавати користувачу задовільний результат

3. Змога самовдосконалюватись. Штучна інтелектуальна система, як і людина, формує свої уміння методом спроб і помилок. Інтегрована в неї система заохочення та покарання дає змогу сформувати пам'ять та досвід. Чим більше вивчено данних, чим більше система давала правильну відповідь і помилялась, тим більша ймовірність того, що наступний результат буде вірним.

4. Адаптивність – здатність до розвитку системи відповідно до об'єктивних змін моделі проблемної області [16].

Види штучних інтелектуальних систем представлені на рисунку 5.



Рисунок 1.5 – види штучних інтелектуальних систем

У розумних баз даних є можливість робити виборки інформації, що задовільняють запиту користувача. Ця інформація може явно не зберігатись в базі, але інтелектуальна система видасть відповідний до запиту результат.

Природній мовний інтерфейс надає змогу інтерпретувати мовні конструкції в апаратний рівень системи. Тобто, зникає необхідність знати синтаксис побудови запитів машині. Замість цього система проводить морфологічний та семантичний аналіз запиту, і, на основі нього, формує апаратний запит в машині. Цей функціонал найшов застосування в таких сферах:

- доступ до інтелектуальних баз даних;
- контекстний пошук текстової інформації;
- голосові команди в системах керування;
- правильний переклад з різних мов з урахуванням всіх особливостей та мовних конструкцій.

Гіпертекстові системи слугують для пошуку інформації за ключовими словами (тегами). Механізм пошуку працює, в першу чергу, з базою знань ключових слів, і тільки потім безпосередньо з текстом.

Контекстні довідкові системи – суміш інтелектуальних гіпертекстових систем і природньо-мовних систем. На відміну від звичайних систем допомоги, тут користувачу треба лише описати проблему, а система сама конкретизує та виділить проблему, які необхідно вирішити. Такі системи відносяться до класу систем розповсюдження знань.

Системи когнітивної графіки дозволяють реалізовувати інтерфейс користувача з штучним інтелектом за допомогою графічних образів, які генеруються відповідно до тих подій, що відбуваються навколо. Такі системи використовуються в моніторингу та управлінні оперативними процесами. Графічні образи у наочному та інтегрованому вигляді описують безліч параметрів досліджуваної ситуації. Наприклад, стан складного керованого об'єкта відображається в вигляді людського обличчя, на якому кожна риса відповідає за будь-який параметр, а загальний вираз обличчя дає інтегровану характеристику ситуації. Системи когнітивної графіки широко використовуються також у

навчальних та тренажерних системах на основі використання принципів віртуальної реальності, коли графічні образи моделюють ситуації, в яких учню необхідно приймати рішення та виконувати певні дії.

Експертні системи призначені для вирішення завдань на основі накопичення знань, що відображають досвід фахівців у даній області.

Багатоагентні системи – це динамічні системи, для яких характерна інтеграція в базі знань кількох різномірних джерел знань, що обмінюються між собою одержуваними результатами на динамічній основі. Для багатоагентних систем характерні такі особливості:

- проведення альтернативних міркувань на основі використання різних джерел знань із механізмом усунення протиріч;
- розподілене вирішення проблем, що розбиваються на паралельно розв'язувані підпроблеми, що відповідають самостійним джерелам знань;
- застосування безлічі стратегій роботи механізму виведення висновків у залежності від типу розв'язуваної проблеми;
- обробка великих масивів даних, що містяться у базі даних;
- використання різних математичних моделей та зовнішніх процедур, збережених у основі моделей;
- здатність переривання розв'язання задач у зв'язку з необхідністю отримання додаткових даних та знань від користувачів, моделей, паралельно вирішуваних підпроблем.

Індуктивні системи використовують узагальнення прикладів за принципом від приватного до загального. Процес класифікації прикладів здійснюється так:

1. Вибирається ознака класифікації з безлічі заданих (або послідовно, або за яким-небудь правилом, наприклад, відповідно до максимального числа одержуваних підмножин прикладів).
2. За значенням обраної ознаки безліч прикладів розбивається на підмножини.

3. Виконується перевірка, чи належить кожна підмножина прикладів, що утворилася одному підкласу.

4. Якщо якась підмножина прикладів належить одному підкласу, тобто у всіх прикладів підмножини збігається значення класотворчої ознаки, то процес класифікації закінчується (при цьому інші ознаки класифікації не розглядаються).

5. Для підмножин прикладів з незбігаючим значенням класотворчого ознаки процес класифікації триває, починаючи з початку [16].

НМ є пристроями паралельних обчислень, що складаються з множини взаємодіючих простих процесорів. Кожен процесор такої мережі має справу тільки з сигналами, які він періодично отримує, та сигналами, які він періодично надсилає іншим процесорам.

Адаптивна інформаційна система – це інформаційна система, яка змінює свої алгоритми відповідно до зміни моделі проблемної області. При цьому:

- адаптивна інформаційна система має у кожний момент часу адекватно підтримувати організацію бізнес-процесів;
- адаптивна інформаційна система повинна проводити адаптацію щоразу, як виникає потреба у реорганізації бізнес-процесів;
- реконструкція інформаційної системи повинна проводитися швидко та з мінімальними витратами.

Ядром адаптивної інформаційної системи є модель, що постійно розвивається, проблемної галузі (підприємства), що підтримується у спеціальній базі знань – репозиторії. На основі ядра здійснюється генерація чи конфігурація програмного забезпечення. Таким чином, проектування та адаптація інтелектуальних систем зводиться, перш за все, до побудови моделі проблемної галузі та її своєчасного коригування [17].

Отже, можна класифікувати штучні інтелектуальні системи за типом проблем, що вони вирішують:

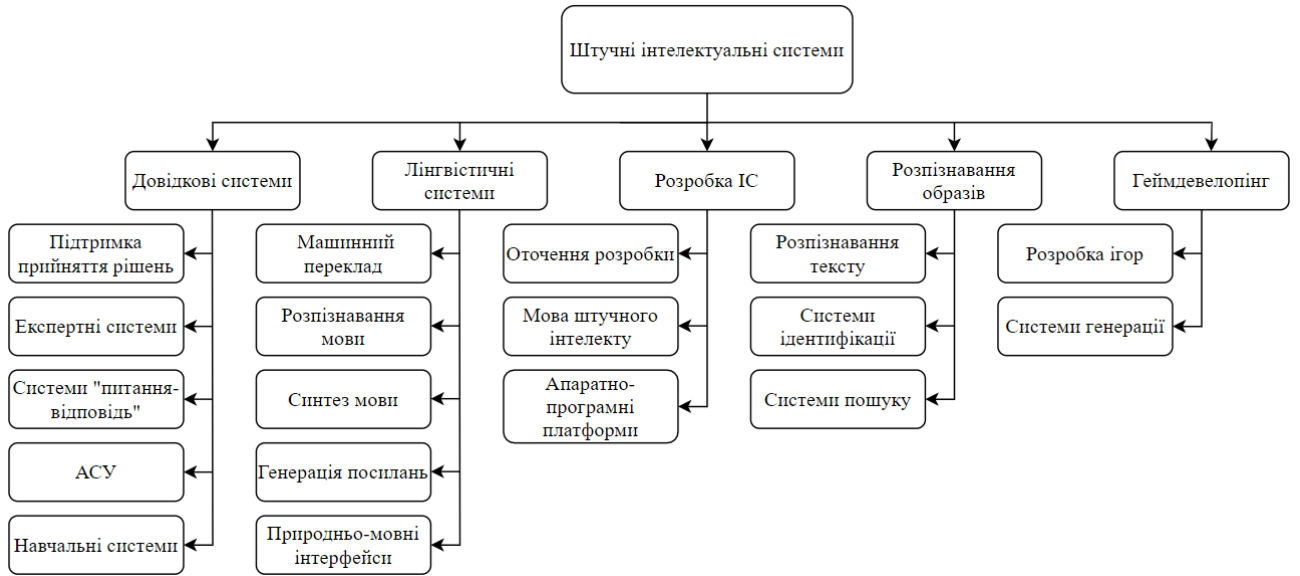


Рисунок 1.6 – Класифікація штучних інтелектуальних систем за типом вирішуваної проблеми

Класифікувати штучні інтелектуальні системи також можна за методами, які вони використовують:

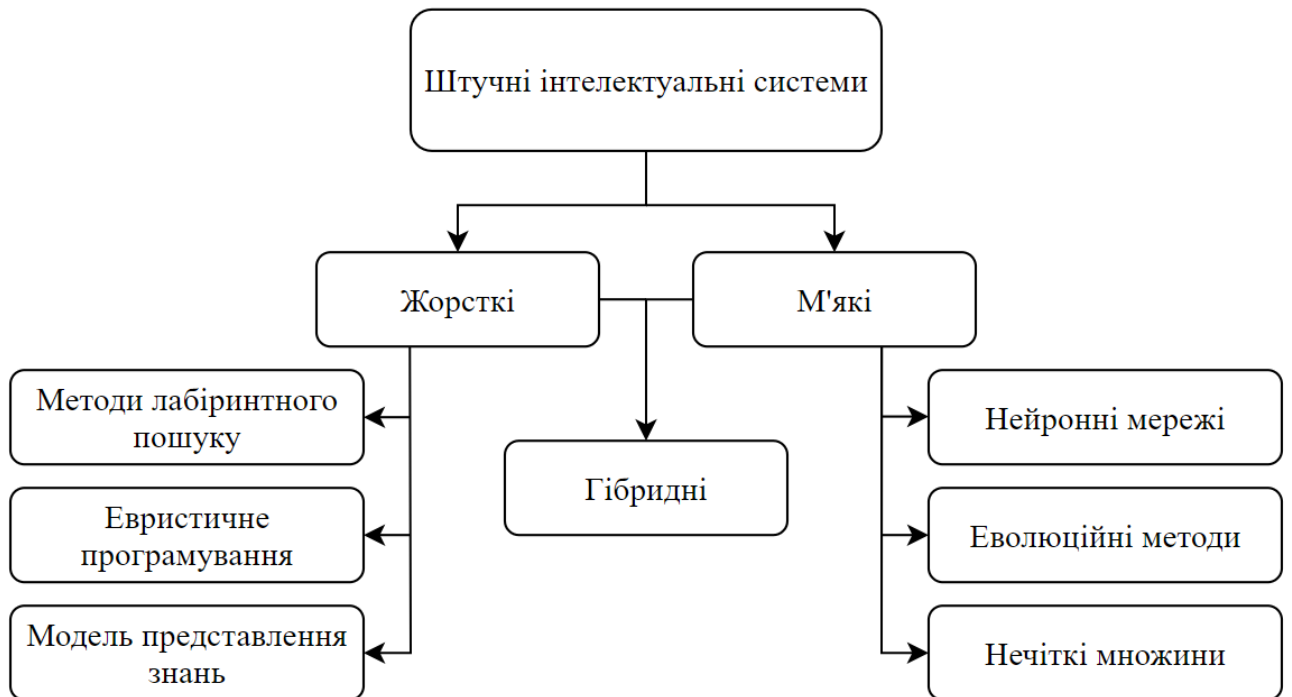


Рисунок 1.7 – Класифікація штучних інтелектуальних систем за використовуваними методами

М'які обчислення (Soft Computing) – це складна комп'ютерна методологія, в основі якої лежать нечітка логіка, генетичні розрахунки, нейрокомп'ютеринг та вираховання ймовірностей.

Жорсткі обчислення – традиційні комп'ютерні обчислення (не м'які).

Гібридні системи – системи, що використовують більш ніж одну комп'ютерну технологію (у разі інтелектуальних систем – технології штучного інтелекту) [16].

Штучні інтелектуальні системи також поділяються на системи загального призначення та спеціалізовані системи:



Рисунок 1.8 – Класифікація штучних інтелектуальних систем за типом призначення

1.6 Поняття інтелектуальної інформаційної технології

Інтелектуальні інформаційні технології – це інформаційні технології, які допомагають людині прискорити аналіз політичної, економічної, соціальної та технічної ситуації, а також – синтез управлінських рішень. При цьому

використовувані методи не обов'язково мають бути логічно несуперечливі чи копіювати процеси людського мислення. Використання ІТ на практиці передбачає облік специфіки проблемної галузі, яка може характеризуватись наступним набором ознак:

- якість та оперативність прийняття рішень;
- нечіткість цілей та інституційних кордонів;
- множинність суб'єктів, які беруть участь у вирішенні проблеми;
- хаотичність, флюктування та квантованість поведінки середовища;
- множинність факторів, що взаємовпливають один на одного;
- слабка формалізованість, унікальність, нестереотипність ситуацій;
- латентність, прихованість, неявність інформації;
- девіантність реалізації планів, значущість малих дій;
- парадоксальність логіки рішень та ін.

ІТ формуються при створенні інформаційних систем та інформаційних технологій для підвищення ефективності прийняття рішень в умовах, пов'язаних з виникненням проблемних ситуацій. У цьому випадку будь-яка життєва чи ділова ситуація – від вибору партнера у житті до соціального конфлікту – описується як деяка пізнавальна модель (когнітивної схеми, архетипу, фрейму та ін.), яка згодом використовується як основа для побудови та проведення моделювання, у тому числі – комп'ютерного. Гносеологічний фундамент ІТ найбільш явно бачиться в роботах Канта, Гегеля, Гуссерля. Явну історію ІТ зручно розпочати із середини ХХ століття, коли з'явився термін «Штучний інтелект». Історія ІТ починається з середини 1970-х років і пов'язується із спільним практичним застосуванням інтелектуальних інформаційних систем, систем штучного інтелекту, систем підтримки рішень та інформаційних систем. Ця історія пов'язана також із розвитком трьох наукових напрямів: комп'ютерної філософії, комп'ютерної психології та просунутої комп'ютерної науки. З організаційно-технологічного боку ІТ доповнюються прогресом у створення: ситуаційних центрів, інформаційно-аналітичних систем. Програмноматематичне забезпечення

складають еволюційні обчислення та генетичні алгоритми, системи підтримки спілкування людини з комп'ютером природною мовою, когнітивне моделювання, системи автоматичного тематичного рубрикування документів, системи стратегічного планування, інструментарій технічного та фундаментального аналізу фінансових ринків, системи управління якістю, системи управління інтелектуальною власністю та ін.

З середини 1940-х аж до ранніх 1970-х років створення ІТ розглядалося переважно у межах логічного розв'язання задач. Цей період розвитку ІТ характеризується порівняно великою визначеністю та низькою динамічністю об'єкта управління. Разом з тим уже в 1943 році з'явилися «продукції Поста» та методи вирішення некоректних (зворотних) завдань на метризованих просторах, а в 1947 для моделювання складних економічних ситуацій активно почали використовувати методи причинного нелогічного висновку, які пізніше лягли в основу методів системної динаміки, немонотонних обчислень, когнітивного моделювання. Створення центрів управління польотами, організація штабних робіт із застосуванням засобів візуалізації та автоматизації, зарубіжні публікації на тему створення спеціальних ситуаційних центрів надихнули у 1970-і роки інженерів на створення ситуаційних кімнат для вдосконалення управління великими соціальними та інституційними системами. У створенні таких кімнат та інтелектуальних технологій більше уваги стало надаватися засобам візуалізації, діалоговим системам, які допомагають використовувати бази знань та моделі для вирішення погано структурованих проблем. У середині 1970-х років на основі ІТ у корпоративному світі починають розвиватися системи підтримки рішень для ефективного управління ресурсами, здійснення контролінгу. Ряд чудових практичних ідей та результатів, наприклад, пов'язаних з теорією нейронних мереж, багатоагентних та активних систем, оптичних та голографічних процесорів виникло саме в цей час. Той період можна відзначити успіхами у створенні всеосяжних моделей ситуаційного управління регіонами у періоди криз. Його характеризує віра у практично необмежені можливості штучного інтелекту.

1.7 Аналіз існуючих рішень

Так як ціллю данної роботи є побудова застосунку з можливістю розпізнавати тварин, необхідно розглянути існуючі рішення, виокремити їхні переваги та недоліки. На сьогоднішній день існує багато додатків, переважно мобільних, які мають функціонал розпізнавання об'єктів на фото, але більшість з них заточена на задачі розпізнавання будівель, продуктів, предметів побуту і т.д.

1.7.1 Google Lens

Google Lens – це технологія розпізнавання зображень, розроблена Google, покликана збирати відповідну інформацію, що стосується об'єктів, які вона ідентифікує за допомогою візуального аналізу на основі нейронної мережі. Вперше оголошена під час Google I/O 2017, спочатку вона була представлена як окремий додаток, а потім інтегрована у стандартний додаток для камер Android.

Google Lens працює наступним чином: під час направлення камери телефону на об'єкт, система намагається його ідентифікувати, аналізуючи штрих-код, QR-коди, мітки та текст. Якщо об'єкт ідентифікований, застосунок запропонує посилання на вебсторінку, де надається повна інформація про нього [18].



Рисунок 1.9 – Приклад роботи застосунку Google Lens

Основними перевагами даної системи є:

- підтримка мобільних платформ;
- швидкодія;
- здатність розпізнавати безліч об'єктів.

До недоліків можна віднести:

- неспроможність використання данного застосунку в системах розумного будинку;
- відсутність української локалізації;
- відсутність можливості використання додатку на стаціонарних комп'ютерах.

1.7.2 Image Recognizer

Застосунок дозволяє визначити, що саме за об'єкт зображений на картинці. Все, що необхідно зробити користувачу – відправити фото на аналіз. Після цього система видасть результат, в якому буде вказано з якою ймовірністю це той чи інший предмет.

Основними перевагами даної системи є:

- підтримка мобільних платформ;
- здатність розпізнавати безліч об'єктів.

До недоліків можна віднести:

- неспроможність використання данного застосунку в системах розумного будинку;
- обмежений функціонал;

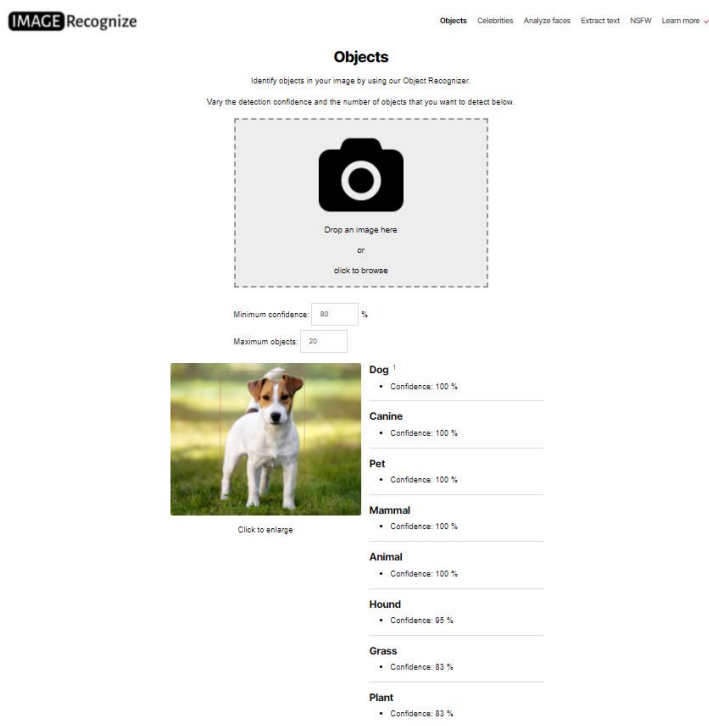


Рисунок 1.10 – Приклад роботи застосунку Image Recognize

1.7.3 CamFind

CamFind — це мобільний додаток для візуального пошуку та розпізнавання зображень. Він дає змогу користувачам ідентифікувати будь-який товар, просто сфотографувавши за допомогою смартфона, надаючи низку інформації, включаючи пов'язані зображення, результати місцевих покупок, порівняння цін та результати в Інтернеті.

Основними перевагами даної системи є:

- підтримка мобільних платформ;
- здатність розпізнавати безліч об'єктів;
- надання інформації про предмети, що є пов'язаними з імпортованим на фото.

До недоліків можна віднести:

- неспроможність використання данного застосунку в системах розумного будинку;

- обмежений функціонал;
- безліч реклами.

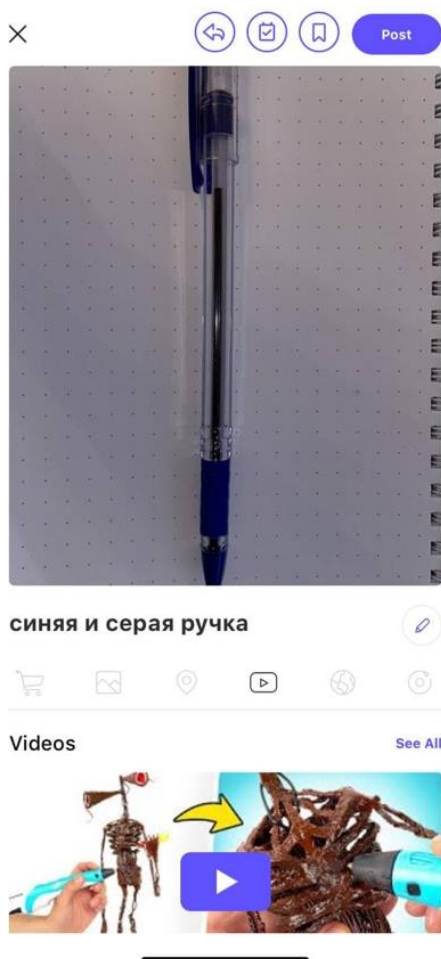


Рисунок 1.11 – Приклад роботи застосунку CamFind

Результат аналізу існуючих рішень з функціоналом розпізнавання об'єктів на фото наведений у таблиці 1.

Таблиця 1 – Порівняння існуючих рішень

Критерій порівняння	Google Lens	Image Recognizer	CamFind
Функціональність	+	-	-
Зручність використання	+	+	+
Можливість розширення функціоналу	-	-	-
Кросплатформенність	-	+	-
Відкрите API	-	+	-

Підсумовуючи результати аналізу декількох існуючих рішень, жоден з них не підходить для інтеграції в системи розумного будинку. Жоден додаток неможливо розширити додатковим функціоналом. Лише Image Recognizer є кросплатформним, і може функціонувати на ПК. Для майбутнього додатку важливими критеріями будуть:

- функціональність;
- простота використання;
- можливість інтеграції в системи розумного будинку;
- кросплатформність.

Висновки до розділу

В даному розділі описано об'єкт та предмет дослідження. Були розкриті поняття машинного навчання, штучних нейронних мереж. Окрім цього розкрита історія створення та прогресу нейронних мереж. Наведено події, які давали потужний поштовх до подальшого розвитку ШНМ.

Було розкрито поняття штучних інтелектуальних систем. Їхні класи, на які вони поділяються, були описані. Також було розкрито поняття інтелектуальних інформаційних технологій.

Був проведений аналіз деяких існуючих рішень. Було обрано три популярних додатки, які мають змогу розпізнавати та класифікувати предмети на зображеннях: Google Lens, Image Recognizer, CamFind. Всі ці додатки добре справляються зі своєю задачею, але їхня інтеграція в систему розумного будинку неможлива. Беручи до уваги їхні переваги та недоліки, були сформовані критерії для майбутнього додатку. Для його розробки слід глибше вивчити питання інтеграції нейронних мереж, обрати алгоритм, що буде задовільняти всім умовам, обрати технології, які нададуть всі ті переваги, яких не вистачало в розглянутих рішеннях.

2 АНАЛІЗ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ

Для інтеграції нейронної мережі у майбутній застосунок, слід проаналізувати існуючі алгоритми та обрати той, який максимально точно та ефективно буде вирішувати поставлену задачу – розпізнавання образів.

2.1 Одношарові нейронні мережі

Одношарова штучна нейронна мережа – найпростіша мережа, що складається з низки нейронів, які утворюють шар (layer).

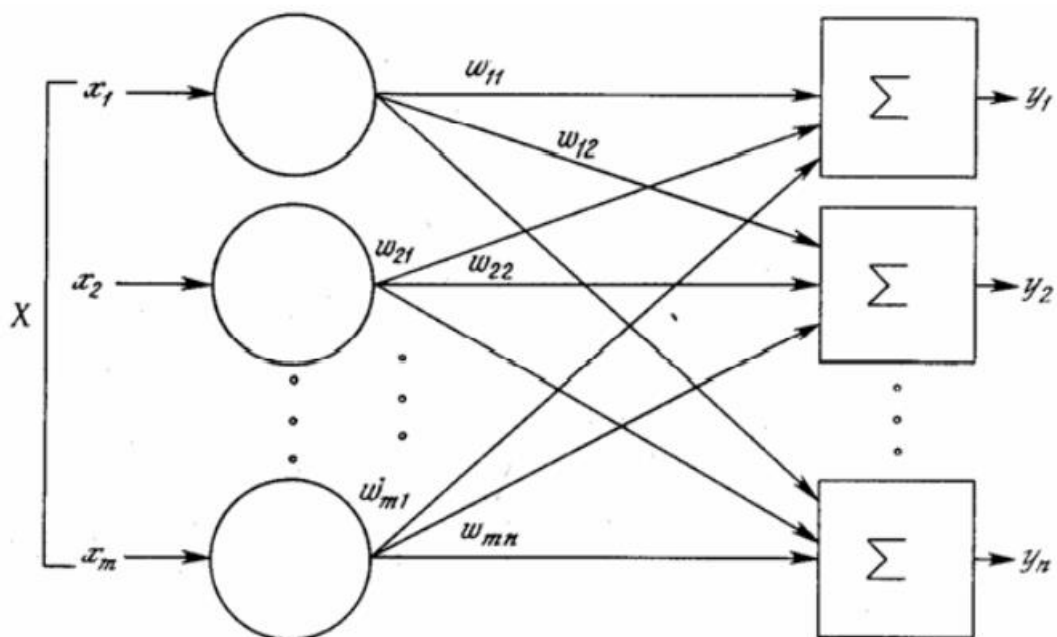


Рисунок 2.1 – Одношарова нейронна мережа

Кола, що знаходяться ліворуч, слугують лише для розподілу вхідних сигналів. Нейрони, що позначені квадратами, роблять певні обчислення, відповідно до поставленої задачі. Кожен елемент з множини входів X сполучений з усіма нейронами. W – ваги всіх елементів матриці. Матриця має m рядків та n стовпців, де m – кількість входів, n – кількість нейронів. Для прикладу, $w_{2,3}$ – являє собою вагу, яка зв'язує третій вхід з другим нейроном. Таким чином, обчислення

вихідного вектора N , компонентами якого є виходи **OUT** нейронів, зводиться множення матриць $N = XW$, де N та X – вектори-рядки [20].

2.2 Багатошарові нейронні мережі

Багатошарові штучні нейронні мережі – складніші та потужніші мережі, що володіють великими обчислювальними можливостями. Багатошарові мережі можуть будуватися з шарів, які складають каскади. Входом одного шару може бути вихід іншого. Така конфігурація нейронів імітує структури певних відділів головного мозку людини.

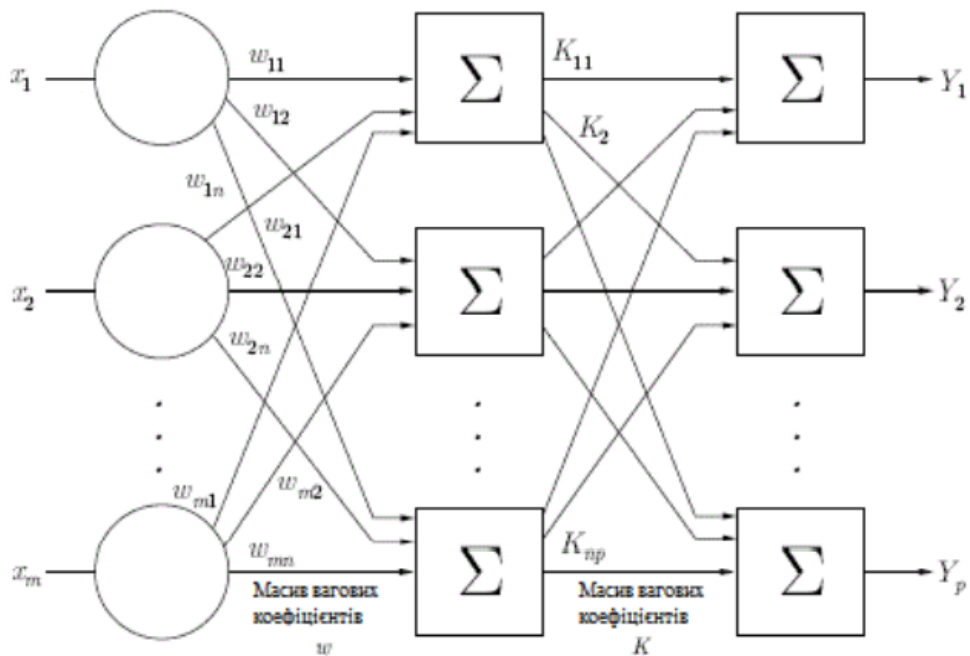


Рисунок 2.2 – Двошарова нейронна мережа

Так як активаційна функція між шарами є лінійною, то не можна безсумнівно сказати, що багатошарова нейронна мережа буде продуктивнішою аніж одношарова. Обчислення виходу шару полягає в множенні вхідного вектора на першу матрицю ваги з подальшим множенням (якщо нелінійна активаційна функція відсутня) результуючого вектора на другу матрицю ваги (2.1):

$$OUT = (XW_1)W_2$$

(2.1)

Оскільки множення матриць є асоціативним, то (2.2)

$$(\mathbf{XW}_1)\mathbf{W}_2 = (\mathbf{XW}_1\mathbf{W}_2) \quad (2.2)$$

З наведеної вище формули випливає, що один шар з ваговою матрицею рівною результату перемноження двох матриць дорівнює двошаровій лінійній мережі. Отже, будь-яка n-шарова лінійна мережа може бути заміщеною еквівалентною одношаровою. Але, обчислювальні здатності одношарових мереж значно програють багатшаровим. Якщо необхідно розширити обчислювальні можливості мереж в порівнянні з одношаровими, слід використовувати нелінійну активаційну функцію [21].

2.3 Рекурентні нейронні мережі

Рекурентні нейронні мережі – вид нейронних мереж, де зв'язки між елементами утворюють спрямовану послідовність. Завдяки цьому з'являється можливість обробляти серії подій у часі або послідовні просторові ланцюжки. На відміну від багатшарових перцептронів, рекурентні мережі можуть використовувати внутрішню пам'ять для обробки послідовностей довільної довжини. Тому такі мережі застосовні у таких завданнях, де щось цілісне розбите на частини, наприклад: розпізнавання рукописного тексту чи розпізнавання мови [22]. Так як у перцептрона немає пам'яті, і попередні результати безслідно зникають, виникає проблема з формуванням чітких послідовностей та рядів. Відповідно до цього, питання формування пам'яті в кожному нейроні набуло чинності. Саме так з'явилися рекурентні нейронні мережі. Данна мережа має змогу легко навчитися вимовляти окремі звуки чи слова. Так як кожен нейрон має пам'ять попереднього вичислення, він може більш точно та правильно передбачити наступний результат.

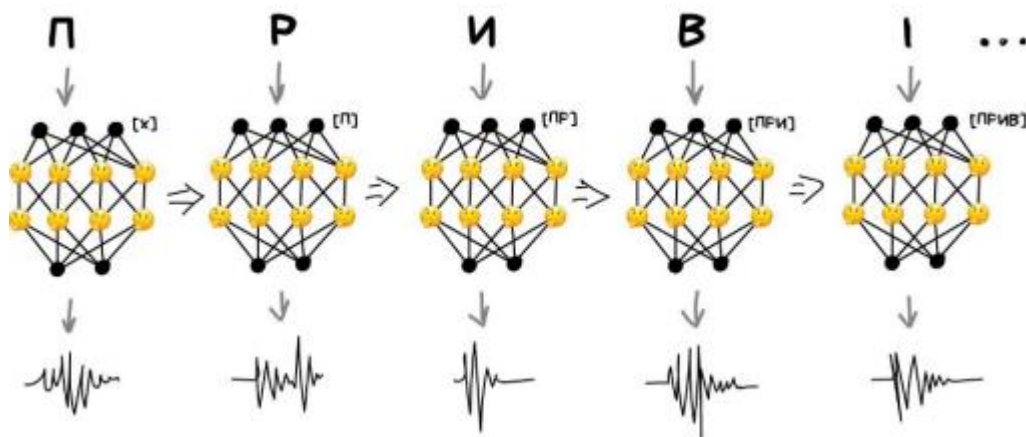


Рисунок 2.3 – Рекурентна НМ

2.4 Нейронна мережа прямого поширення та перцептрон

Спосіб роботи та схема поширення сигналу нейронної мережі прямого поширення і перцептрону є дуже простими: сигнал на вході нейрону є сигналом на вихідному сигналом попереднього. Найпростіша робоча мережа складається з двох вхідних і одного вихідного нейрона і може моделювати логічний вентиль – базовий елемент цифрової схеми, що виконує елементарну логічну операцію. Нейронну мережу прямого поширення зазвичай навчають методом зворотного поширення помилки, подаючи моделі на вхід пари вхідних і очікуваних вихідних даних. Під помилкою зазвичай розуміються різні ступені відхилення вихідних даних від очікуваних (наприклад, середньоквадратичне відхилення або сума модулів різниці). За умови, що мережа має достатню кількість прихованих нейронів, теоретично вона завжди зможе встановити зв'язок між вхідними та вихідними даними. На практиці використання мереж прямого поширення обмежена, і частіше вони використовуються спільно з іншими мережами [21].

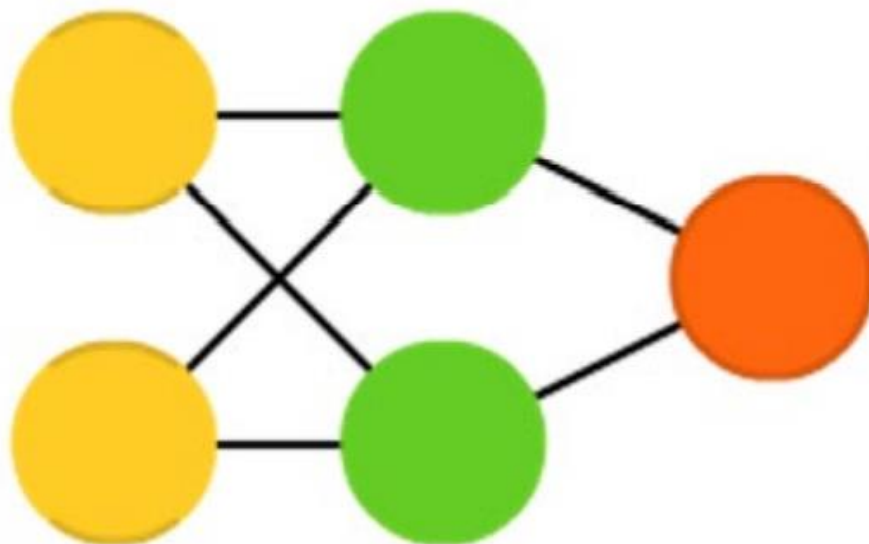


Рисунок 2.4 – Нейронна мережа прямого поширення

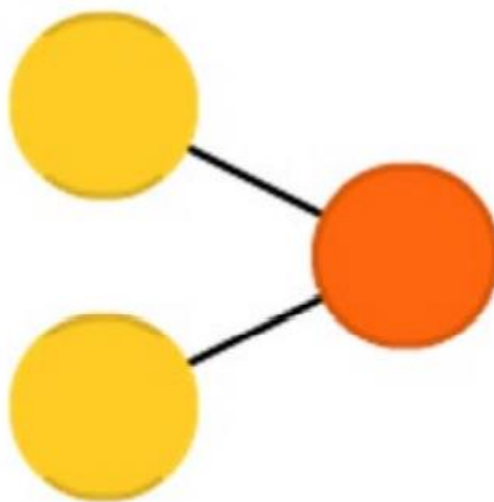


Рисунок 2.5 – Перцептрон

2.5 Нейронна мережа Гопфілда

Нейронна мережа Гопфілда – повнозв’язна мережа. Кожен нейрон з’єднаний один з одним.

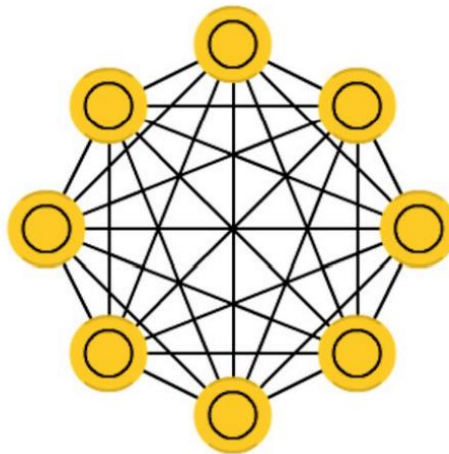


Рисунок 2.6 – НМ Гопфілда

У процесі роботи динаміка таких мереж сходиться (конвергує) одного з положень рівноваги. Ці положення рівноваги визначаються заздалегідь у процесі навчання, є локальними мінімумами функціоналу, так звана енергія мережі (у найпростішому разі — локальними мінімумами негативно певної квадратичної форми на n -мірному кубі). Така мережа може бути використана як асоціативна пам'ять, як фільтр, а також для вирішення деяких завдань оптимізації. На відміну від багатьох нейронних мереж, що працюють до отримання відповіді через певну кількість тактів, мережі Гопфілда працюють до досягнення рівноваги, коли наступний стан мережі точно дорівнює попередньому: початковий стан є вхідним чином, а при рівновазі отримують вихідний образ[23].

Нейронна мережа Гопфілда влаштована так, що її відгук на запам'ятовані еталонні образи складають самі ці образи, а якщо образ трохи спотворити і подати на вхід, він буде відновлений і у вигляді відповіді буде отримано оригінальний образ. Таким чином, мережа Гопфілда здійснює корекцію помилок та перешкод. Як приклад, коли людина бачить частину картинки, вона подумки може її завершити, так і нейронна мережа – отримуючи неповну інформацію вона її добудує.

2.6 Ланцюги Маркова

Ланцюги Маркова – предок машини Больцмана чи мережі Гопфілда. У них задається ймовірність переходу з поточного стану в сусідній. Також, ланцюги не запам'ятовують попередній результат, і не можуть вистроювати результат на основі попередніх спроб.

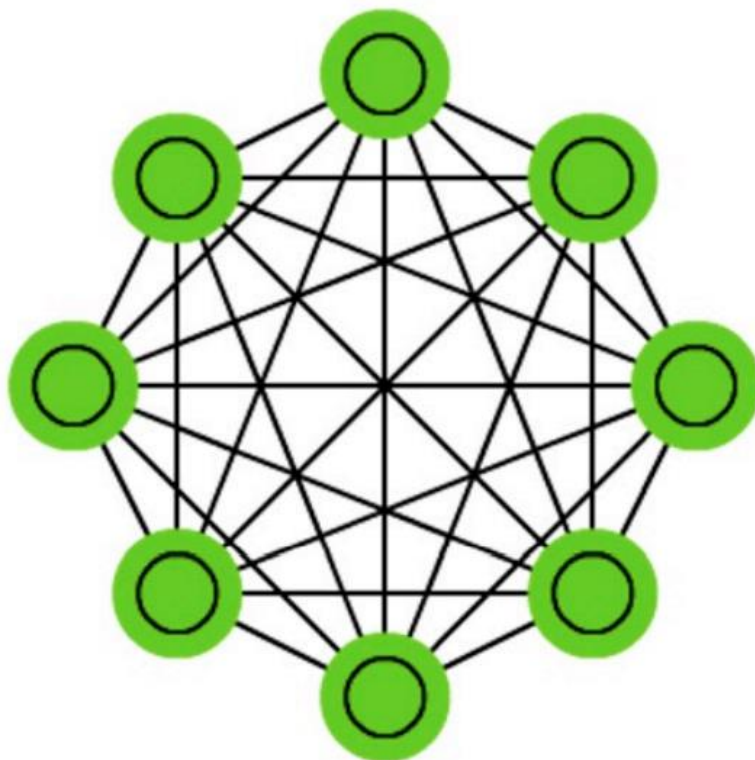


Рисунок 2.7 – Ланцюги Маркова

2.7 Машина Больцмана

Машини Больцмана є дуже схожими на мережі Гопфілда, але в них деякі нейрони є вхідними, а деякі помічаються прихованими. Після оновлення стану всіма нейронами, вхідні нейрони стають протилежними – вихідними. На початку вагові коефіцієнти рандомно набувають деякі значення. Після цього відбувається процес навчання методом зворотного поширення похибки.. Машина Больцмана – стохастична нейронна мережа, так як в навчанні задіяно ланцюг Маркова. Процес

навчання та роботи тут майже такий же, як і в мережі Гопфілда: нейронам задають певні початкові стани, а потім ланцюг починає вільно функціонувати. В процесі роботи нейрони можуть приймати будь-який стан, і ми постійно переміщаємося між вхідними і прихованими нейронами. Активація регулюється значенням загальної температури, при зниженні якої скорочується і енергія нейронів. Скорочення енергії викликає стабілізацію нейронів. Таким чином, якщо температура задана вірно, система досягає рівноваги[24].

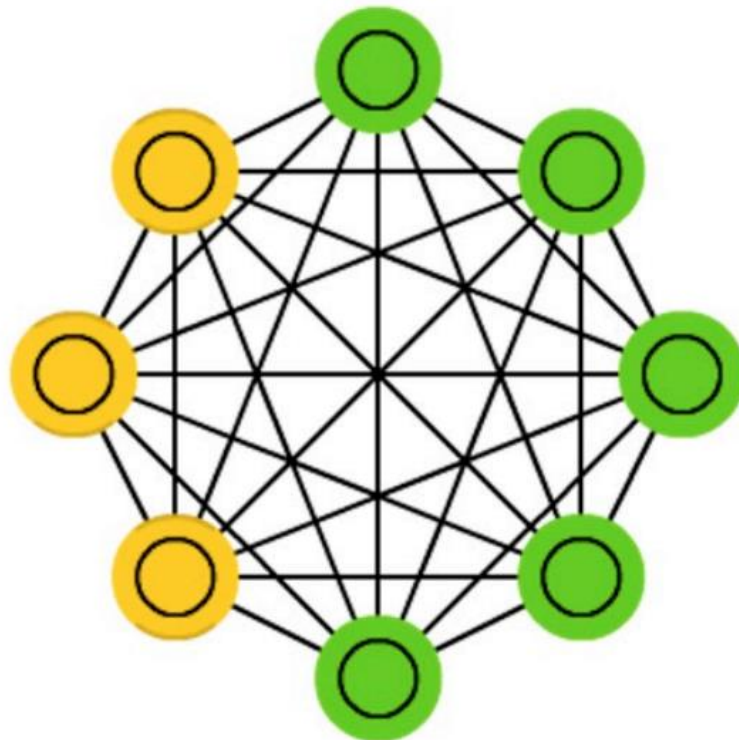


Рисунок 2.8 – Машина Больцмана

2.8 Згорткові нейронні мережі

Згорткові нейронні мережі – використовуюються для обробки зображень, в рідких випадках для аудіо- та інших типів вхідних даних. Типовий спосіб застосування даної мережі – класифікація зображень.

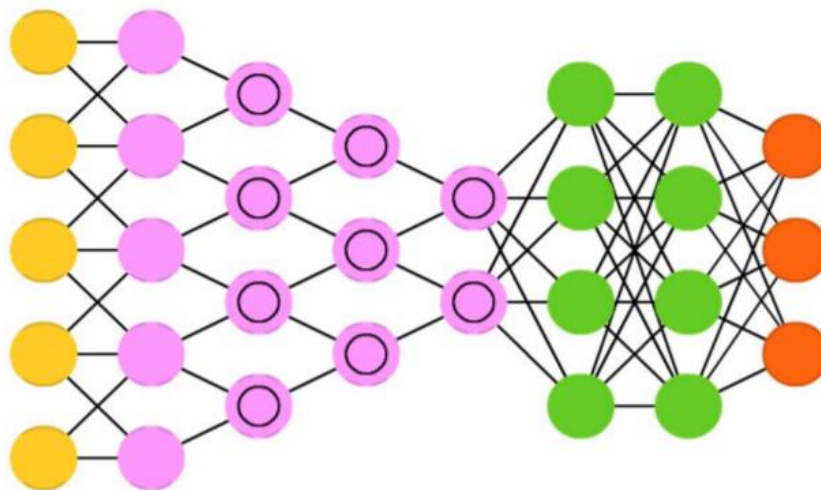


Рисунок 2.9 – Згорткова нейронна мережа

При подачі зображення мережа не будує шар з кількості вузлів, що відповідає кількості пікселів даного зображення. Замість цього мережа обраховує в 100 разів менший квадрат, починаючи з лівого верхнього куту зі здвигом в 1 піксель. Потім, ці данні передаються через згорткові шари, в яких вузли з'єднанні тільки з сусідніми. Шари, в свою чергу, мають властивість стискуватись з ростом глибини, причому вони зменшуються на який-небудь дільник кількості вхідних даних (20 вузлів в наступному шарі перетворяться в 10, в наступному – 5) [25]. Вагомим аспектом є те, що ми не розбиваємо зображення на сектори, а пересуваємось по ньому зі здвигом.

2.9 Нейронна мережа для розпізнавання тварин

Отже, провівши аналіз існуючих типів НМ, для системи що буде спеціалізовуватись на розпізнаванні тварин з фото, оптимальним варіантом буде згорткова нейронна мережа – мережа, що спеціалізується на розпізнаванні об'єктів на фото.

2.10 Алгоритм роботи ЗНМ

У звичайному перцептроні, який є повнозв'язковою нейронною мережею, кожен нейрон пов'язаний з усіма нейронами попереднього шару, причому кожен зв'язок має свій персональний ваговий коефіцієнт. У згортковій нейронній мережі в операції згортки використовується лише обмежена матриця ваг невеликого розміру, яку «рухають» по всьому шару (на самому початку — безпосередньо по вхідному зображенню), формуючи після кожного зсуву сигнал активації для нейрона наступного шару з аналогічною позицією. Тобто для різних нейронів вихідного шару використовуються та сама матриця ваг, яку також називають ядром згортки. Її інтерпретують як графічне кодування будь-якої ознаки, наприклад, наявність похилої лінії під певним кутом. Тоді наступний шар, що вийшов в результаті операції згортки такою матрицею ваг, показує наявність даної ознаки в шарі, що обробляється, і її координати, формуючи так звану карту ознак (англ. feature map).

У згортковій нейронній мережі набір ваг не один, а ціла гама, що кодує елементи зображення (наприклад, лінії та дуги під різними кутами). При цьому такі ядра згортки не закладаються дослідником заздалегідь, а формуються самостійно шляхом навчання мережі класичним методом зворотного поширення помилки. Прохід кожним набором терезів формує свій власний екземпляр карти ознак, роблячи нейронну мережу багатоканальною (багато незалежних карт ознак на одному шарі). Також слід зазначити, що при переборі шару матрицею вагів її пересувають зазвичай не на повний крок (розмір цієї матриці), а на невелику відстань. Так, наприклад, при розмірності матриці ваги 5×5 її зрушують на один або два нейрони (пікселі) замість п'яти, щоб не «переступити» шукану ознаку.

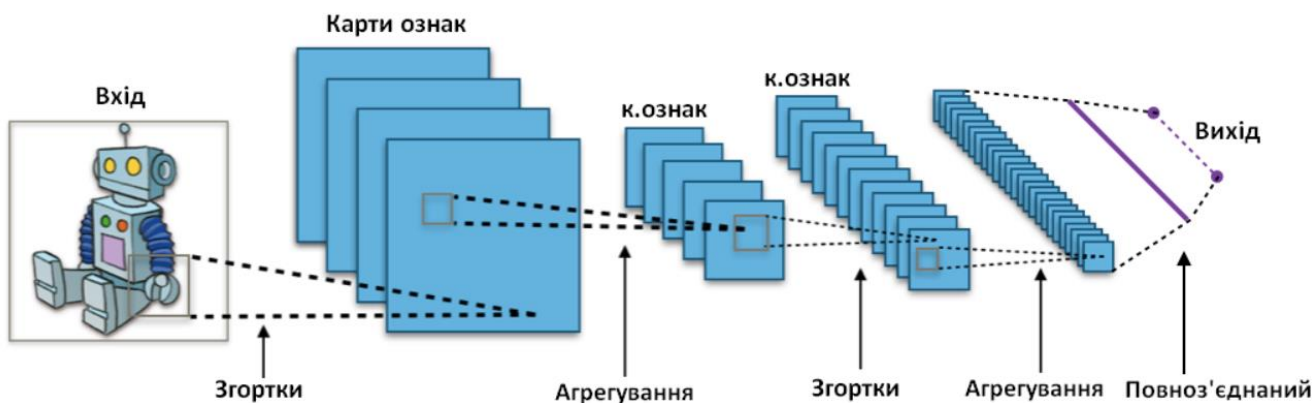


Рисунок 2.10 – Архітектура згорткової нейронної мережі

Розглянемо алгоритм роботи згорткової нейронної мережі:

1. Імпортуємо зображення розміром 32x32 пікселя.
2. Вхідні данні мають вигляд $[32 \times 32 \times 3]$, де 32 – це ширина та висота, 3 – R, G, B канали
3. Згортковий шар перемножує вихідні значення пікселів на значення фільтру, а добутки – сумуються. Відбувається процес згортки.

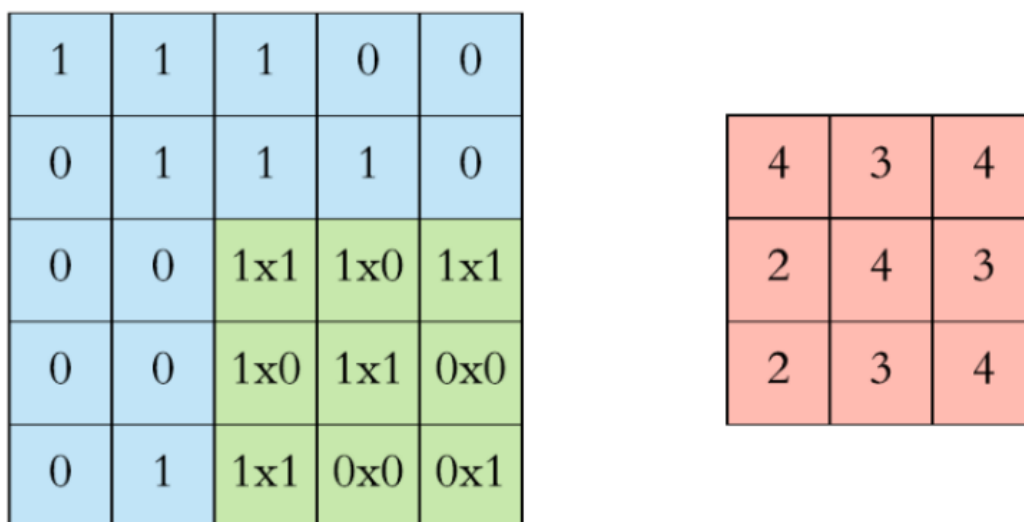


Рисунок 2.11 – Процес згортки

4. Пул-шар зменшує розмірність фото до значень $[16 \times 16 \times 12]$. Ідея полягає в тому, що при виявленні деяких ознак на попередньому кроці образ буде ущільнюватись до менш детальної картинку.

5. Повнозв'язний шар виводить N -мірний вектор для визначення класу шляхом звернення до попереднього класу.

2.11 Види навчання згорткових нейронних мереж

Найпростіший та найпопулярніший спосіб навчити нейронну мережу – це використання методу з учителем (метод зворотнього розповсюдження помилки). Це – ітеративний градієнтний алгоритм для мінімізації помилок роботи і отримання бажаного результату. Ідея полягає в розповсюдженні сигналів похибки від виходів до входів, в зворотньому напрямку розповсюдження сигналів в звичайному режимі роботи. В якості функції активації найчастіше використовують декілька видів сигмоїд:

- Раціональна сигмоїда (2.3):

$$f(s) = \frac{s}{|s| + \alpha} \quad (2.3)$$

При $\alpha = 0$ функція стає так званою пороговою функцією активації.

- Функція Фермі або експоненційна сигмоїда (2.4):

$$f(s) = \frac{1}{1 + e^{-2\alpha s}} \quad (2.4)$$

- Гіперболічний тангенс (2.5):

$$f(s) = \text{th} \frac{s}{\alpha} = \frac{e^{\frac{s}{\alpha}} - e^{-\frac{s}{\alpha}}}{e^{\frac{s}{\alpha}} + e^{-\frac{s}{\alpha}}} \quad (2.5)$$

де s – вихід суматору нейрону, α – довільна константа.

Раціональна сигмоїда така популярна, бо потребує найменшу кількість процесового часу для обчислення. Обчислення гіперболічного тангенсу потребує найбільшу кількість тактів роботи процесору.

В тих випадках, коли можливо оцінити роботу мережі, навчання нейронних мереж можна трактувати як задачу оптимізації. Для цього будується функція оцінки. Вона, як правило, явно залежить від вихідних сигналів і неявно від всіх її параметрів. Найрозповсюдженіший приклад оцінки мережі – сума квадратів відстаней від вихідних сигналів до бажаних значень (2.4):

$$H = \frac{1}{2} \sum_{\tau \in v_{out}} (Z(\tau) - Z^*(\tau))^2 \quad (2.4)$$

де $Z^*(\tau)$ – бажані значення вихідного сигналу.

Інший різновид методів навчання – без учителя. Автоасоціатор є одним з таких методів. Це — спеціальна архітектура штучних нейронних мереж, що дозволяє застосовувати навчання без учителя під час використання методу зворотного поширення помилки. Найпростіша архітектура автоасоціатора – мережа прямого поширення, без зворотних зв'язків, найбільш схожа з перцептроном і містить вхідний шар, проміжний шар та вихідний шар. На відміну від перцептрону, вихідний шар автокодувальника повинен містити стільки ж нейронів, як і вхідний шар. Основний принцип роботи та навчання мережі автоасоціатора – отримати на вихідному шарі відгук, найближчий до вхідного. Щоб рішення не виявилось тривіальним, на проміжний шар автоасоціатора накладають обмеження: проміжний шар повинен бути меншою розмірності, ніж вхідний і вихідний шари, або штучно обмежується кількість одночасно активних нейронів проміжного шару – розріджена активація. Ці обмеження змушують нейромережу шукати узагальнення і кореляцію в даних, що надходять на вхід, виконувати їх стиснення. Таким чином, нейромережа автоматично навчається виділяти із вхідних даних загальні ознаки, які кодуються у значеннях ваг штучної нейронної мережі. Так, при навчанні мережі на наборі різних вхідних зображень, нейромережа може самостійно навчитися розпізнавати лінії та смуги під різними кутами [26].

Інший метод навчання – метод k -середніх. Це один з найпопулярніших методів кластеризації. Суть алгоритма полягає в мінімізації сумарного квадратичного відхилення точок кластерів від їхніх центрів (2.5).

$$V = \sum_{i=1}^k \sum_{x \in s_i} (x - \mu_i)^2 \quad (2.5)$$

Тут k – число кластерів, s_i – результуючі кластери, μ_i – центр мас всіх векторів x з кластеру s_i .

Принцип роботи алгоритму полягає в розбитті множин елементів векторного простору на завчасно відому кількість кластерів. Основна ідея полягає в тому, що на кожній ітерації перераховується центр мас для кожного кластеру, отриманого на

попередньому кроці. Після цього вектори знову розбиваються на кластери відповідно до того, наскільки якийсь із нових центрів став ближче за вибраною метрикою. Алгоритм завершає роботу тоді, коли на ітерації не відбувається змін відстаней в середині кластера. Нижче наведена демонстрація алгоритму у двовимірному варіанті.

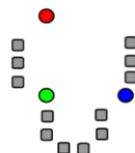


Рисунок 2. 12 – Вихідні та випадково вибрані початкові точки

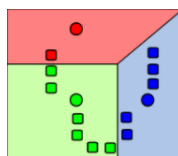


Рисунок 2.13 – Точки, рознесені до початкових центрів

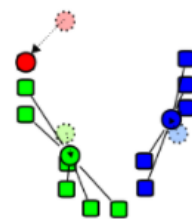


Рисунок 2.14 – Обчислення нових центрів кластерів

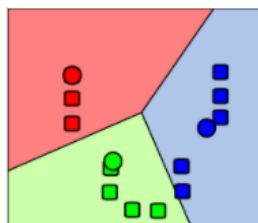


Рисунок 2.15 – Результуючий вигляд

Попередні кроки, окрім першого, повторюються до тих пір, поки алгоритм не зійдеться.

2.12 Архітектура для згорткової нейронної мережі

Розглянемо декілька типів архітектур для вибору оптимальної для майбутнього додатку.

2.12.1 LeNet

LeNet – одна з найперших згорткових нейронних мереж, розроблена Ян Ле Куном (Yann LeCun) в 1994-у році.

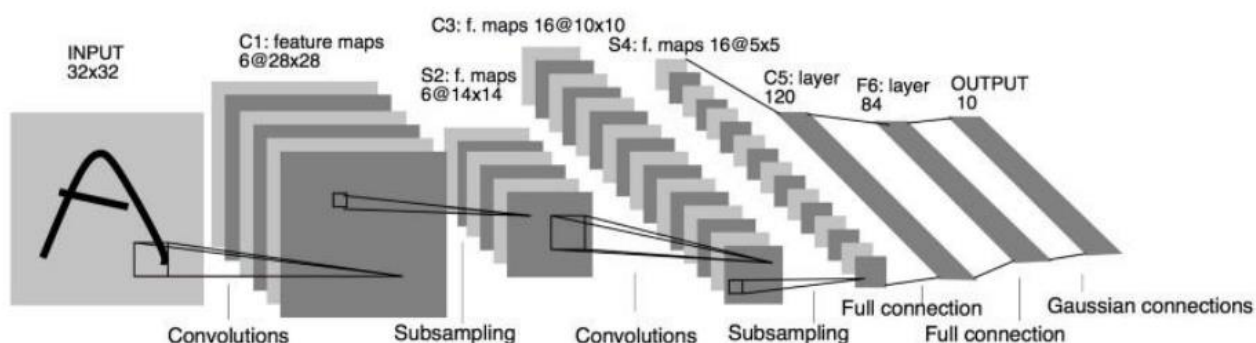


Рисунок 2.16 – Архітектура LeNet

Архітектура LeNet стала фундаментальною для глибокого навчання, особливо з точки зору розподілу властивостей зображення по всій картинці. Згортки з параметрами, що навчаються, дозволяли з допомогою декількох параметрів ефективно виокремлювати однакові властивості з різних місць. Основною відмінністю і перевагою архітектури виявилась можливість зберігати параметри і результати обчислень, на відміну від використання кожного пікселя в якості окремих вхідних даних для великою багатшарової мережі.

2.12.2 AlexNet

У 2012-у році Алекс Крижевський опублікував AlexNet – поглиблену та більш розширену версію LeNet, яка з великим відривом перемогла в важкому змаганні ImageNet.

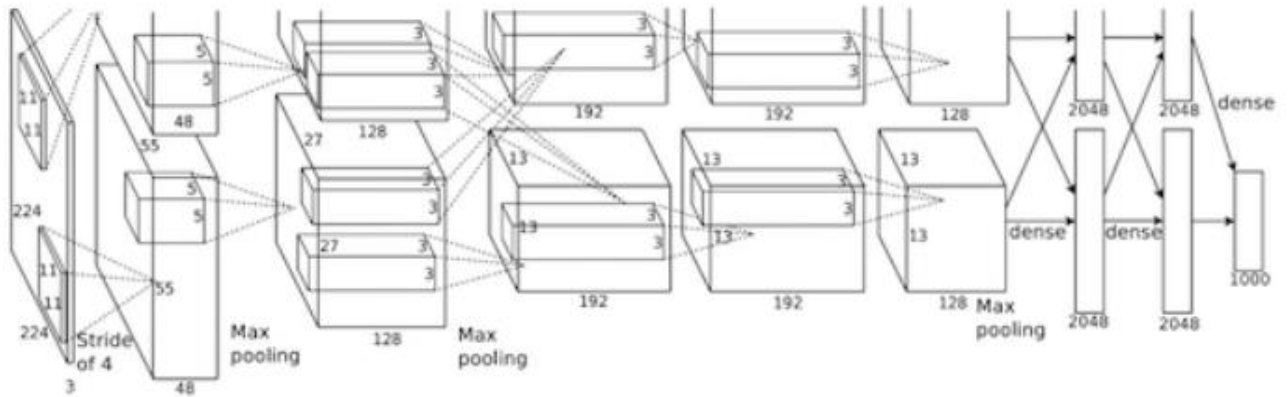


Рисунок 2.17 – Архітектура AlexNet

В AlexNet результати обчислень масштабовані в більшу нейронну мережу, яка здатна вивчити більш складні об'єкти і їхні ієрархії. Особливості цього рішення:

- Використання блоків лінійної ректифікації (ReLU) в якості нелінійностей;
- Використання методики відкидування для вибіркового ігнорування окремих нейронів по ходу навчання;

Успіх AlexNet запустив революцію. Згорткові нейронні мережі стали досить корисними в різних сферах діяльності.

2.12.3 VGG

Мережу VGG розробили в Оксфорді. Їхньою особливістю було те, що в кожному згортковому шарі вперше застосували фільтри 3x3, і об'єднали ці шари в послідовності згортки.

Це протирічить закладеним в LeNet принципам, згідно з якими великі згортки використовувались для виокремлення однакових властивостей зображення. Замість фільтрів 9x9 та 11x11, що застосовувались в AlexNet, використовувались більш дрібні, небезпечно близькі до згортки 1x1. Перевагою VGG стало відкриття,

що декілька послідовних згорток 3×3 можуть емулювати крупніші згортки, наприклад 5×5 або 7×7 .

2.12.4 Network-in-network

В основі архітектури NIN лежить проста ідея: використання згорток 1×1 для покращення властивостей комбінаторності в згорткових шарах.

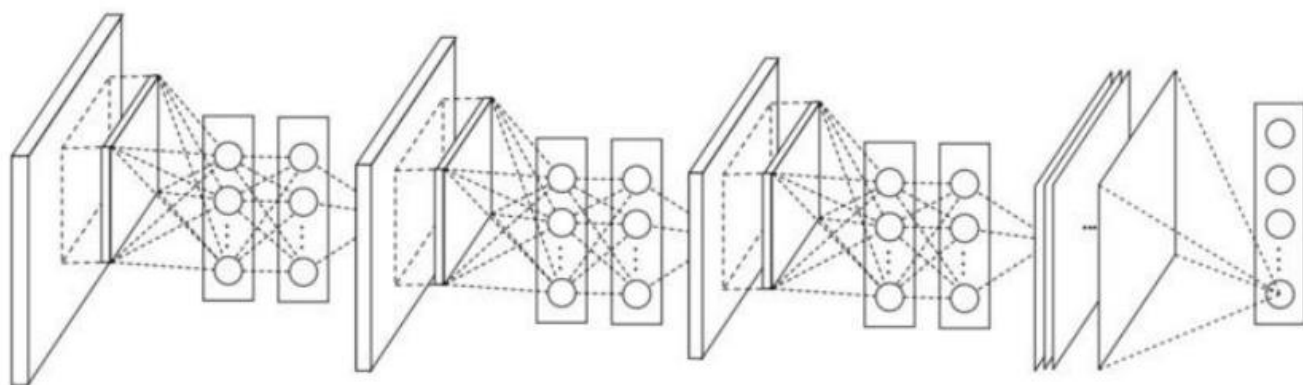


Рисунок 2.18 – Архітектура NIN

В NIN після кожної згортки застосовують просторові MLP-шари, аби краще комбінувати властивості перед подачою в наступний шар. MLP дозволяє досить сильно покращити ефективність окремих згорткових шарів шляхом їхніх комбінацій в більш складні групи. Ця ідея пізніше була використана в інших архітектурах, таких як ResNet, Inception і їхніх варіаціях.

2.12.5 ResNet

Дана мережа була опублікована в грудні 2015-го року. В ній закладені прості ідеї: подача вихідних даних двох успішних згорткових шарів і обхід вхідних даних для наступного шару.

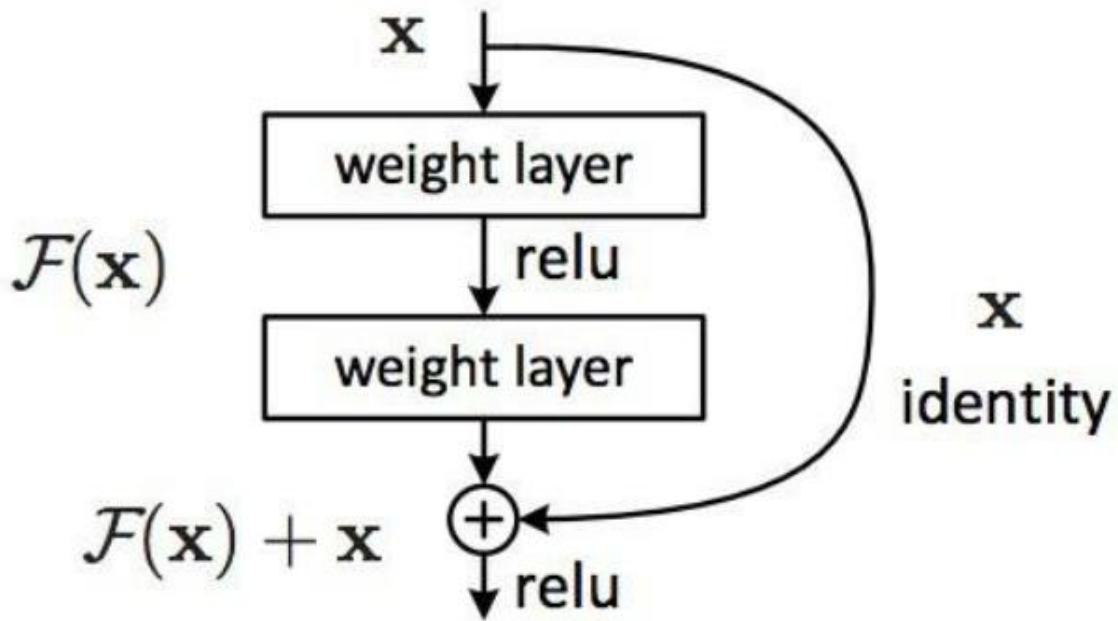


Рисунок 2.19 – Архітектура ResNet

Дану архітектуру можна розглядати як систему одночасно паралельних і послідовних модулів: в багатьох модулях in\out сигнал приходить паралельно, а вихідні сигнали кожного модуля з'єднуються послідовно.

2.12.6 Вибір архітектури для ЗНМ

Для нейронної мережі, що буде інтегрована в майбутній додаток, прийнято рішення використовувати архітектура ResNet. На рисунку 2.20, що наведено нижче, видно, що дана архітектура має дуже високий показник точності відносно кількості операцій, необхідних для одного прямого проходження.

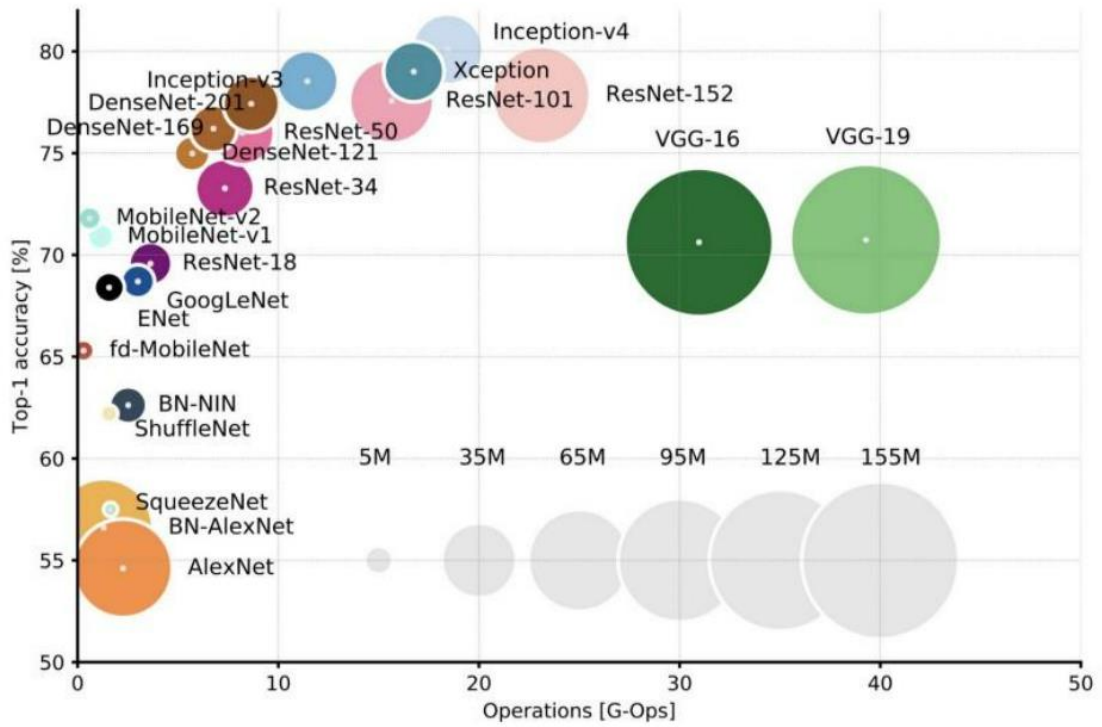


Рисунок 2.20 – Результат випробування різних архітектур НМ.

Щодо результатів виміру питомої точності архітектур – ResNet з 18 шарів входить в першу п’ятірку нейронних мереж.

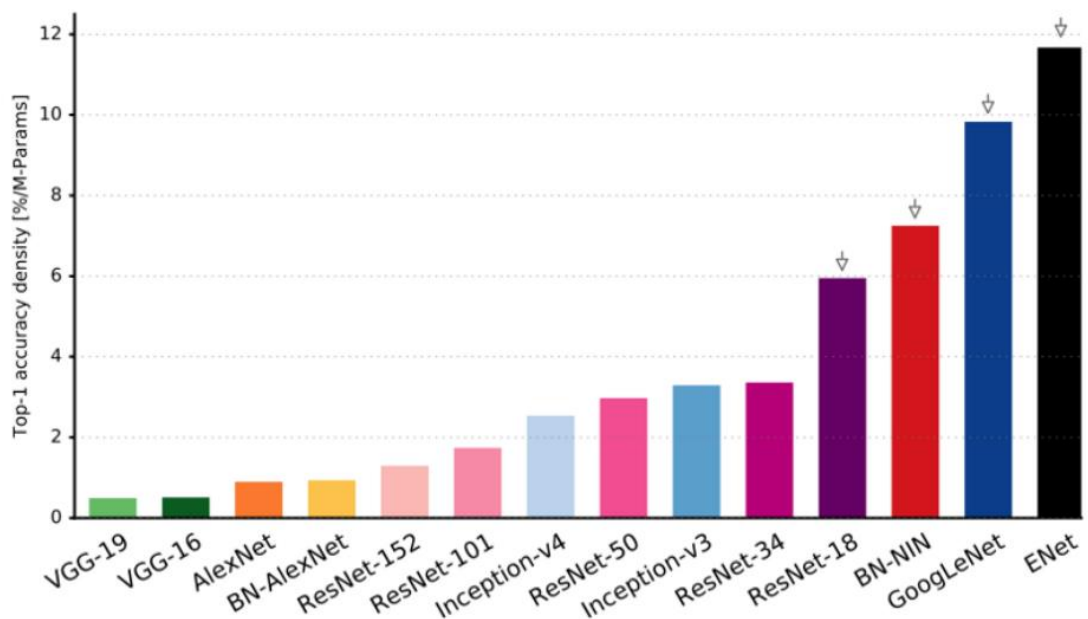


Рисунок 2.21 – Результат виміру питомої точності архітектур.

Аби впевнитись, що вибір архітектури був зроблений правильно, порівняємо результат їхнього навчання. Для цього було обрано LeNet, AlexNet, NiN та ResNet архітектури нейронних мереж. Слід зазначити, що всі мережі навчались на одному комп'ютері за допомогою однакових наборів даних (по 200 зображень котів та собак) ресурсами центрального процесору (CPU). У таблиці 2 наведений час навчання кожної архітектури. Він приблизно однаковий і не критично відрізняється

Таблиця 2 – Час навчання порівняних архітектур

Тип архітектури	Час навчання, с
LeNet	784
AlexNet	759
NiN	732
ResNet	742

Слід зазначити, що тестуючи ці нейромережі на комп'ютері з іншими характеристиками, або іншим набіром даних, або використовуючи для цього графічний адаптер (GPU), результат може відрізнятися.

Щоб протестувати надійність та якість роботи нейронних мереж, змусимо їх розпізнати по 10 зображень кожного виду. Результатом розпізнавання нейронної мережі є вірогідність, з якою той чи інший об'єкт відноситься до заданого класу. В таблиці 3 наведено середню вірогідність, визначену різними нейромережами, з якою вони класифікували зображення котів та собак.

Таблиця 3 – результат випробувань нейронних мереж

Тип архітектури	Середнє значення % розпізнавання зображень котів	Середнє значення % розпізнавання зображень собак
LeNet	96.8	91.3
AlexNet	97.6	95.4
NiN	98.7	94
ResNet	98.3	96.1

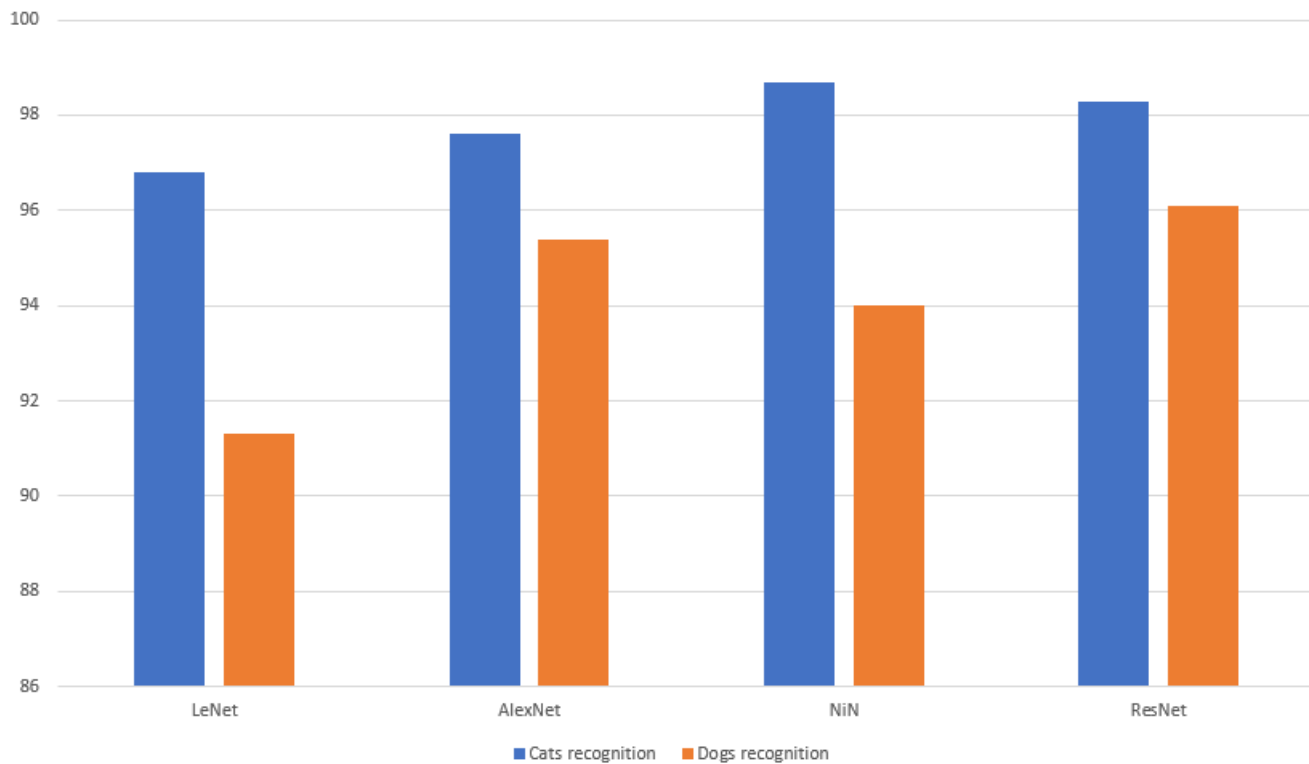


Рисунок 2.22 – результат випробувань нейронних мереж з різними архітектурами

Висновки до розділу

В даному розділі було проведено оглядовий аналіз декількох видів нейронних мереж для її оптимального вибору в майбутній застосунок. Для задач ідентифікації об'єктів на зображеннях використовують згорткові нейронні мережі, оскільки їхньою головною особливістю є 3-вимірна розмірність нейронів, що зменшує загальну кількість зв'язків між нейронами. Завдяки цьому ця мережа спеціалізується на роботі з такими ієрархічними даними, як зображення. Саме згорткову нейронну мережу було обрано для майбутнього застосунку.

Було проаналізовано декілька найпопулярніших архітектур для згорткової нейронної мережі. Прийнято рішення використовувати ResNet архітектуру для нейронної мережі, що буде інтегрована в майбутній додаток. Вона показала один з найкращих показників часу навчання серед обраних, та найвищі результати розпізнавання. Способом навчання згорткової нейронної мережі обрано метод зворотнього поширення похибки.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Вибір платформи та архітектури

Додаток буде складатися з трьох логічних частин: бази даних, серверної частини додатку та клієнтської частини.

3.1.1 Частина бази даних

Для додатку, в ролі сховища даних буде виступати ядро MS SQL Server. На ньому буде побудовану реляційну базу даних застосунку. MS SQL Server має ряд переваг:

- масштабованість;
- стійкість до відмов;
- висока степінь безпеки;
- висока продуктивність;

Крім цього, в даній СУБД наявна повна незалежність даних, простота в розробці БД, опрацюванню та отриманню даних.

3.1.2 Серверна частина

Майбутній додаток слід наділити наступними властивостями:

- кросплатформенність та незалежність від ОС, де буде працювати застосунок;
- гнучкість системи;
- простота в подальшому масштабуванні системи;
- простота в розширенні функціоналу;
- простота в пошуку проблемних місць системи.

В ролі серверної частини буде виступати веб-додаток, побудований на технологіях .NET, а саме:

- платформа – .NET 5;
- мова програмування – C#;
- доступ до бази даних – ORM система Entity Framework.

Платформа .NET 5 – модульна платформа від компанії Microsoft. Вона є кросплатформенною і сумісною з операційними системами Windows, Linux та macOS. Підтримує безліч мов програмування, в тому числі C#. Є логічним розвитком платформи .NET Framework, який позбавився її недоліків і здобув безліч нового функціоналу для ефективної розробки.

Мова програмування C# – одна з багатьох мов, які підтримує платформа .NET та є вінцем творіння Microsoft. Цей інструмент справляється з широким спектром задач, а саме:

- веб-розробка;
- розробка мобільних додатків;
- розробка настільних додатків;
- розробка ігор і т.д.

ORM система Entity Framework – продовженням технології Microsoft ActiveX Data та надає можливість роботи з базами даних через об'єктно-орієнтований код C#. Цей підхід надає ряд істотних переваг: не потрібно турбуватися про код доступу до даних, не потрібно знати деталей роботи СУБД SQL Server та синтаксису мови запитів T-SQL, натомість взаємодія з таблицями бази даних як із класами C#, з полями цих таблиць - як із властивостями класів, а синтаксис SQL-запитів, що у ADO.NET раніше потрібно було вставляти в код C# як команди, замінено більш зручний підхід з LINQ. Entity Framework бере на себе обов'язки перетворення коду C# в SQL-інструкції. При роботі з Entity Framework присутні величезні можливості створення моделі бази даних за допомогою інтегрованого середовища розробки (IDE) Visual Studio. Починаючи з версії Entity Framework 4.1, існує три підходи з проектування бази даних:

- Database-First – підходить для проектувальників баз даних. Спочатку створюється база даних за допомогою різних інструментів (наприклад, SQL Server Management Studio), а потім генерується EDMX-модель бази даних (зручний графічний інтерфейс для взаємодії з базою даних у вигляді діаграм та об'єктну модель у вигляді класів C#). В даному випадку потріб працювати з SQL Server і добре знати синтаксис T-SQL, але не потрібно розбиратися в C#.

- Model-First – підходить для архітекторів. Спочатку створюється графічна модель EDMX Visual Studio (у фоновому режимі створюються класи C# моделі), а потім генерується на основі діаграми EDMX база даних. При цьому не потрібно знати ні деталей T-SQL ні синтаксису C#.

- Code-First – підходить для програмістів. При даному підході модель EDMX взагалі не використовується і вручну налаштовуються класи C# об'єктної моделі (цей підхід підтримує як генерацію сутнісних класів з існуючої бази даних, так і створення бази даних із створеної вручну моделі об'єктів C#)

Майбутній додаток буде побудований на основі підходу code-first.

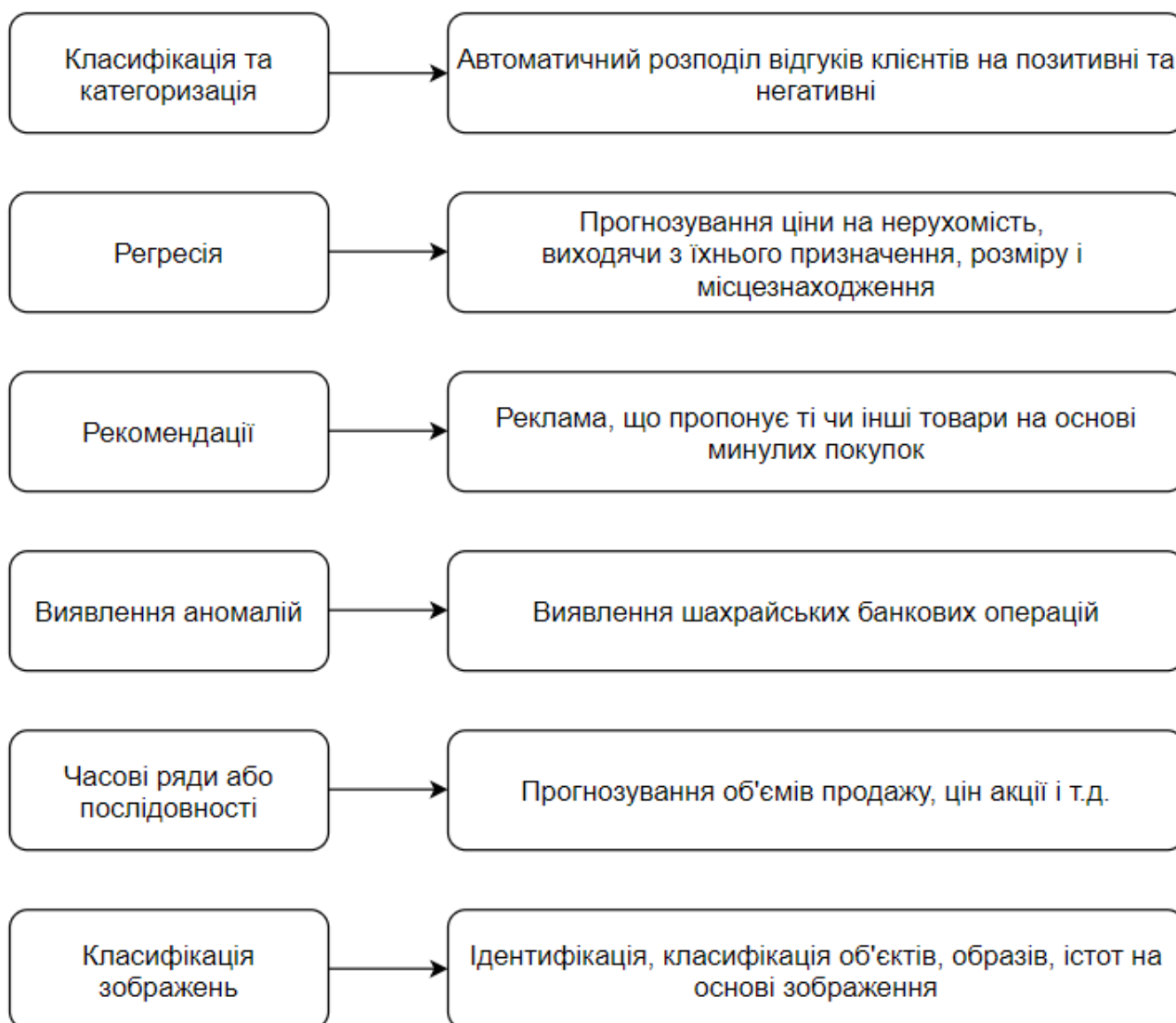


Рисунок 3.1 – Типи підтримуваного прогнозування в ML.NET

3.2 Реалізація серверної частини додатку

Серверна частина додатку буде реалізована на 3-рівневій монолітній архітектурі.

1. Data access layer (DAL) – рівень доступу до даних. На ньому будуть реалізовані фундамент застосунку та доступ до бази даних: основні CRUD операції (create, read, update, delete).

2. Business logic layer (BLL) – рівень бізнес логіки. На цьому рівні буде реалізований весь функціонал додатку. На цьому рівні буде інтегрована

бібліотека ML.NET та побудовані функції для обробки зображень, виокремлення інформації та подальшої її маніпуляції в угоду цілям користувача.

3. WEB Api – найвищий рівень серверної частини додатку. В ньому буде список Api (Get, Post, Put, Delete) для обробки web-запитів користувача.

Щоб навчити нейронну мережу розрізняти тварин, їй необхідно надати великий масив вхідних даних. Для цього слід завантажити зображення тварин, і чим більше – тим краще.



Рисунок 3.2 – Масив даних з зображенням собак



Рисунок 3.3 – Масив даних з зображенням котів

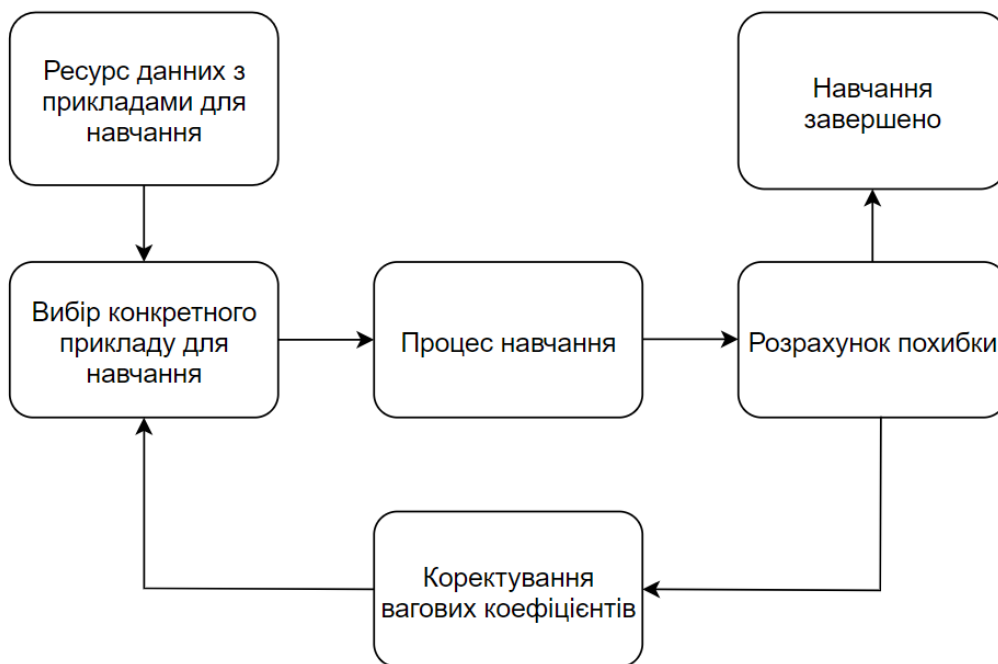


Рисунок 3.4 – графічне зображення процесу навчання мережі

Після процесу автоматичного навчання можемо перевірити модель на працездатність.



Рисунок 3.5 – Перший результат перевірки моделі



Рисунок 3.6 – Другий результат перевірки моделі



Рисунок 3.7 – Результат роботи програми

В першому випадку система з вірогідністю 99% визначила кота. Це пов'язано з тим, що всі зображення, на яких навчалась модель, були гарної якості – так само як і зображення з яким проводилось тестування системи. В другому випадку результат був 97% – трохи нижчим аніж в першому, але досі високим.

ВИСНОВКИ

У ході виконання магістерської роботи було запропоноване рішення на основі інтеграції нейронних мереж, що значно розширює функціонал системи розумного будинку.

В першому розділі «Аналіз предметної області» було розглянуто історію розвитку нейронних мереж. Наведені основні події, які давали потужний поштовх до подальшого вдосконалення алгоритмів та створення нових. Були розкриті поняття штучного та біологічного нейронну, поняття штучної нейронної мережі. Також, було проведено порівняльний аналіз додатків зі здатність розпізнавати об'єкти на фото, а саме: Google Lens, CamFind, Image Recognizer. Вони чудово справляються зі своєю задачею, але мають суттєві недоліки: відсутність списку відкритих АРІ та залежність від платформи.

В другому розділі «Аналіз інформаційного забезпечення» було більш детально розглянуто та порівняно деякі види нейронних мереж. Також, було розглянуто різні архітектури, які використовує ЗНМ, та різні види навчання нейронних мереж. Для майбутнього додатку було обрано згорткову нейронну мережу, побудовану на архітектурі ResNet, так як вона спеціалізується на обробці зображень та виділення з них інформації.

В третьому розділі «Розробка програмного забезпечення» описані архітектура, основні технології та процес створення застосунку. Вибір платформи впав на .NET 5, так як вона надає кросплатформеність. Мовою програмування обрана С#. В ролі сховища даних виступає MS SQL Server. Використано бібліотеку ML.NET, щоб навчити застосунок розпізнавати тварин на фото.

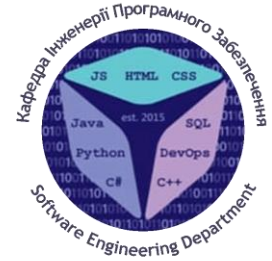
ПЕРЕЛІК ПОСИЛАНЬ

1. Ethem Alpaydin (2020). Introduction to Machine Learning (Fourth ed.). MIT..
2. Н. Винер. Кибернетика. 2-е изд., 1961, гл. I.
3. Шахнов В.А., Власов А.И., Кузнецов А.С. Нейрокомпьютеры: архитектура и схемотехника / М.: Изд-во Машиностроение. 2000.
4. Голубев Ю. Ф. Нейросетевые методы в мехатронике. — М.: Изд-во Моск. унта, 2007.
5. Бонгард М. М. Проблемы узнавания. — М.: Физматгиз, 1967
6. J. J. Hopfield, «Neural networks and physical systems with emergent collective computational abilities», Proceedings of National Academy of Sciences, vol. 79 no. 8 pp. 2554–2558, April 1982
7. Rumelhart D. E., Hinton G. E., Williams R. J. Learning Internal Representations by Error Propagation // Parallel Distributed Processing. Vol. 1. — Cambridge, MA: MIT Press, 1986.
8. <https://ru.bmstu.wiki>
9. <https://www.osp.ru/cw/2001/09/39289>
10. Крисилов В.А. Представление исходных данных в задачах нейросетевого программирования / Одесса: ОНПУ. 2003
11. <https://habr.com/ru/post/312450/>
12. Шитиков В.К., Розенберг Г.С., Зинченко Т.Д. Методы системной идентификации / Тольятти: ИЭВБ РАН. 2003. 463 с
13. Шахнов В.А., Власов А.И., Кузнецов А.С. Нейрокомпьютеры: архитектура и схемотехника / М.: Изд-во Машиностроение. 2000. 64 с.
14. Хайкин С. Нейронные сети. Полный курс / М.: Вильямс, 2006
15. Аверкин А. Н., Гаазе-Рапопорт М. Г., Поспелов Д. А. Толковый словарь по искусственному интеллекту. М.: Радио и связь, 1992. -256с.
16. <https://lonbmt.jimdofree.com/app/download/11324513093/>

17. Рассел С., Норвиг П. Искусственный интеллект: современный подход / Изд-во: Вильямс. 2006. 1424 с.
18. <https://timesofindia.indiatimes.com/gadgets-news/google-lens-app-gets-two-new-features/articleshow/69568473.cms>
19. <https://imagerecognize.com/>
20. Rosenblatt Frank // Principles of Neurodynamics
21. Rosenblatt, Frank. “The perceptron: a probabilistic model for information storage and organization in the brain.” Psychological review 65.6 (1958): 386.
22. Graves, A.; Liwicki, M.; Fernandez, S.; Bertolami, R.; Bunke, H.; Schmidhuber, J. (АНГЛ.)рус.. A Novel Connectionist System for Improved Unconstrained Handwriting Recognition (АНГЛ.) // IEEE Transactions on Pattern Analysis and Machine Intelligence (АНГЛ.)рус. : journal. — 2009. — Vol. 31, no. 5.
23. J. J. Hopfield, «Neural networks and physical systems with emergent collective computational abilities», Proceedings of National Academy of Sciences, vol. 79 no. 8 pp. 2554–2558, April 1982.
24. Hinton, Geoffrey E., and Terrence J. Sejnowski. “Learning and relearning in Boltzmann machines.” Parallel distributed processing: Explorations in the microstructure of cognition 1 (1986): 282-317.
25. LeCun, Yann // Gradient-based learning applied to document recognition
26. Autoencoder for Words, Liou, C.-Y., Cheng, C.-W., Liou, J.-W., and Liou, D.-R., Neurocomputing, Volume 139, 84-96 (2014),



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ



Кафедра інженерії програмного забезпечення

МАГІСТЕРСЬКА РОБОТА
«РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РОЗУМНОГО
БУДИНКУ ДЛЯ РОЗПІЗНАВАННЯ ДОМАШНІХ ТВАРИН ЗА
ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ»

Виконав: студент групи ПДМ-61, Горелов Артем Ігорович

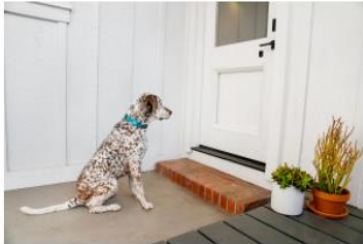
Керівник: к.т.н., доц. кафедри ІІЗ, Трінтіна Наталія Альбертівна

Київ - 2021

Мета роботи: побудова додатку з використанням нейронної мережі з функціоналом розпізнавання об'єктів, образів на зображеннях, для його інтеграції в системи розумного будинку.

Об'єкт дослідження: системи розпізнавання об'єктів, образів на зображеннях, що використовують для цього нейронні мережі.

Предмет дослідження: різні типи нейронних мереж, їхніх архітектур, що надають змогу розпізнавати образи на зображеннях.



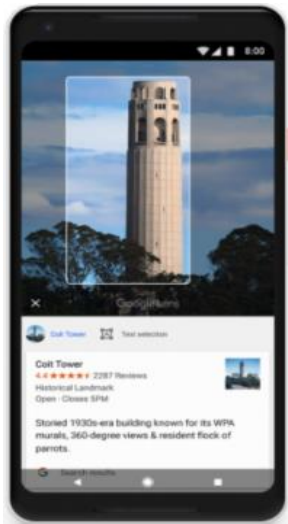
myQ Pet Portal

Boston creative house

Недоліки:

- не є повністю автоматизованими рішеннями;
- обмежений функціонал;
- не є надійними з точки зору безпеки;
- відсутність можливості розширення функціоналу.

Google Lens



CamFind

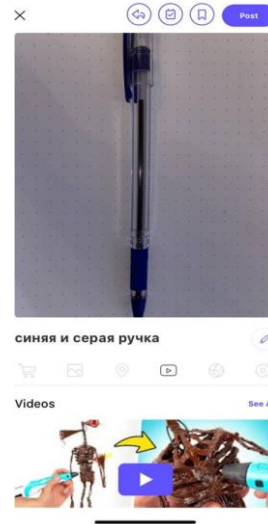
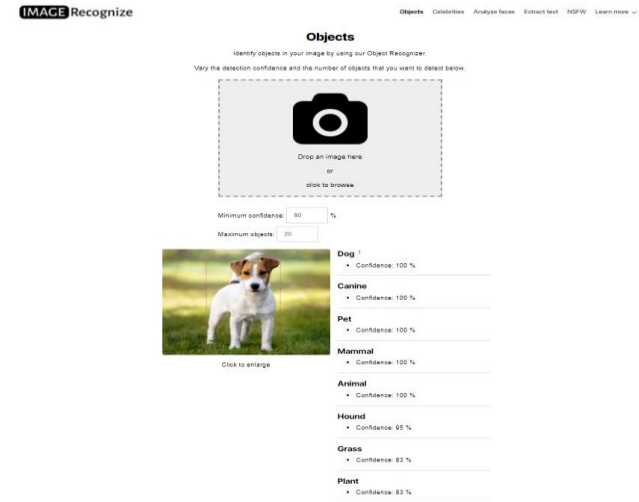
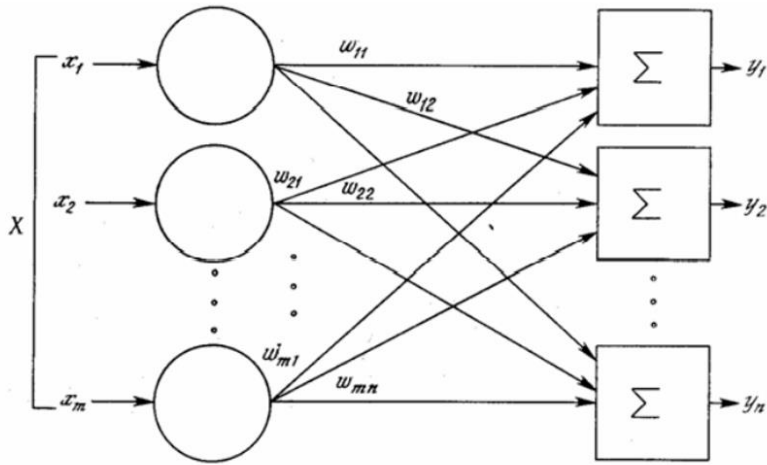


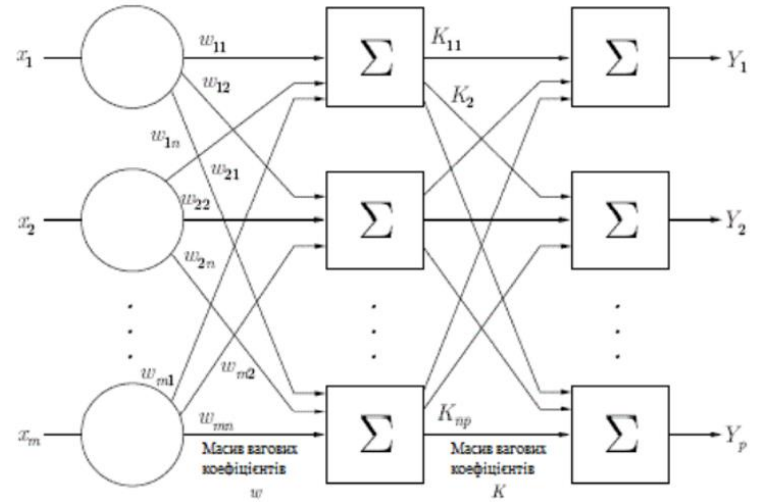
Image Recognize



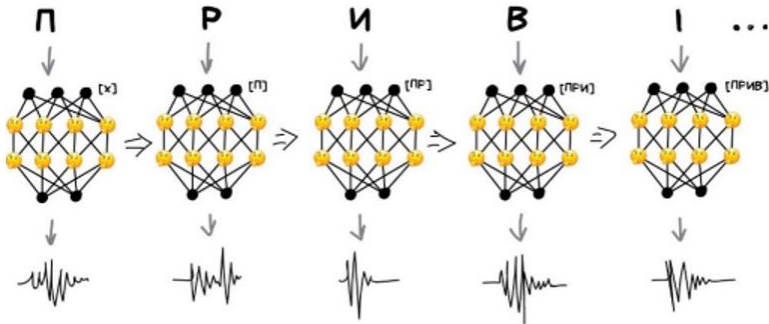
Критерій порівняння	Google Lens	Image Recognizer	CamFind
Функціональність	+	-	-
Зручність використання	+	+	+
Можливість розширення функціоналу	-	-	-
Кросплатформенність	-	+	-
Відкрите API	-	-	-



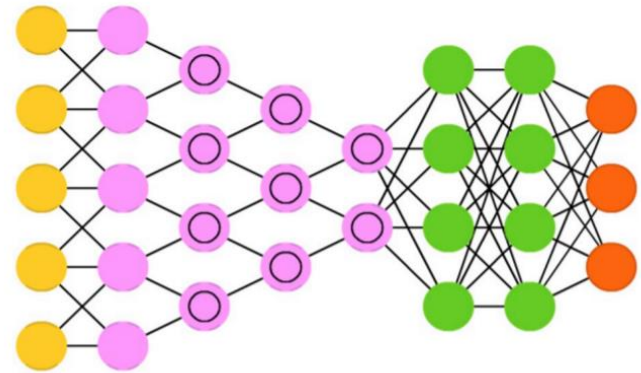
Одношарова нейронна мережа



N-шарова нейронна мережа



Рекурентна нейронна мережа



Згортова нейронна мережа

- Було використано згорткову нейронну мережу.
- Метод навчання – зворотнє розповсюдження похибки:

$$\Delta\omega_{i,j}(n) = \alpha\Delta\omega_{i,j}(n-1) + (1-\alpha)\eta\delta_j o_i \quad (1)$$

$$\omega_{i,j}(n) = \omega_{i,j}(n-1) - \Delta\omega_{i,j}(n), \quad (2)$$

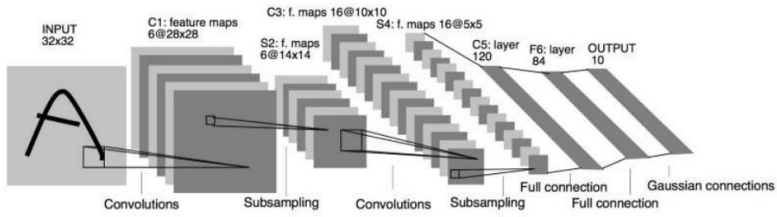
де $\omega_{i,j}$ – вага.

- В якості функції активації обрана сигмоїдальна функція – раціональна сигмоїда:

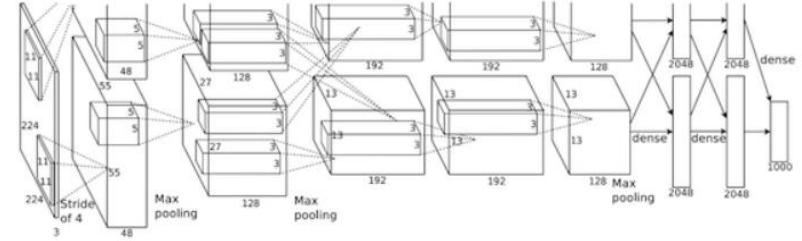
$$f(s) = \frac{s}{|s|+\alpha}, \quad (3)$$

де s – вихід суматору нейрону,

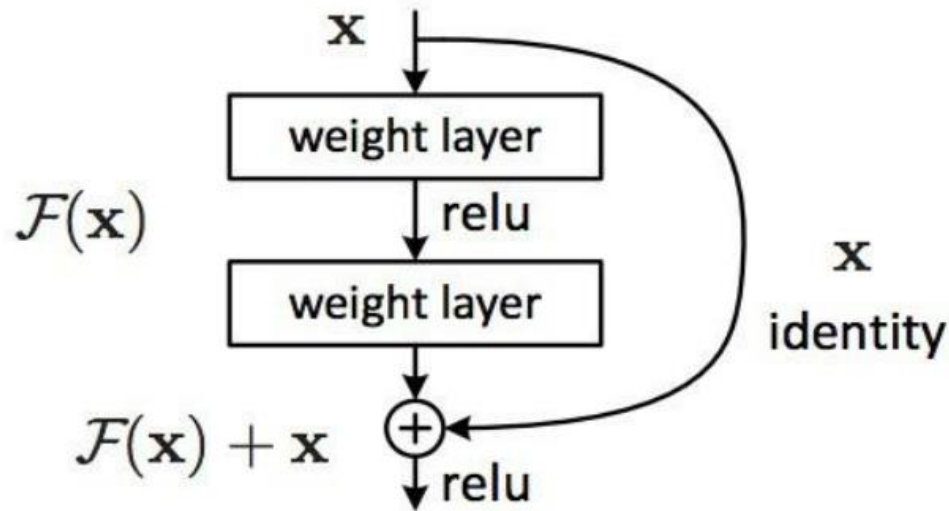
α – довільна константа.



Архітектура LeNet

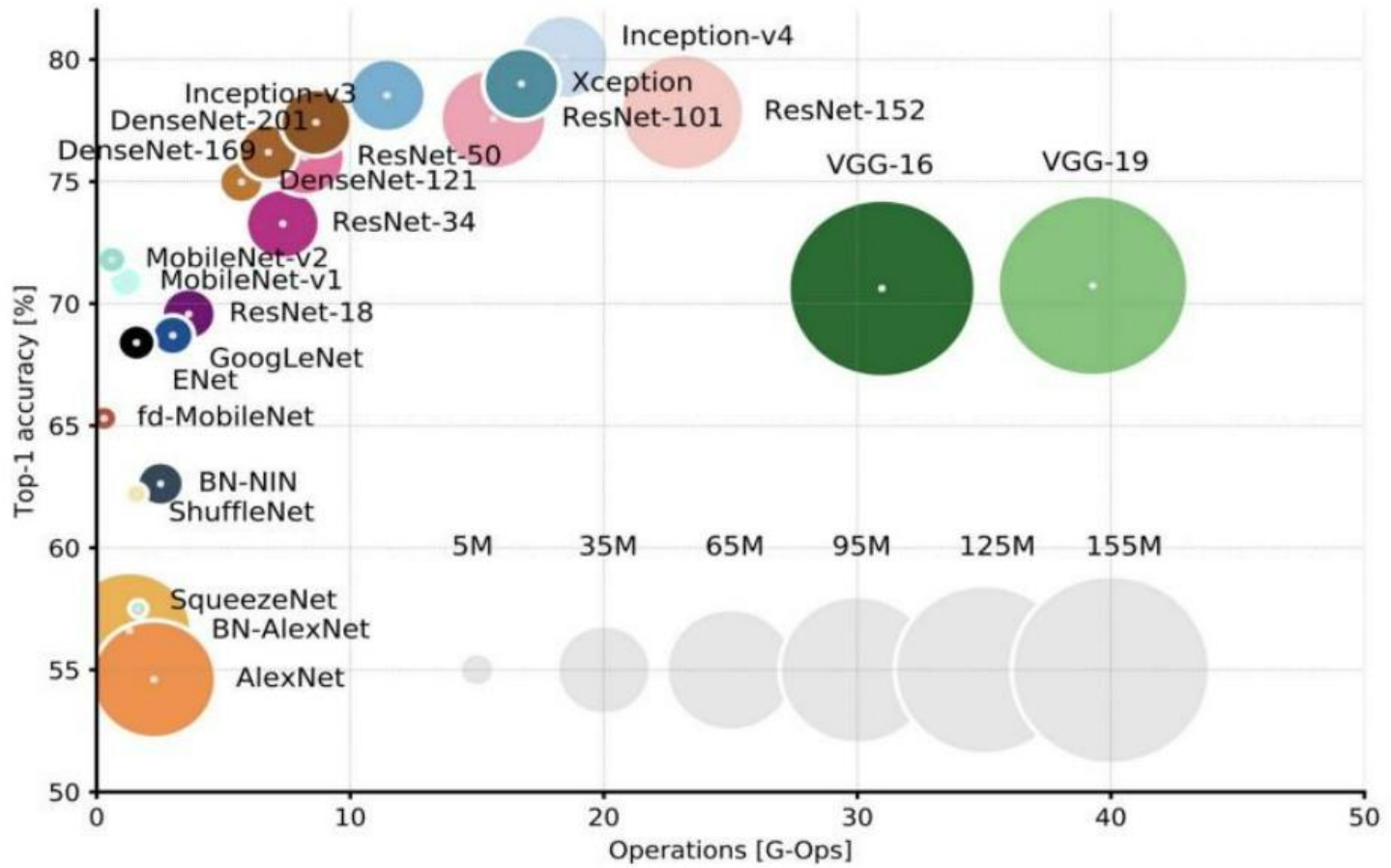


Архітектура AlexNet



Архітектура ResNet

РЕЗУЛЬТАТ ВИПРОБУВАНЬ

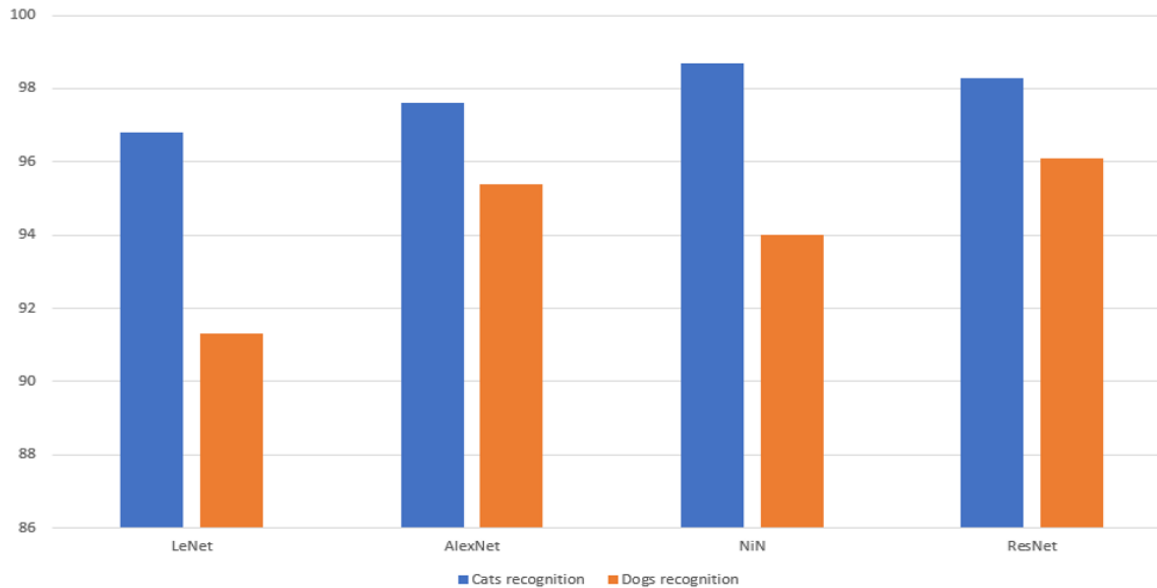


Результат випробувань різних архітектур нейронних мереж

РЕЗУЛЬТАТ ВЛАСНИХ ВИПРОБУВАНЬ

Тип архітектури	Час навчання, с
LeNet	784
AlexNet	759
NiN	732
ResNet	742

Час навчання нейронних мереж



Результат випробувань

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

10



Результат перевірки моделі



Результат роботи застосунку

В результаті виконання магістерської роботи було розроблено застосунок на основі згорткової нейронної мережі, що дає змогу значно розширити функціонал системи розумного будинку.

При виконанні роботи було виконано наступні задачі.

1. Проаналізовано існуючі рішення, що дають змогу розпізнавати та класифікувати зображення.
2. Проаналізовано існуючі типи нейронних мереж та обрано згорткову нейронну мережу для інтеграції в застосунок.
3. Проаналізовано існуючі архітектури згорткових мереж та обрано ResNet для неї.
4. Розроблено та проведено тестування програмного забезпечення, що може розпізнавати тварин (в даному випадку котів і собак) та класифікувати їх.

Статті:

1. Горелов А.І., Трінтіна Н.А., Негоденко О.В. Вибір оптимальної нейронної мережі системи розумного будинку для розпізнавання об'єктів. // Зв'язок. №3, 2021 (Подана до друку)

Тези доповідей на конференціях:

1. Горелов А.І. Використання нейронних мереж для розпізнавання образів в системах розумного будинку.// XIII Міжнародна науково-технічна конференція студентства та молоді «Світ інформації та телекомунікації». – Київ: ДУТ, 2021. – с.162-163.

ДЯКУЮ ЗА УВАГУ!