


КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА


на тему Метод для ідентифікації хвороб рослин за зображенням
засобами штучного інтелекту


Галузь знань 12 – Інформаційні технології
Шифр і назва галузі знань

Спеціальність 122 – Комп'ютерні науки
Шифр і назва спеціальності

Освітня програма Комп'ютерні науки
Назва освітньої програми

Виконав: студент 4 курсу, група КН-20-1  Валентин СОКОЛОВСЬКИЙ
Курс, група виконавця Підпис Ініціали, прізвище

Керівник: д.т.н., професор кафедри КН  Едуард МАНЗЮК
Науковий ступінь, посада Підпис Ініціали, прізвище

Нормоконтроль: к.т.н., доцент кафедри КН  Руслан БАГРІЙ
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:

Зав. кафедри КН, д.т.н., професор



Олександр БАРМАК

Підпис

Ініціали, прізвище

14 06 2024 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерних наук

Освітній ступінь бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

Освітня програма освітньо-професійна програма підготовки бакалавра

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

(підпис)

д.т.н., професор Олександр БАРМАК

« 16 » 02 2024 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА**

1. Тема кваліфікаційної роботи бакалавра: «Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту»
2. Завдання видано студенту Валентину СОКОЛОВСЬКОМУ
(прізвище, ім'я, по батькові)
3. Керівник роботи д.т.н. професор кафедри КН Едуард МАНЗЮК
(посада, прізвище, ім'я, по батькові)
4. Затверджено наказом університету від « 15 » 02 2024 р. № 8
5. Дата видачі завдання студенту: « 16 » 02 2024 р.
6. Зміст пояснювальної записки (перелік задач) та вихідні дані:


Мета кваліфікаційної роботи бакалавра полягає в покращенні ідентифікації хвороб рослин за зображенням засобами штучного інтелекту. Для досягнення поставленої мети необхідно провести аналіз предметної області, оглянути методи, які використовуються для розпізнавання захворювань рослин за допомогою штучного інтелекту. Розробити метод ідентифікації хвороб рослин за зображеннями за допомогою штучного інтелекту. Провести експериментальні дослідження ефективності застосування розробленого методу для розпізнавання захворювань рослин за допомогою штучного інтелекту.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником	грудень 2023	виконано
2	Ознайомлення з предметною областю, формулювання мети та задач дослідження, визначення об'єкта та предмета дослідження	січень 2024	виконано
3	Робота над розділом 1 – Характеристика предметної області та постановка задачі	січень 2024	виконано
4	Робота над розділом 2 – Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту	березень 2024	виконано
5	Робота над розділом 3 – Експериментальна перевірка методу для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту	квітень 2024	виконано
6	Оформлення пояснювальної записки згідно вимог	травень 2024	виконано
7	Попередній захист кваліфікаційної роботи бакалавра	травень 2024	виконано
8	Захист кваліфікаційної роботи бакалавра на засіданні Екзаменаційної комісії	червень 2024	виконано

Виконавець:

студент 4 курсу, група КН-20-1
Курс, група виконавця


Підпис

Валентин
СОКОЛОВСЬКИЙ
Ініціали, прізвище

Керівник:

д.т.н. професор кафедри КН
Науковий ступінь, посада


Підпис

Едуард МАНЗЮК
Ініціали, прізвище

Анотація

Тема кваліфікаційної роботи бакалавра: Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.

Виконавець кваліфікаційної роботи бакалавра: студент групи КН-20-1 Валентин СОКОЛОВСЬКИЙ

Керівник кваліфікаційної роботи бакалавра: д.т.н. професор кафедри КН Едуард МАНЗЮК

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
75	11	3	41	1

Мета кваліфікаційної роботи бакалавра полягає в покращенні ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.

Для досягнення поставленої мети визначені наступні задачі дослідження: визначити послідовність застосування методу ідентифікації хвороб рослин за зображенням засобами штучного інтелекту; розробити метод ідентифікації хвороб рослин за зображенням засобами штучного інтелекту; провести експериментальні дослідження ефективності застосування розробленого методу.

Результатом виконання кваліфікаційної роботи бакалавра є розроблений метод ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.

Ключові слова: ідентифікація хвороб рослин, аналіз зображення, машинне навчання.

Виконавець: студент 4 курсу, група КН-20-1
Курс, група виконавця


Підпис

Валентин СОКОЛОВСЬКИЙ
Ініціали, прізвище

Зміст

Вступ.....	7
Розділ 1 Характеристика предметної області та постановка задачі	9
1.1 Аналіз предметної області ідентифікації хвороб рослин за зображенням	9
1.2 Методи обробки зображення з використання штучного інтелекту	15
1.3 Мета та постановка задачі.....	20
Розділ 2 Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту	21
2.1 Структура нейронної мережі визначення захворювань рослин за зображеннями	22
2.3 Налаштування гіперпараметрів нейронної мережі.....	32
2.4 Модифікація архітектури шарів	35
2.5 Розширення даних для визначення хвороб	38
2.5 Метрики оцінювання ефективності моделі	44
Висновки до розділу 2	47
Розділ 3 Експериментальна перевірка методу для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.....	48
3.1 Методика проведення експерименту	48
3.2 Реалізація методу для ідентифікації хвороб рослин	54
3.3 Експериментальна перевірка ефективності додавання додаткових шарів ..	60
Висновки до розділу 3	65
Висновок	67
Перелік посилань.....	69
ДОДАТКИ.....	74

Перелік скорочень

Скорочення, термін, позначення	Пояснення
КРБ	Кваліфікаційна робота бакалавра
КН	Комп'ютерні науки
CNN	Згорткова нейронна мережа
DCNN	Глибока згорткова нейрона мережа
DL	Глибоке навчання
TL	Трансферне навчання

Вступ

Кваліфікаційна робота бакалавра націлена на розробку методу для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту. Цей метод дозволяє ідентифікувати хвороби рослин за зображенням та визначати їх оптимальні параметри, а також вести моніторинг з прогресом розвитку хвороб рослин базуючись на зображеннях рослин.

Актуальність – ідентифікація хвороб рослин за допомогою штучного інтелекту є важливою задачею в сільському господарстві, оскільки точність та швидкість діагностики можуть впливати на урожайність та якість вирощених культур. Для цієї предметної області проводиться аналіз різних методів, які застосовуються для автоматичної ідентифікації хвороб рослин на основі їх зображень.

Методи, що використовуються для ідентифікації хвороб рослин, базуються на алгоритмах машинного навчання, зокрема на глибокому навчанні, яке дозволяє аналізувати зображення з точністю. Такі методи використовують нейронні мережі для виявлення характерних ознак на зображеннях та подальшої класифікації рослин за наявністю або відсутністю хвороби. В роботі розроблено метод ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.

Об'єкт дослідження – процес ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Предмет дослідження – методи та технології машинного навчання для роботи із зображеннями

Мета кваліфікаційної роботи бакалавра полягає в покращенні якості ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Завдання кваліфікаційної роботи бакалавра.

Розробка методу ідентифікації хвороб рослин за зображенням включає наступні завдання:

- зробити аналіз методів аналізу набору зображень рослин з різними хворобами та здорових екземплярів;
- визначити послідовність застосування метода аналізу із застосуванням обробки зображень, включаючи видалення шуму, розмір зображення та інші техніки попередньої обробки;
- сформулювати модель глибокого навчання для класифікації хвороб за зображеннями;
- здійснити тренування моделі з використанням навчального набору зображень для навчання моделі на виявлення хвороб;
- провести оцінювання ефективності моделі на тестовому наборі даних для визначення її точності та надійності.

Розділ 1 Характеристика предметної області та постановка задачі

1.1 Аналіз предметної області ідентифікації хвороб рослин за зображенням

Актуальність теми дослідження ідентифікації хвороб рослин за зображенням засобами штучного інтелекту визначається необхідністю підвищення ефективності аграрного сектору через автоматизацію процесів діагностики та контролю за станом рослин. Сільське господарство стикається з постійною загрозою втрат врожаю внаслідок захворювань рослин, що можуть поширюватися швидко і важко виявлятися за звичайних умов [1, 2]. Використання штучного інтелекту для аналізу зображень дозволить оперативно виявляти хвороби, що забезпечить своєчасні заходи по їх лікуванню та запобіганню подальшого поширення [3, 4].

Технології машинного навчання, зокрема глибоке навчання, розвиваються стрімко і дозволяють досягати високої точності при класифікації зображень. Застосування цих технологій у сільському господарстві дозволить підвищити якість та кількість врожаю, знизити витрати на лікування рослин і зробити процес агроуправління більш ефективним. Такий підхід також дозволить забезпечити виробників сільськогосподарської продукції засобами для вчасного реагування на потенційні загрози для урожаю, що важливо у контексті змін клімату та зростання глобальної популяції. Таким чином, розробка та впровадження методу ідентифікації хвороб рослин за зображенням засобами штучного інтелекту має великий потенціал для покращення ситуації в аграрному секторі та підвищення його стійкості до внутрішніх та зовнішніх факторів [5–7].

З ростом світової популяції зростає попит на продукти сільського господарства, тому забезпечення високої якості та врожайності стає критично важливим. Використання штучного інтелекту для виявлення хвороб рослин може допомогти збільшити якість сільськогосподарських угідь, знизити втрати врожаю та забезпечити стійкість сільського господарства до зовнішніх впливів.

Крім того, автоматизована система ідентифікації хвороб рослин може допомогти фермерам зосередитися на інших аспектах виробництва, таких як оптимізація використання ресурсів, впровадження новітніх технологій та створення стійких інноваційних сільськогосподарських систем. Загалом, впровадження штучного інтелекту в сільське господарство для ідентифікації хвороб рослин може сприяти підвищенню ефективності та стійкості цієї галузі, зробивши її більш конкурентоспроможною та стабільною у майбутньому.

Без сільського господарства людське існування стає неможливим, оскільки воно забезпечує необхідні продукти для здорового життя. Сільське господарство є основою кожного суспільства у всьому світі, допомагаючи забезпечити працевлаштування, харчування, одяг та охорону здоров'я. Розвиток будь-якої країни в значній мірі залежить від сільського господарства. Крім того, сільське господарство відіграє важливу роль у забезпеченні продовольчої безпеки, особливо з урахуванням зростання світового населення та збільшення попиту на продукти харчування. Тому вдосконалення сільськогосподарських технологій має велике значення для задоволення цих потреб [8–10].

Здоров'я сільськогосподарських посівів є ключовим аспектом для досягнення оптимальної якості, тому необхідно систематично використовувати передові технології для моніторингу стану посівів. Моніторинг є важливим для сільського господарства, оскільки дозволяє активно боротися з хворобами та шкідниками, ефективно використовувати ресурси, оптимізувати врожайність та приймати обґрунтовані рішення на основі зібраних даних. Цей підхід допомагає фермерам оперативно виявляти труднощі, приймати обґрунтовані рішення та створювати стійкі та продуктивні системи сільського господарства. Наявність захворювань листя може негативно впливати на результативність сільського господарства та якість продуктів харчування. Раннє виявлення цих проблем може допомогти фермерам зекономити час і зусилля, сприяючи при цьому здоровому зростанню рослин. Однак точна ідентифікація хвороб листя може бути складною через різноманітність хвороб та шкідників, що атакують рослини.

Використання методів комп'ютерного зору для діагностики захворювань у листках рослин було досліджено в працях [11–15]. Ці методи дозволили ефективно виявляти різні захворювання, такі як борошниста роса, вірус жовтого скручування листя томатів, хвороба позеленіння цитрусових, хвороба іржі сої та інші. Однак, на шляху виявлення захворювань у листках рослин стояли певні труднощі, такі як отримання високоякісних зображень, що чітко фіксують характеристики здорових і хворих листків, вимагає спеціального обладнання та досвіду. Додатковим викликом є розробка алгоритмів, які можуть ефективно виявляти різні захворювання у різних видів рослин. Для досягнення успішної ідентифікації та класифікації об'єктів важливо поєднувати обробку зображень, машинне навчання і методи комп'ютерного зору. Такий підхід допомагає подолати проблеми сегментації зображень, виділення ознак та класифікації захворювань у рослин.

Хоча певні дослідники розробили модель машинного навчання для діагностики захворювань, їм довелося стикається з деякими викликами. Ці виклики включають обмежену здатність моделі обробляти невеликі обсяги даних, а також потребу у сегментації та вилученні ознак. Деякі попередні дослідження зосереджені на використанні традиційних методів машинного навчання для виявлення захворювань листя рослин на зображеннях, але ці методи надають менш точні результати. Інші дослідники провели порівняльний аналіз між традиційними алгоритмами машинного навчання і глибоким навчанням для класифікації зображень. Вони використовували різні набори даних для оцінки точності і часу роботи методів [16–19]. В результаті аналізу було встановлено, що традиційні моделі машинного навчання мають свої переваги, але також мають обмеження, які можна покращити за допомогою моделей глибокого навчання. Тому вони рекомендують використовувати моделі глибокого навчання для підвищення точності розпізнавання зображень.

Для виявлення та класифікації зображень листя рослин глибоке навчання дійсно демонструє здатність моделювати та вирішувати проблеми з великими обсягами даних. Серед різних підходів глибокого навчання для класифікації

зображень широко використовується згортова нейронна мережа [20–23]. Проте ці моделі мають велику кількість параметрів і потребують значних обчислень. Тому навчання з попереднім навчанням дає кращі результати.

У попередніх дослідженнях ідентифікації хвороб листя рослин на зображеннях розглядалися як класичні методи машинного навчання, такі як опорна векторна машина і дерева рішень, а також підходи глибокого навчання, такі як згортові нейронні мережі, попереднє навчання. Однак ці підходи мають свої обмеження. У дослідженні була запропонована модель згортової нейронної мережі, яка була навчена з використанням підходів до доповнення даних і показала високу точність класифікації [21]. Ця модель перевершує класичні методи машинного та попереднього навчання і демонструє послідовність та надійність, що робить її значним доповненням до цієї області досліджень. Об'єднання високоякісних смартфонів та комп'ютерного зору завдяки глибокому навчанню дозволяє використовувати смартфони для діагностики захворювань. У цій галузі застосовуються різні моделі глибокого навчання з точністю понад 99,53%. У даному дослідженні аналізуються архітектури з використанням передавання знань та глибокого вилучення функцій з допомогою зразків даних з Plant Village. Результати показують, що SVM є найкращим класифікатором для виявлення хвороб листя на зразках [24].

У наступній публікації пропонується нова модель глибокого навчання, яка базується на комбінації початкових і залишкових поєднань ознак, спрямованих на зменшення кількості параметрів. Це дозволяє досягти високої точності при класифікації захворювань рослин, в той час як модель виявляється менш вимогливою до обчислювальних ресурсів порівняно з іншими складними моделями глибокого навчання. Підхід був протестований на трьох різних наборах даних і продемонстрував вражаючі результати: точність на наборі даних PlantVillage складає 99,39%, на даних про захворювання рису – 99,66%, і на даних про захворювання маніюки – 76,59%. Ці результати підтверджують, що модель виявляється ефективнішою за рахунок оптимізації параметрів, що є

важливим в контексті розгляду розповсюдження глибокого навчання на різні пристрої [25].

Після завершення навчання моделі її якість може виявитися недостатньою для точної класифікації та виявлення. Для покращення її ефективності використовуються різні техніки, такі як налаштування гіперпараметрів, застосування ансамблю алгоритмів, вибір ознак, додавання додаткових шарів, використання додаткових наборів даних, зміна розміру зображення, збільшення кількості епох тощо. Для проекту вибрали техніки, такі як налаштування параметрів швидкості навчання, додавання додаткових шарів і застосування методів розширення даних. Ці методи довели свою ефективність у глибокому навчанні, зокрема в задачах класифікації зображень. Налаштування швидкості навчання допомагає моделі швидше збігатися і мінімізувати.

Під час додавання додаткових шарів до моделі існує ризик перенавчання, оскільки це може підвищити здатність моделі витягувати ознаки з вхідних даних. Техніки доповнення даних сприяють покращенню загальної здатності моделі до узагальнення шляхом створення нових навчальних прикладів і зменшення ризику перенавчання. Існують різні операції з даними, такі як обертання, горизонтальне та вертикальне перегортання, довільне кадрування, зміна кольору та інші, що можуть бути використані для доповнення навчального набору даних. Однак не всі комбінації цих операцій рівно ефективні для покращення якості моделі. Деякі операції, такі як випадкова ротація та комбінація кадрів, виявляються особливо ефективними, оскільки вони сприяють різноманітності навчального набору даних і допомагають уникнути перенавчання. Мета полягає в побудові моделі з найвищою точністю для виявлення хвороб листя рослин і класифікації зображень листя рослин.

Також застосовується трансферне навчання, що є потужним інструментом глибокого навчання, який дозволяє покращити ефективність моделей, особливо в умовах обмежених ресурсів або обмеженого обсягу даних. Основна ідея трансферного навчання полягає в тому, щоб скористатися знанням, набутим під час розв'язання одного завдання, для покращення результатів в

іншому, схожому завданні. Замість того, щоб навчати модель з нуля для кожного нового завдання, трансферне навчання дозволяє використовувати попередньо навчені ваги та знання з інших схожих задач.

Одним з основних переваг трансферного навчання є зменшення потреби у великих обсягах анотованих даних для нового завдання. Модель, навчена на великому датасеті для одного завдання, може бути успішно адаптована до іншого завдання з допомогою невеликого обсягу додаткових даних. Це особливо корисно у випадках, коли створення великих наборів анотованих даних є витратним або складним процесом.

Крім того, трансферне навчання допомагає уникнути перенавчання моделі на новому наборі даних, оскільки вона вже має певні загальні знання про властивості даних, які можна використовувати для нового завдання. Цей підхід підвищує стабільність та швидкість навчання моделі, роблячи його більш ефективним для вирішення реальних проблем, таких як класифікація захворювань листя рослин на зображеннях [26–28].

Трансферне навчання зменшує ризик перенавчання шляхом використання переданих знань із попередньо навчених моделей, що покращує розуміння інформації за допомогою вже перевірених архітектур. Цей підхід спрощує доступ до знань для фахівців і дозволяє користувачам використовувати попередні результати, що сприяє зменшенню бар'єрів у знаннях. Крім того, використання переданих представлень дозволяє усунути зміщення даних і забезпечити більшу надійність. Технологія також зменшує час навчання та витрати на обчислення, оскільки моделі трансферного навчання є легкими та ефективними для роботи на різних типах обладнання. Крім того, підвищує точність узагальнення моделей глибокого навчання, дозволяючи їм ефективно працювати з новими та невідомими даними. Загалом трансферне навчання підвищує якість, ефективність і здатність моделей глибокого навчання до узагальнення.

Раніше розпізнавання та діагностика захворювань листя рослин вимагали багато часу, грошей, спеціалізованих знань і часто були неточними. Але останні

досягнення у галузі комп'ютерного зору та машинного навчання значно спростили цей процес. Сучасні методи включають аналіз зображень листя рослин і використання спеціальних алгоритмів для виявлення паттернів і захворювань. Оскільки багато хвороб листя рослин мають схожий зовнішній вигляд, розпізнавання захворювань за зображеннями є важливим напрямом досліджень. Тому для точної ідентифікації та класифікації захворювань у листках рослин необхідно поєднання обробки зображень, машинного навчання та методів комп'ютерного зору.

Останні роки принесли значний прогрес у розвитку глибокого навчання, особливо в контексті класифікації зображень для швидкого та точного виявлення хвороб листя рослин. Однак моделі глибокого навчання мають свої ризики та проблеми, включаючи потребу у великих обсягах даних для ефективного навчання. Техніка трансферного навчання вирішує багато з цих проблем, використовуючи знання, набуте під час раніше вивчених завдань. Хоча трансферне навчання часто асоціюється з невеликими обсягами даних, його можна успішно застосовувати в різних ситуаціях, включаючи великі набори даних.

1.2 Методи обробки зображення з використання штучного інтелекту

Виявлення об'єктів за допомогою комп'ютерного зору є передовою технологією, яка дозволяє комп'ютерам аналізувати та розпізнавати об'єкти на цифрових зображеннях та відео. Ця технологія використовує методи комп'ютерного зору та обробки зображень для ідентифікації конкретних класів об'єктів, таких як автомобілі, велосипеди, люди тощо. Виявлення об'єктів має широкі застосування, включаючи сфери спостереження, автономного водіння, медичної візуалізації та робототехніки. Алгоритми виявлення об'єктів здатні ефективно аналізувати візуальні дані, виділяти ключові ознаки та точно розпізнавати та локалізувати об'єкти інтересу за допомогою складних алгоритмів та моделей глибокого навчання [29–33]. Ці моделі можуть бути навчені досягати

високої точності завдяки навчальним наборам даних, що містять різноманітні об'єкти з різних класів. Виявлення об'єктів, поєднуючи можливості комп'ютерного зору та обробки зображень, дозволяє комп'ютерам розуміти візуальне середовище та взаємодіяти з ним, відкриваючи широкі перспективи для творчості та вирішення проблем.

Нейронні мережі моделюються на основі моделі функціонування мозку і мають різні рівні, включаючи вхідний, прихований і вихідний. Кожен рівень складається з нейронів, які з'єднані з нейронами наступного рівня. Зв'язки між нейронами мають ваги, які використовуються для обчислення вхідних даних на наступний рівень. Кожен нейрон обчислює вихід, використовуючи вхідні дані, ваги зв'язків та власне зміщення, і використовує функцію активації для генерації виходу, який подається наступному рівню. На вихідному рівні використовується інша функція активації для отримання остаточного виходу мережі. Такий підхід дозволяє моделювати складні взаємозв'язки між даними і забезпечує високу гнучкість для розв'язання різноманітних завдань [34–37].

Глибокі нейронні мережі представляють собою тип нейронних мереж, які мають кілька рівнів з великою кількістю окремих вузлів. Ця глибина архітектури дозволяє мережам ефективно розв'язувати складні завдання, що вимагають великої обробки даних. Однак через більшу складність DL-мережі можуть вимагати більш тривалого процесу навчання порівняно з традиційними нейронними мережами. Головна ідея мереж полягає в тому, щоб забезпечити їм здатність працювати аналогічно до людського мозку, здатного обробляти дані і приймати рішення на основі широкого спектру шаблонів і типів даних. Це досягається через навчання системи на великому обсязі даних, що розширює її здатність розпізнавати та інтерпретувати складні структури вхідних даних.

Згорткові нейронні мережі широко використовуються для розпізнавання зображень через свою ефективність у зменшенні часу обчислень. Вони використовують фільтри для зменшення розміру вхідних даних зображень, що робить їх особливо ефективними. CNN складається з блоків згортки та пулінгу, які виконують обробку зображення через застосування різних фільтрів. Ці

фільтри допомагають виділяти особливості зображення, такі як границі чи форми, що забезпечує точніше розпізнавання та класифікацію. Після обробки фільтрами зображення проходить крізь кілька шарів згортки та пулінгу, щоб зменшити його розмір перед надсиланням до повністю підключених шарів для фінального аналізу. Ця складна структура дозволяє досягти високої ефективності в задачах обробки зображень, таких як розпізнавання об'єктів і сегментація [38–41].

Після проходження через шари згортки та пулінгу, зменшене зображення використовується в повністю зв'язаному шарі для остаточного аналізу. Повністю зв'язаний шар приймає векторизоване подання зображення і використовує його для класифікації чи визначення характеристик зображення.

Однією з ключових переваг CNN є їх здатність автоматично виявляти і використовувати важливі особливості зображення без необхідності ручної попередньої обробки. Фільтри в згорткових шарах вчаться автоматично під час процесу навчання, що дозволяє мережі адаптуватись до різних типів зображень і завдань.

Згорткові нейронні мережі є хорошим інструментом для багатьох задач аналізу зображень, таких як розпізнавання облич, класифікація об'єктів, виявлення елементів на зображенні та багато інших. Вони широко застосовуються у відеоспостереженні, медичному зображенні, автомобільній промисловості, віртуальній реальності та інших сферах, де обробка зображень відіграє важливу роль у розв'язанні завдань.

Трансферне навчання використовується для передачі знань, набутих у процесі навчання на одній області, до іншої пов'язаної області або завдання. Цей підхід особливо корисний у глибоких згорткових нейронних мережах, де навчання моделей вимагає великих обсягів мічених даних. Трансферне дозволяє побудувати ефективні класифікатори для конкретних даних, використовуючи знання, передане з попередньо навчених моделей на великому наборі даних. Його перевагою є прискорення процесу навчання та зменшення помилок.

Трансферне навчання відіграє важливу роль у глибокому навчанні, особливо в контексті глибоких згорткових нейронних мереж, де зазвичай вимагається значна кількість анотованих даних для успішного навчання моделей. Збір великого обсягу анотованих даних у специфічній області часто є складним завданням, але дозволяє використовувати знання, набуте з попередньо навчених моделей на інших задачах або наборах даних.

Основна ідея полягає в тому, щоб скористатися знанням, яке вже було зібрано й використано в іншому контексті, і застосувати його до нової задачі або області. Наприклад, якщо маємо попередньо навчену модель для класифікації зображень на великому наборі даних, можемо скористатися цією моделлю як базою або стартовою точкою для навчання моделі на меншому обсязі даних, але в тій же області. Це дозволяє ефективно використовувати знання, вже вивчене з попередніх даних, замість починати навчання спочатку з нуля.

Переваги включають зменшення часу та витрат на навчання моделі, оскільки частину необхідних знань модель вже засвоїла з попереднього досвіду. Крім того, може допомогти у вирішенні проблеми з перенесенням знань та узагальненням, оскільки модель навчається на різних, але пов'язаних даних.

Миттєва обробка зображення дозволяє швидко приймати рішення і втручатися у сільське господарство. Глибока архітектура моделі дозволяє засвоювати складні характеристики і шаблони зображень, що є необхідним для точної класифікації. Основна мотивація для використання моделі ResNet у дослідженні полягає в її здатності досягати високої ефективності порівняно з іншими моделями, що показало хороші результати у схожих задачах розпізнавання та класифікації зображень.

В цій роботі було використано різні техніки, такі як додавання додаткових шарів, регулювання швидкості навчання та операції збільшення даних, для покращення якості моделі ResNet у класифікації зображень хвороб листя рослин. Ці методи обрано з метою вирішення складних завдань, оптимізації навчання та забезпечення ефективної роботи з даними для навчання.

Інші дослідження показали, що є прогалини в вивченні конкретних методів, таких як налаштування швидкості навчання, розширення даних і додавання додаткових рівнів для виявлення хвороб рослин. Головною метою застосування цих методів є поліпшення якості моделі ResNet у виявленні та класифікації хвороб листя рослин на зображеннях рослин. Таким чином, основна мета дослідження полягає в пошуку ефективної стратегії для підвищення точності моделі ResNet у виявленні та класифікації хвороб листя рослин на зображеннях рослин. Успішне виявлення хвороб рослин є важливим для попередження захворювань, покращення врожайності та ефективного контролю за хворобами.

На основі аналізу було вирішено використати модель ResNet для дослідження через її відмінну якість у завданнях класифікації зображень, що робить її хорошим вибором для використання при діагностуванні хвороб рослин за зображеннями. Використання моделі ResNet для ідентифікації хвороб листя рослин і класифікації зображень дозволяє отримати значні переваги, такі як висока якість, ефективність.

Основна мета дослідження полягає у виборі найефективнішої техніки для підвищення точності та якості моделі у класифікації хвороб листя рослин. Це дослідження спрямоване на профілактику захворювань рослин, підвищення врожайності та контроль за станом здоров'я рослин. Підхід зробить суттєвий внесок у сільське господарство та вивчення патології рослин.

Під час виконання роботи необхідно провести такі етапи:

1. Розділити набір зображень листя рослин на дані для тренування та валідації.
2. Створити та навчити модель ResNet на навчальному наборі зображень листя рослин.
3. Випробувати різні методи, такі як налаштування швидкості навчання, додавання додаткових шарів та застосування операцій збільшення даних до моделі.

4. Оцінювати якість кожної моделі за відповідними метриками, порівнювати їх та вибрати найбільш ефективний метод для підвищення точності моделі.

1.3 Мета та постановка задачі

На основі проведеного детального аналізу сформульовану мету роботи, яка полягає у покращенні якості ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Розробка методу ідентифікації хвороб за зображенням включає наступні завдання:

- зробити аналіз методів аналізу набору зображень рослин з різними хворобами та здорових екземплярів;
- визначити послідовність застосування метода аналізу із застосуванням обробки зображень, включаючи видалення шуму, розмір зображення та інші техніки попередньої обробки;
- сформулювати модель глибокого навчання для класифікації хвороб за зображеннями;
- здійснити тренування моделі з використанням навчального набору зображень для навчання моделі на виявлення хвороб;
- провести оцінювання ефективності моделі на тестовому наборі даних для визначення її точності та надійності.

Розділ 2 Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

ResNet є гнучкою моделлю нейронної мережі, яка знаходить широке застосування у сфері комп'ютерного зору. Її легка структура та ефективні операції роблять її ідеальним вибором для пристроїв з обчислювальними ресурсами. Модель широко використовується у завданнях класифікації зображень, виявлення об'єктів, семантичної сегментації та передбачення. Її висока ефективність дозволяє точно розпізнавати об'єкти, виявляти хвороби листя рослин та аналізувати вирази обличчя. Загалом, її універсальність і сумісність з різними завданнями дозволяють їй ефективно працювати, що робить її потужним інструментом для задач глибокого навчання та комп'ютерного зору.

Архітектура моделі ResNet складається з багатьох рівнів, із яких певні рівні містять блоки, за якими слідує звичайний згортковий шар розміром 1×1 , глобальний середній рівень об'єднання та класифікаційний рівень. Будівельні блоки є основними компонентами архітектури ResNet. Вони містять послідовність операцій, які виконуються кілька разів і включають згортки, розділені по глибині для більш ефективної обробки даних.

Структура ResNet це спеціальна архітектура глибокої нейронної мережі, яка використовує резидуальні з'єднання "residual", що дозволяють ефективно тренувати дуже глибокі мережі з мінімальною втратою точності або зникненням градієнтів.

Основна ідея ResNet полягає у використанні блоків з резидуальним з'єднанням, які дозволяють перейти повз кілька шарів мережі, не втрачаючи важливу інформацію або можливість ефективної оптимізації під час навчання. Така конструкція дозволяє зберігати інформацію про вихід шарів, що полегшує розповсюдження градієнтів під час зворотного поширення.

2.1 Структура нейронної мережі визначення захворювань рослин за зображеннями

Структура ResNet складається з таких основних блоків. Базові згорткові блоки – ці блоки включають стандартні операції згортки, активації (наприклад, ReLU), а також можуть включати операції пакування (наприклад, пулінг).

Residual блоки – основний будівельний блок ResNet, який містить резидуальні з'єднання. Опишемо структура блоку.

Вхідна гілка – стандартна послідовність операцій (згортки, активації) над вхідними даними.

Резидуальне з'єднання – це з'єднання від входу блоку до виходу, яке додається безпосередньо до виходу після деяких шарів операцій.

Вихідна гілка – стандартні послідовності операцій над об'єднаною інформацією.

ResNet зазвичай має глибоку структуру з декількома блоками residual, де кількість шарів може сягати декількох сотень у найпотужніших варіаціях моделі. Загальна ідея резидуальних з'єднань полягає в тому, що мережа може "вчитися" додавати до вхідних даних що-небудь нове, а не намагатися повністю замінити вхід на вихід. Це дозволяє покращити стійкість до зникаючих градієнтів під час глибокого навчання.

У цій архітектурі використовуються блоки.

1. Блок з кроком 1, який зберігає розмір карти функцій таким самим, як вхідні дані. Цей блок містить два типи згорток, результати яких додаються безпосередньо до вхідних даних для збереження важливих деталей зображення та підвищення ефективності мережі.

2. Блок зі зменшенням розміру карти функцій удвічі, який використовує згортки по глибині для зменшення розміру карти функцій, а потім застосовує поточкову згортку для перетворення отриманих даних перед їх подальшим передачею до наступного блоку.

Додатково в архітектурі використовується згортковий шар розміром 1×1 для зменшення кількості каналів в блоках з обмеженими можливостями. Також, в архітектурі використовується згортковий шар розміром 1×1 у блоках з обмеженими можливостями для зменшення кількості каналів. Це дозволяє зберігати важливі функції при зменшенні обсягу обчислень.

У будівельних блоках використовуються різні стратегії згортки, що дозволяє моделі здобувати більш глибокі інтерпретації при переході через шари. Залишкові блоки з кроком 1 зберігають розмір карти функцій, що сприяє збереженню важливих деталей зображення. Блоки зі зменшенням розміру (з кроком 2) дозволяють зменшити розмір карти функцій наполовину, що сприяє підвищенню ефективності мережі при обробці зображень.

Одним із ключових елементів архітектури є також згортковий шар розміром 1×1 , що допомагає зменшити кількість каналів в обмежених блоках. Це оптимізує використання ресурсів і поліпшує якість мережі при збереженні її точності та ефективності.

Загалом, є досить складною і ефективною моделлю, яка використовує різні стратегії згортки та блоків для досягнення високої точності при роботі з обчислювальними ресурсами.

Продовження архітектури включає також використання глобального середнього рівня об'єднання після блоків згортки. Цей етап використовується для стиснення простору ознак, отриманих після згортки, в один вектор, який потім передається до класифікаційного рівня. Глобальний середній рівень об'єднання дозволяє отримати компактне представлення ознак і зменшити кількість параметрів, що обчислюються, що є важливим для ефективної роботи на пристроях.

Останнім етапом архітектури є класифікаційний рівень, який використовується для прогнозування класу зображення. Цей рівень зазвичай включає повнозв'язаний шар з відповідною кількістю вихідних класів для класифікації зображення.

Узагальнюючи, архітектура використовує різні стратегії згорток та блоків для створення легкої, ефективної і точної моделі глибокого навчання. Кожен етап архітектури має свою функціональність, спрямовану на зменшення обсягу обчислень та збереження важливих ознак зображення для точної класифікації та розпізнавання об'єктів.

Після класифікаційного рівня, де здійснюється прогнозування класу зображення, може використовуватися додатковий прошарок, наприклад, Softmax, для отримання ймовірностей присутності різних класів на зображенні. Крім того, після навчання моделі її можна використовувати для різних завдань комп'ютерного зору, таких як класифікація зображень, виявлення об'єктів або семантична сегментація. Ця модель володіє універсальністю і може успішно працювати на пристроях, завдяки своїй архітектурі та ефективним операціям згортки.

Моделі глибокого навчання, такі як ResNet, є дуже важливими для застосувань у сфері мобільного комп'ютерного зору, де потрібна висока точність та ефективність обробки зображень. Такі моделі можуть бути використані для розв'язання різних завдань від розпізнавання облич, класифікації об'єктів до аналізу медичних зображень або розпізнавання тексту в реальному часі.

У цілому, ResNet є потужним і гнучким інструментом у сфері комп'ютерного зору, який дозволяє створювати ефективні та точні моделі глибокого навчання. Для практичного застосування розглянемо два різні типи архітектури. Ці архітектури пов'язані з згортковими нейронними мережами які широко використовуються для завдань, таких як розпізнавання зображень та комп'ютерний зір.

В першому типі архітектурі потік починається з шару "введення", який потім передається через серію шарів – лінійний згортковий шар, це згортковий шар, який лінійний повністю пов'язаний шар. Далі шар розділення, згортковий шар, за яким іде функція активації ReLU. Вихід цих шарів потім передається назад лінійного згорткового шару, створюючи зворотний зв'язок або рекурентне з'єднання.

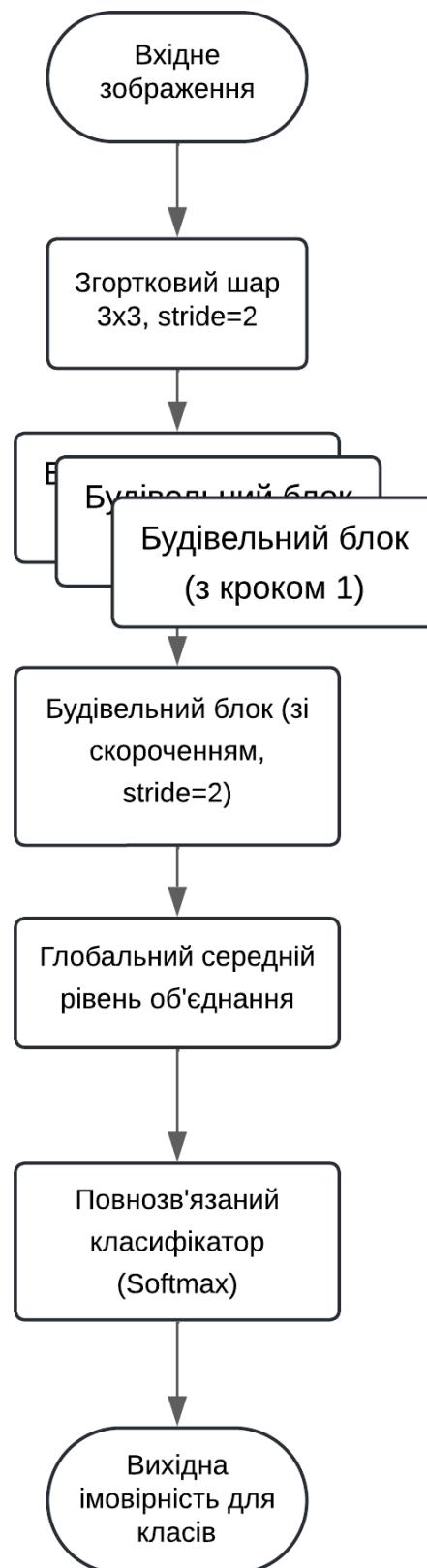


Рисунок 2.1 – Схематичне представлення архітектури моделі

На цій схемі вхідне зображення подається на перший згортковий шар розміром 3×3 з кроком 2, що допомагає зменшити розмір зображення.

Послідовно застосовуються будівельні блоки з різними типами: блоки з кроком 1, які зберігають розмір карти функцій, і блоки зі скороченням (з кроком 2), які зменшують розмір карти функцій наполовину.

Після останнього будівельного блоку використовується згортковий шар розміром 1×1 для подальшої обробки ознак. Після цього слідує глобальний середній рівень об'єднання, який об'єднує просторові розміри ознак у вектор фіксованого розміру.

На останньому етапі застосовується повнозв'язаний класифікатор з функцією активації Softmax для прогнозування ймовірностей належності зображення до різних класів.

Ця схема показує послідовність операцій у моделі ResNet, що дозволяє отримувати ефективно та точно розпізнавання об'єктів на зображеннях.

Детальне пояснення архітектури моделі полягає в наступному.

1. Згортковий шар 3×3 з кроком 2. Цей шар приймає вхідне зображення і застосовує згорткові фільтри розміром 3×3 з кроком 2 для зменшення розміру зображення. Він використовується для вилучення базових ознак зображення.

2. Будівельні блоки:

– після першого згорткового шару використовуються послідовно будівельні блоки;

– будівельні блоки можуть мати крок 1 (залишковий блок) або крок 2 (блок зі скороченням);

– залишковий блок зберігає розмір карти функцій таким самим, як у вхідних даних, і додає інформацію безпосередньо до вихідних даних.

– блок зі скороченням зменшує розмір карти функцій наполовину за допомогою згорткових шарів і з'єднання зі скороченням.

3. Згортковий шар 1×1 :

– після останнього будівельного блоку застосовується згортковий шар розміром 1×1 ;

– цей шар використовується для подальшої обробки ознак і зменшення їхньої глибини.

4. Глобальний середній рівень об'єднання:

– після згорткового шару 1×1 використовується глобальний середній рівень об'єднання;

– цей етап зменшує просторові розміри ознак, обчислюючи середнє значення по кожному каналу функцій.

5. Повнозв'язаний класифікатор (Softmax):

– на завершальному етапі використовується повнозв'язаний класифікатор з функцією активації Softmax;

– цей класифікатор приймає згорнуті ознаки після глобального середнього рівня об'єднання і прогнозує ймовірності належності зображення до різних класів.

В іншій архітектурі потік подібний, але порядок шарів трохи відрізняється. Він починається з шару "введення", який потім передається через лінійний згортковий шар. Далі іде шар розділення, за яким іде лінійний шар, функція активації ReLU. Вихід цих шарів потім передається до кінцевого шару .

Ключова різниця між двома архітектурами, полягає у значенні кроку, яке впливає на поведінку даунсемплінгу, пулінгу згортковим шарам. Крок 1 зберігає просторові розміри входу, тоді як крок 2 зменшував би просторові розміри удвічі.

Загалом, ці архітектури, досліджують різні способи структурування та з'єднання шарів CNN, можливо, для досягнення певних цілей якості чи архітектури.

Архітектура мережі з коригуванням ваг.

1. Вхідний шар – початковий шар, який приймає вхідні дані, такі як зображення чи вектори.

2. Згортковий шар – використовується для вилучення ознак з вхідних даних шляхом згортки фільтрів через вхідні дані. Це допомагає виявити важливі особливості.

3. Пулінговий шар - Відповідає за зменшення простору вздовж простору ознак, зберігаючи важливі інформаційні ознаки.

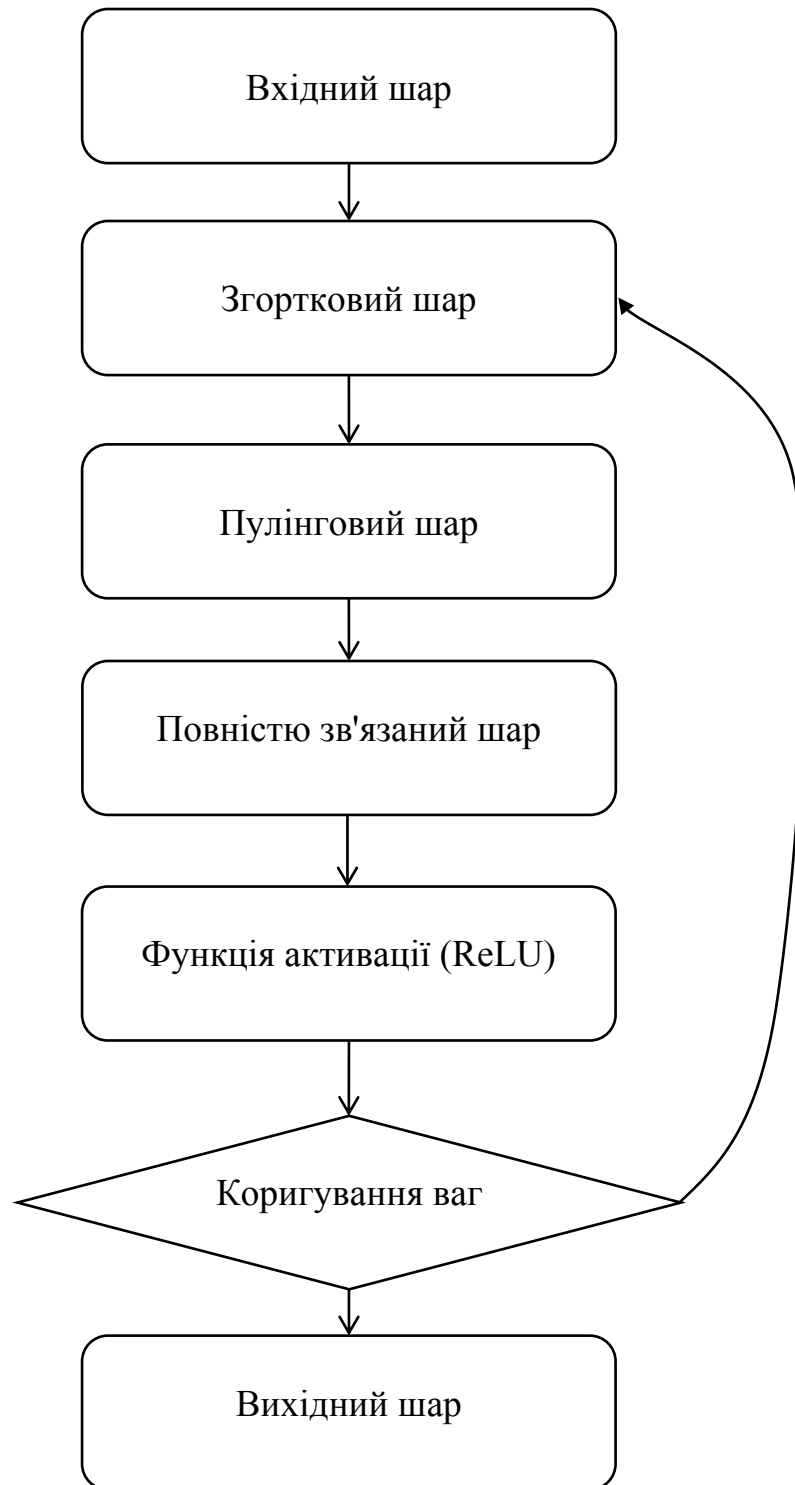


Рисунок 2.2 – Архітектура мережі з коригуванням ваг

4. Повністю зв'язаний шар – шар, в якому кожен нейрон зв'язаний з кожним нейроном попереднього шару. Він допомагає розпізнавати складні залежності між ознаками.

5. Функція активації – функція використовується для нелінійного перетворення вихідних даних після повністю зв'язаного шару, щоб додати нелінійність до моделі.

6. Зворотне з'єднання – шар допомагає здійснювати зворотний прохід для навчання моделі, коригуючи ваги шарів для мінімізації функції втрат.

7. Вихідний шар – останній шар, який видає вихідні дані моделі, наприклад, прогнози класів для задачі класифікації або значення для задачі регресії.

Архітектура застосовується для типів глибоких нейронних мереж, які зазвичай використовуються в зображеннях або інших типах даних з великою кількістю ознак.

Лінійна архітектура мережі:

1. Вхідний шар – початковий шар, який приймає вхідні дані.

2. Лінійний згортковий шар – цей шар використовується для вилучення ознак з вхідних даних шляхом згортки фільтрів. Це допомагає виявити корисні шаблони або ознаки у вхідних даних.

3. Шар розділення – цей шар може використовуватись для розділення або редукції розмірності даних перед подальшою обробкою.

4. Лінійний шар – шар, в якому кожен нейрон зв'язаний з кожним нейроном попереднього шару. Він використовується для впровадження лінійних перетворень у дані.

5. Функція активації – функція застосовується після лінійного шару для введення нелінійності в модель. ReLU використовується для активації більшості нейронних мереж через його простоту та ефективність.

6. Кінцевий шар – це останній шар, який видає вихідні дані моделі. Наприклад, це може бути шар для класифікації за допомогою softmax або регресії з одним нейроном виходу.

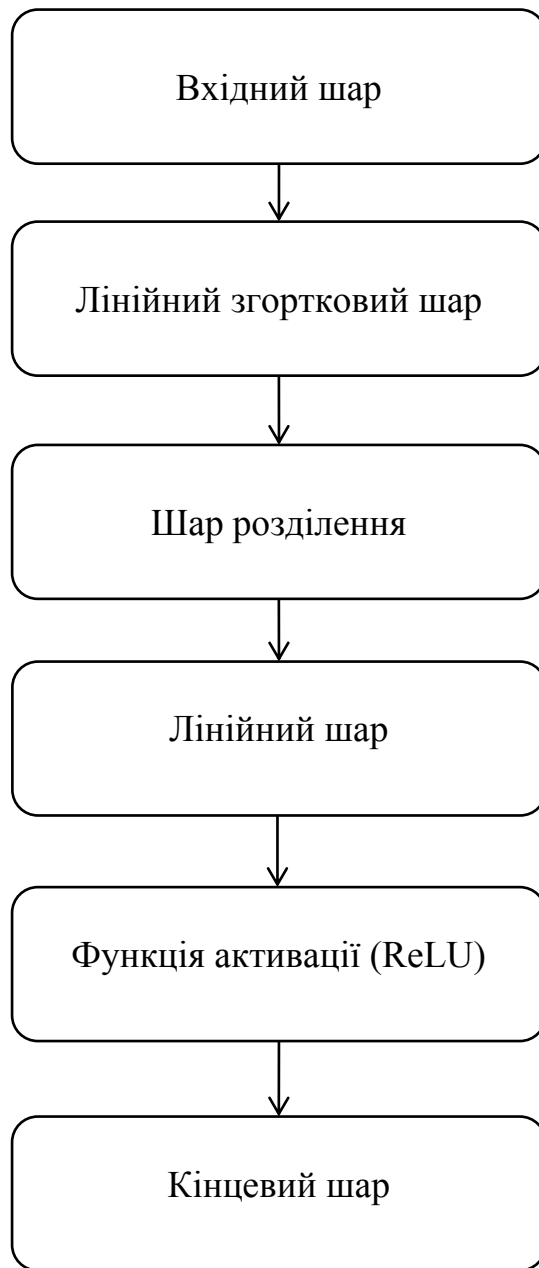


Рисунок 2.3 – Лінійна архітектура мережі

Ця архітектура є поширеною для глибоких нейронних мереж, що використовуються для обробки зображень або інших типів даних з великою кількістю ознак.

Основні відмінності між першою і другою описаною архітектурою полягають у порядку та типі шарів, що використовуються. Розглянемо ці відмінності детальніше:

Перша архітектура.

1. Початковий шар – "введення".

2. Послідовність шарів наступна: Лінійний згортковий шар → Згортковий шар → Повністю зв'язаний шар → Шар розділення → Згортковий шар з функцією активації ReLU → Зворотне з'єднання → Вихідний шар.

У першій архітектурі використовується послідовність згорткових і повністю зв'язаних шарів, після чого йде шар розділення і зворотне з'єднання для здійснення зворотного поширення помилок під час навчання.

Друга архітектура.

1. Початковий шар - "введення".

2. Послідовність шарів: Лінійний згортковий шар → Шар розділення → Лінійний шар з функцією активації ReLU → Кінцевий шар.

У другій архітектурі спочатку використовується лінійний згортковий шар, після чого йде шар розділення, лінійний шар з функцією активації ReLU і кінцевий шар.

Основні відмінності полягають в наступному.

1. Перша архітектура використовує послідовність згорткових, повністю зв'язаних та зворотньо зв'язаних шарів, в той час як друга архітектура використовує лінійний згортковий шар, шар розділення, лінійний шар з функцією активації і кінцевий шар.

2. Хоча обидві архітектури включають згорткові та лінійні шари, вони розташовані в різному порядку, що може впливати на вихідні характеристики моделі.

3. Цей шар присутній лише в першій архітектурі і відповідає за певну обробку чи розділення даних перед подальшою обробкою.

4. Тільки в першій архітектурі є зворотне з'єднання, яке дозволяє здійснювати зворотний прохід для навчання моделі.

Ці відмінності визначають різні підходи до обробки даних у мережі, що може вплинути на їх ефективність і здатність моделі до навчання і прогнозування.

Яка архітектура є більш ефективною, залежить від конкретної задачі, даних та умов навчання моделі. Обидві описані архітектури можуть бути ефективними відповідно до контексту.

Фактори, які можуть впливати на вибір ефективної архітектури.

1. Деякі архітектури краще підходять для обробки певних типів даних. Наприклад, для зображень зазвичай ефективно використовувати згорткові нейронні мережі, тоді як для послідовних даних, таких як текст чи аудіо, можуть підходити рекурентні або трансформерні архітектури.

2. Деякі задачі можуть вимагати більшої складності моделі з багатьма шарами та нейронами, тоді як інші можуть успішно розв'язуватися менш складними архітектурами.

3. Великі набори даних можуть потребувати більших та складніших моделей для досягнення високої точності.

4. Деякі архітектури можуть бути більш обчислювально витратними, що може впливати на їх практичну застосовність в залежності від доступних обчислювальних ресурсів.

5. Якщо потрібно виконати класифікацію, регресію або генерацію тексту, то різні архітектури можуть бути більш або менш підходящими для цих конкретних завдань.

У кінцевому підсумку, найкраща архітектура буде залежати від специфіки задачі, а також можливостей і обмежень доступних ресурсів. Часто ефективність моделі визначається експериментально шляхом порівняння різних архітектур на наборі даних та оцінювання їх результатів.

2.3 Налаштування гіперпараметрів нейронної мережі

Налаштування гіперпараметрів є критичною у розробці глибоких моделей, оскільки воно суттєво впливає на їх ефективність. Гіперпараметри – це параметри, які визначають поведінку моделі, наприклад, швидкість навчання, розмір пакету для тренування та кількість шарів. Вибір оптимальних

гіперпараметрів важливий для досягнення найкращої якості моделі. Це передбачає вибір найбільш підходящих комбінацій значень, які приведуть до оптимальних результатів.

Підбір оптимальних гіперпараметрів для глибокої нейронної мережі включає в себе деякі кроки і стратегії. Для кожного гіперпараметра, наприклад, швидкості навчання, кількості шарів, розміру пакету тощо можна спробувати кілька різних значень. Важливо зберігати записи про результати для подальшого аналізу.

Замість випадкового вибору значень гіперпараметрів, можна скористатися методами розсіювання, такими як розумний пошук або випадковий пошук, для систематичного пошуку простору гіперпараметрів.

Після тренування моделей з різними комбінаціями гіперпараметрів слід аналізувати результати. Це включає оцінку якості моделі за метриками якості (наприклад, точність, втрати) на валідаційному наборі даних.

На основі аналізу результатів вибирають оптимальні комбінації гіперпараметрів. Цей процес може займати декілька ітерацій для досягнення найкращої якості моделі. Досвід і знання про типові гіперпараметри для подібних завдань також можуть допомогти у виборі оптимальних значень.

У складних моделях може бути корисно використовувати автоматизовані інструменти для автоматичного підбору гіперпараметрів (наприклад, бібліотеки оптимізації гіперпараметрів). Налаштування гіперпараметрів – це ітеративний процес, який може вимагати багато спроб і помилок, але він допомагає підібрати оптимальні налаштування для моделі, щоб досягти найкращих результатів на конкретній задачі.

Крос-валідація дозволяє оцінити ефективність різних комбінацій гіперпараметрів шляхом розділення навчального набору даних на кілька піднаборів для тренування і валідації. Це допомагає уникнути перенавчання та забезпечити більш об'єктивну оцінку ефективності моделі.

Деякі гіперпараметри, такі як швидкість навчання і розмір пакету, можуть мати великий вплив на якість моделі. Потрібно бути особливо уважним у їх

виборі. Деякі гіперпараметри, які впливають на регуляризацію моделі (наприклад, коефіцієнти L_1 або L_2), можуть бути використані для запобігання перенавчанню і покращення загальної якості моделі. При виборі гіперпараметрів важливо враховувати обмеження на обчислювальні ресурси, такі як час тренування і доступну пам'ять. Наприклад, більші моделі можуть вимагати більше часу та ресурсів для тренування.

Після вибору оптимальних гіперпараметрів і тренування моделі на валідаційному наборі, важливо оцінити її якість на тестовому наборі, щоб переконатися в загальній ефективності. Після випробування різних комбінацій гіперпараметрів і отримання результатів може виникнути необхідність внесення коректив у їх значення для подальшого покращення моделі.

Регулювання швидкості навчання є важливим методом для оптимізації якості нейронних мереж. Цей параметр впливає на те, як швидко модель навчається і змінює свої ваги під час тренування. Правильно налаштована швидкість навчання дозволяє моделі оптимізуватися краще, що може покращити швидкість збіжності та точність.

Переваги коректного регулювання швидкості навчання включають:

- відповідні зміни швидкості навчання дозволяють моделі швидше досягати оптимального рішення, що скорочує час тренування;
- вірно налаштована швидкість навчання допомагає моделі краще оптимізувати свою функцію втрат і підвищити точність прогнозів;
- використання оптимальної швидкості навчання допомагає уникнути перенавантаження моделі на навчальних даних і покращити її здатність узагальнюватися до нових даних;
- параметр швидкості навчання можна налаштовувати для досягнення оптимальної якості, спостерігаючи за впливом на результати моделі.

Налаштування швидкості навчання вимагає досвіду і експериментів. Недооптимальна швидкість навчання може спричинити повільну збіжність або ускладнити досягнення глобального мінімуму функції втрат. Через це важливо

дбати про баланс між швидкістю збіжності та точністю моделі при налаштуванні цього параметра.

Точне налаштування швидкості навчання є важливим аспектом тренування нейронних мереж. Основна мета полягає в тому, щоб забезпечити оптимальний баланс між швидкістю збіжності і уникненням перенавчання чи втрати глобального мінімуму.

Якщо швидкість навчання встановлена надто високою, це може призвести до того, що модель буде "скакати" навколо точки мінімуму функції втрат без збіжності. Це відоме як проблема "великих кроків". З іншого боку, якщо швидкість навчання занадто низька, процес тренування може бути дуже повільним, а модель може застрягти в локальних мінімумах, не досягаючи глобального мінімуму.

Ітеративний підхід до налаштування швидкості навчання дозволяє виявити оптимальне значення, при якому модель навчається ефективно і швидко збігається до бажаного результату. Це може включати поступове збільшення швидкості навчання протягом тренування, спостереження за метриками ефективності такими як функція втрат, і зупинку тренування, якщо ефективність моделі погіршується.

Загальним підходом є використання методів оптимізації, таких як стохастичний градієнтний спуск з адаптивним регулюванням швидкості навчання, які автоматично налаштовують швидкість навчання на основі градієнтів, що обчислюються під час тренування. Правильно налаштований параметр може суттєво підвищити ефективність навчання нейронної мережі і поліпшити її загальну якість.

2.4 Модифікація архітектури шарів

Додавання додаткових шарів до нейронних мереж є популярним методом для підвищення якості моделей. Завдяки цим шарам модель може отримати більш детальні представлення вхідних даних, що призводить до більш точних

прогнозів. Ця стратегія також допомагає уникнути перенавчання, коли модель стає занадто складною і починає реагувати на шум в даних, а не на основні патерни.

Шари, які можна додати до нейронних мереж, включають згорткові шари, шари пулінгу об'єднання, шари відкидання та нормалізації. Кожен тип шару має свою конкретну функцію, наприклад, зменшення просторових розмірів вхідних даних або введення випадковості для попередження перенавчання.

В архітектурі нейронної мережі існує багато різних рівнів, таких як згорткові, повністю зв'язані і шари пулінгу. Кожен з цих шарів грає важливу роль у прискоренні навчання та покращенні якості моделі. Наприклад, згортковий шар є основним будівельним блоком у архітектурі згорткових нейронних мереж, де він виконує згорткові операції над вхідними даними, перетворюючи їх у більш складні візуальні ознаки.

Згортковий шар є ключовим компонентом у архітектурі згорткових нейронних мереж. Цей шар використовує згорткові операції для обробки вхідних даних, розпізнаючи в них різноманітні візуальні шаблони та особливості. Операція згортки полягає в застосуванні фільтрів ядер до різних частин вхідних даних, щоб виділити корисні ознаки. Результати цих операцій передаються наступним шарам для подальшого аналізу.

Після згорткового шару зазвичай додається шар пулінгу об'єднання. Шар пулінгу використовується для зменшення розмірності простору даних, зберігаючи важливі ознаки. Це дозволяє зменшити кількість параметрів у мережі і сприяє уникненню перенавчання.

Інший корисний шар для запобігання перенавчання - це шар відкидання. Цей шар випадковим чином вимикає певну частину нейронів під час кожної ітерації навчання. Такий випадковий вибір допомагає моделі стати більш стійкою до перенавчання, оскільки нейрони не можуть надмірно спеціалізуватися на конкретних шаблонах.

Нормалізаційні шари також важливі для покращення якості моделі. Вони допомагають стандартизувати дані, забезпечуючи більш стійке навчання і

прискорюючи процес оптимізації. Усі ці додаткові шари дозволяють покращити якість прогнозів моделі та зробити її більш ефективною у роботі з великими обсягами даних. Однак важливо збалансувати складність моделі з обсягом доступних даних і уникати перевантаження нейронної мережі.

Повнозв'язний шар забезпечує повну зв'язність між кожним нейроном попереднього шару і кожним нейроном поточного шару. Це дозволяє проводити більш складний аналіз вхідних даних, оскільки кожен вихідний нейрон обробляє інформацію від усіх нейронів попереднього шару. У повнозв'язному шарі відбувається множення матриці на вектор, де параметри матриці є навчальними параметрами моделі, які оновлюються під час навчання за допомогою алгоритму зворотного поширення помилки. Вихід повнозв'язного шару є вектором фіксованого розміру, що визначається кількістю нейронів у цьому шарі.

Шар об'єднання використовується для зменшення просторових розмірів вхідних даних шляхом дискретизації. Операція пулінгу застосовується незалежно до кожної карти ознак вхідного об'єму, при цьому глибина об'єму залишається незмінною. Одним із найпоширеніших видів операції пулінгу є максимальне об'єднання, де обирається максимальне значення в певному вікні вхідних даних. Це сприяє зменшенню кількості параметрів у моделі та допомагає уникнути перенавчання, підкреслюючи найважливіші ознаки вхідних даних.

Повнозв'язний шар, також відомий як щільний шар, є ключовим елементом у багатьох типах нейронних мереж, зокрема у звичайних штучних нейронних мережах і глибоких нейронних мережах. Цей шар забезпечує повне з'єднання між входами нейронами попереднього шару і виходами нейронами поточного шару, що дозволяє моделі виконувати складний аналіз вхідних даних. Кожен нейрон повнозв'язного шару приймає на вхід значення всіх нейронів попереднього шару, і його вихід обчислюється за допомогою лінійної комбінації цих вхідних значень з вагами, які є навчальними параметрами моделі.

Шар пулінгу, або шар об'єднання, використовується для зменшення просторових розмірів вхідних даних, зберігаючи при цьому найбільш важливі ознаки. Це дозволяє знизити кількість параметрів у моделі і скоротити

обчислювальні витрати, що є важливими для покращення швидкодії нейронних мереж. Одним з найпоширеніших видів операції пулінгу є максимальне об'єднання, де для кожної області вхідних даних обирається максимальне значення. Це допомагає виділити найбільш важливі ознаки з вхідних даних і підвищує робастність моделі до варіацій вхідних зображень або даних.

Загальна архітектура моделі нейронної мережі зазвичай включає послідовність різних шарів, включаючи згорткові, повнозв'язні, шари пулінгу та інші, які використовуються для ефективного обробки і аналізу вхідних даних. Оптимальний вибір типів шарів і їх параметрів дозволяє досягти кращої якості моделі, забезпечуючи оптимальне співвідношення між складністю моделі і здатністю уникнути перенавчання.

2.5 Розширення даних для визначення хвороб

Розширення даних використовується для створення додаткових навчальних даних на основі наявних. Це корисний підхід, особливо ефективний у завданнях комп'ютерного зору та обробки природної мови. Збільшення даних допомагає знизити ризик перенавчання та підвищує загальну здатність моделі до узагальнення. Шляхом створення різних варіацій вихідних даних збільшується розмір набору даних, що сприяє уникненню перенавчання і забезпечує моделі різноманітні приклади для навчання.

Існує багато методів розширення даних, які можна застосовувати до навчального набору. Деякі з найпоширеніших прийомів включають обертання, горизонтальне та вертикальне відображення, зміну колірної палітри та випадковий вибір зображень з набору даних для створення різноманітних варіацій.

Розширення даних є важливою стратегією для підвищення ефективності навчання нейронних мереж, особливо у випадках обробки зображень та текстів. Цей метод дозволяє створювати додаткові приклади даних на основі існуючих,

що допомагає уникнути перенавчання моделі та підвищити її здатність до узагальнення на нові дані.

Існує кілька основних операцій розширення даних, які можна застосувати до навчального набору.

1. Обертання. Поворот зображення на випадковий кут допомагає створити нові варіації огляду об'єктів із збереженням їх сутності.

Обертання зображень є одним з ефективних методів розширення даних, особливо у контексті навчання нейронних мереж, що працюють з візуальними даними, такими як зображення. Цей процес полягає у повороті зображення на випадковий кут відносно центру або іншої точки, що дозволяє створити нові варіації огляду об'єктів, при цьому зберігаючи їхню сутність і характеристики.

При обертанні зображення можна здійснювати різні типи поворотів або ж випадковий поворот на будь-який кут у заданому діапазоні. Цей процес дозволяє моделі навчатися розпізнавати об'єкти під різними кутами та перспективами, що робить її більш стійкою до варіацій у вхідних даних.

Основна мета обертання полягає у тому, щоб створити різні варіанти зображень, які можуть зустрічатися в реальних умовах, і навчити модель розпізнавати об'єкти у різних геометричних конфігураціях. Це сприяє покращенню узагальнюючої здатності моделі, оскільки вона навчається виділяти ключові ознаки об'єктів незалежно від їх положення та орієнтації на зображенні.

Після обертання зображення можуть застосовуватися додаткові операції, такі як зміна розміру, обрізання або додавання шуму, щоб створити ще більше різноманітних варіацій для навчання моделі. Цей процес розширення даних допомагає покращити загальну ефективність і точність нейронної мережі, зокрема в умовах недостатнього обсягу навчальних даних.



Рисунок 2.5 – Зображення листя яблуні навчального набору даних з обертанням

2. Горизонтальне та вертикальне відображення. Відображення зображення по горизонталі або вертикалі дозволяє отримати нові перспективи об'єктів. Горизонтальне та вертикальне відображення є важливими техніками розширення даних, особливо у випадках обробки зображень. Ці методи дозволяють створювати нові варіації зображень шляхом відображення їх по горизонталі або вертикалі, що допомагає моделі учитися розпізнавати об'єкти з різних позицій та перспектив.

Горизонтальне відображення полягає у відображенні зображення відносно вертикальної вісі, що еквівалентно відображенню зображення зліва направо. Наприклад, якщо на зображенні зображено обличчя людини, то після горизонтального відображення обличчя буде виглядати, ніби дивиться в іншу сторону. Це дозволяє створювати нові зображення, які можуть зустрічатися в реальних ситуаціях, з іншим розташуванням об'єктів.

Вертикальне відображення відображається відносно горизонтальної вісі, що еквівалентно відображенню зображення зверху вниз. Наприклад, якщо на зображенні зображено пейзаж з небом угорі і землею внизу, після вертикального відображення небо опиниться внизу, а земля - угорі. Це дозволяє навчити модель розпізнавати об'єкти з інших точок зору та підготувати її до різних умов освітлення та орієнтації.

Ці техніки є корисними для створення різноманітних варіацій даних під час навчання нейронної мережі. Вони допомагають зробити модель більш стійкою до змін у вхідних даних і підвищують її здатність до узагальнення на нові дані, що робить її більш ефективною в реальних застосуваннях.



Рисунок 2.6 – Зображення листя яблуні з навчального набору даних (зеркальне відображення)

3. Зміна кольору. Випадкові зміни параметрів кольору, такі як яскравість, контрастність або насиченість, можуть створити різні варіації того ж зображення.

Зміна кольору зображень за допомогою випадкових трансформацій параметрів, таких як яскравість, контрастність або насиченість, є корисним методом для створення різноманітних варіацій тих самих зображень. Ці трансформації дозволяють моделі навчатися на більш різноманітних даних, що сприяє покращенню її здатності до узагальнення на нові умови та зміни.

Яскравість зображення може змінюватися шляхом додавання або віднімання певного значення інтенсивності кожного пікселя. Збільшення яскравості робить зображення яскравішим, тоді як зменшення може затемнити його.



Рисунок 2.7 – Зображення зі зміною яскравості

Контрастність визначає різницю між світлими і темними областями зображення. Збільшення контрастності робить зображення більш насиченим та чітким, а зменшення може зменшити різницю між тонами. Насиченість впливає на яскравість кольорів на зображенні. Збільшення насиченості робить кольори більш насиченими та живими, а зменшення може зробити їх більш приглушеними.

Ці випадкові зміни кольору допомагають збагатити набір даних і покращити здатність моделі до розпізнавання об'єктів у різних умовах освітлення, кольорових тонів та налаштувань. Вони забезпечують моделі більш широкий спектр навчання та допомагають їй ефективно пристосовуватися до реальних умов застосування.

4. Випадковий вибір. Вибір випадкової частини зображення для навчання допомагає моделі вчитися розпізнавати об'єкти у різних контекстах.

Техніка випадкового вибору частини зображення для навчання є важливим методом розширення даних, що допомагає моделі навчатися розпізнавати об'єкти у різних контекстах та під різними умовами. При використанні цієї техніки випадковим чином обирається певна частина зображення для навчання, що дозволяє моделі бачити об'єкти в різних ракурсах і з різних точок зору.

Наприклад, якщо маємо зображення пейзажу з дорогою і деревами, випадковий вибір частини зображення може означати використання лише дороги або лише частини дерев для навчання моделі. Це дозволяє моделі фокусуватися на конкретних деталях та вчитися розпізнавати об'єкти в різних контекстах.

Використання випадкового вибору частини зображення також допомагає уникнути перенавчання, оскільки модель навчається на більш різноманітних даних, що може підвищити її здатність до узагальнення на нові, реальні зображення.

Ця техніка розширення даних використовується в задачах машинного бачення, де важливо мати навчальний набір, який адекватно відображає різні умови та ситуації, з якими може зіткнутися модель у реальному світі. Вона сприяє покращенню загальної роботи моделі та її здатності до точного та надійного розпізнавання об'єктів на зображеннях.

5. Додавання шуму. Додавання випадкового шуму до зображення або тексту також може створити різні варіанти вхідних даних.



Рисунок 2.8 – Частина зображення несфокусована

Додавання шуму до зображення або тексту є ефективним способом розширення даних, який допомагає створити різні варіації вхідних даних для навчання моделі. Ця техніка полягає в додаванні випадкових змін шуму до

оригінального зображення або тексту, що імітує різні умови збереження або перепади даних у реальному світі.

Наприклад, у випадку зображень, додавання шуму може включати випадкові збої пікселів, що спричиняють візуальні спотворення, такі як малюнки, плями або перешкоди. Це допомагає моделі навчатися впоратися з різними рівнями шуму та інтерпретувати вхідні дані більш гнучко.

У випадку текстових даних, додавання шуму може включати випадкові зміни символів або словосполучень, помилкові друкарські помилки або зміни форматування тексту. Це дозволяє моделі навчатися розпізнавати текст навіть у випадку присутності помилок або неточностей.

Переваги додавання шуму включають зменшення ризику перенавчання, оскільки модель навчається розпізнавати об'єкти або патерни у різних умовах, а також підвищення стійкості моделі до шуму та інших аномалій у вхідних даних. Таким чином, додавання шуму є важливим етапом у підготовці навчальних даних для моделей машинного навчання, оскільки воно допомагає збільшити різноманітність даних та покращити здатність моделі до узагальнення на нові, реальні умови.

Ці операції розширення даних допомагають моделі збільшити свою робочу область, навчаючи її на більш різноманітних прикладах. Це робить модель більш стійкою до різних умов і допомагає забезпечити кращі результати на нових, реальних даних.

2.5 Метрики оцінювання ефективності моделі

Метрики оцінювання використовуються для визначення ефективності моделі шляхом вимірювання того, наскільки точно вона прогнозує результати і які показники її роботи є найбільш корисними. Ці метрики допомагають визначити рівень істинно позитивних та негативних класифікацій, а також кількість помилкових позитивних і помилкових негативних результатів, що допомагає зрозуміти, як модель виконується у різних сценаріях.

Налаштування метрик оцінювання є важливим кроком у роботі з моделями машинного навчання, оскільки вони дозволяють отримати об'єктивну оцінку ефективності моделі на основі її прогнозів. Основні показники оцінювання включають такі характеристики.

1. Істинно позитивний – кількість випадків, коли модель правильно класифікувала позитивні (1) екземпляри як позитивні.

2. Істинно негативний – Кількість випадків, коли модель правильно класифікувала негативні (0) екземпляри як негативні.

3. Помилково позитивний – кількість випадків, коли модель помилково класифікувала негативні екземпляри як позитивні.

4. Помилково негативний – кількість випадків, коли модель помилково класифікувала позитивні екземпляри як негативні.

Ці показники дозволяють розрахувати різні метрики, такі як точність (accuracy), чутливість (recall), специфічність (specificity), точність (precision) та F1-оцінка, які відображають різні аспекти ефективності моделі в залежності від конкретної задачі.

Наприклад, точність (accuracy) вимірює загальну точність моделі, враховуючи всі класи прогнозів. Повнота (recall) вимірює здатність моделі виявляти всі позитивні екземпляри, тоді як специфічність (specificity) вимірює її здатність правильно ідентифікувати негативні екземпляри.

Загальна мета налаштування метрик оцінювання полягає в тому, щоб обрати такі показники, які найкраще відображають вимоги конкретної задачі і допоможуть зрозуміти, як модель працює у реальних умовах. Наприклад, для задач з високою важливістю уникнення помилкових позитивних результатів може бути корисна висока точність (precision), тоді як для завдань з високою важливістю виявлення всіх позитивних випадків — висока чутливість (recall).

Точність (accuracy) визначається таким чином:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

де: TP – кількість істинно позитивних (правильно класифікованих позитивних випадків);

TN – кількість істинно негативних (правильно класифікованих негативних випадків);

FP – кількість помилково позитивних (негативні випадки, які були помилково класифіковані як позитивні);

FN – кількість помилково негативних (позитивні випадки, які були помилково класифіковані як негативні).

Повнота (recall):

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2.2)$$

Повнота вимірює здатність моделі правильно виявляти всі позитивні екземпляри.

Специфічність (specificity):

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (2.3)$$

Специфічність вимірює здатність моделі правильно ідентифікувати негативні екземпляри.

Точність (precision):

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2.4)$$

Точність вимірює відсоток позитивних прогнозів, які були правильними.

F1-оцінка:

$$F1_{\text{Score}} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.5)$$

F1-оцінка є гармонічним середнім між точністю і чутливістю і використовується для збалансованого оцінювання моделі.

Ці формули дозволяють обчислити основні показники ефективності моделі на основі її прогнозів та реальних класів. Інтерпретація цих метрик допомагає зрозуміти, наскільки добре модель працює у вирішенні конкретної задачі класифікації або прогнозування.

Висновки до розділу 2

У розділі було розроблено метод використання згорткових нейронних мереж, зокрема архітектури ResNet, для розпізнавання та класифікації хвороб рослин за зображеннями листя. Представлено детальний опис архітектури ResNet з її модифікаціями для ефективного застосування в предметній області .

Висвітлено ключові етапи архітектури, такі як згорткові шари, залишкові блоки, шари об'єднання тощо, які дозволяють ефективно витягувати візуальні ознаки та класифікувати зображення. Описано важливість правильного налаштування гіперпараметрів, зокрема швидкості навчання, для забезпечення оптимальної якості моделі.

Також розглянуто різні методи розширення даних, включаючи обертання, відображення, зміни кольору, випадковий вибір та додавання шуму. Ці техніки допомагають збільшити різноманітність навчальних даних, покращити узагальнюючу здатність моделі та запобігти перенавчанню.

Представлено низку метрик оцінювання, таких як точність, чутливість, специфічність, точність та F1-оцінка, які використовуються для оцінки ефективності моделі на основі її прогнозів. Правильний вибір і інтерпретація цих метрик дозволяє зрозуміти сильні та слабкі сторони моделі в конкретній задачі класифікації.

Розділ 3 Експериментальна перевірка методу для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

3.1 Методика проведення експерименту

У рамках роботи використано експериментальний підхід для досягнення цілей дослідження та відповіді на поставлені запитання. Провели експеримент на основі архітектури ResNet, що базується на CNN, застосувавши три різні методи. Для навчання моделі скористались новим набором даних про хвороби листя рослин з kaggle. Цей експеримент був проведений після початкового навчання моделі без застосування технік, щоб порівняти ефективність кожної моделі. На основі порівняння трьох різних методів було визначено найбільш ефективний спосіб підвищення точності моделі.

Процес експерименту можна описати так.

Крок 1: Вхідними даними для методу є набір зображень листя рослин із відповідним файлом анотацій, в яких містяться шляхи до зображень, координати обмежувальних рамок та класифікаційні мітки.

Крок 2: Базова модель виявлення об'єктів ResNet побудована на основі вагових коефіцієнтів. Додаткові шари додаються поверх базової архітектури моделі для адаптації її до виявлення хвороб листя рослин і класифікації зображень листя рослин.

Крок 3: Модель готується до навчання шляхом компіляції з використанням функції втрат середньоквадратичної помилки.

Крок 4: Під час навчання модель тренується за допомогою спеціального генератора, який створює пакети зображень і відповідні мітки. Процес навчання включає декілька епох, де кожна епоха включає подачу пакетів зображень до моделі, обчислення втрат і вдосконалення ваг моделі з використанням градієнтів.

Крок 5: Після навчання модель зберігається у файлі для подальшого використання. Збережена модель завантажується, а тестовий набір даних обробляється для оцінки якості моделі. Тестовий набір містить зображення та

відповідні анотації. Модель прогнозує мітки на основі тестових даних, і результати порівнюються з реальними мітками для обчислення показників оцінки, таких як втрати, точність, відтворюваність та оцінка F1.

Опишемо процес методології експерименту, яка якісно представить кроки та взаємозв'язки у роботі з моделлю для ідентифікації хвороб листя рослин на зображеннях.

1. Вхідні дані:

– колекція зображень листя рослин з відповідними анотаціями шляхи зображень, координати обмежувальних рамок, класифікаційні мітки.

2. Базова модель:

– побудована на основі підготовлених вагових коефіцієнтів;
– додаткові шари для адаптації до виявлення хвороб листя рослин і класифікації.

3. Компіляція моделі:

– використання оптимізатора функції втрат середньоквадратична помилка.

4. Навчання моделі:

– використання спеціального генератора для пакетів зображень і міток;
– тренування моделі протягом декількох епох з оновленням ваг за допомогою градієнтів.

5. Збереження моделі. Збереження навченої моделі для подальшого використання.

6. Оцінка якості моделі:

– завантаження збереженої моделі;
– обробка тестового набору даних для оцінки точності моделі;
– порівняння прогнозів моделі з реальними мітками для обчислення метрик оцінки.

Ця схема передає послідовність кроків експерименту з використання моделі ResNet для розв'язання завдання ідентифікації хвороб листя рослин на

зображеннях. Вона відображає, як вхідні дані проходять через підготовку моделі, навчання, збереження та оцінку якості, показуючи важливі етапи експерименту.

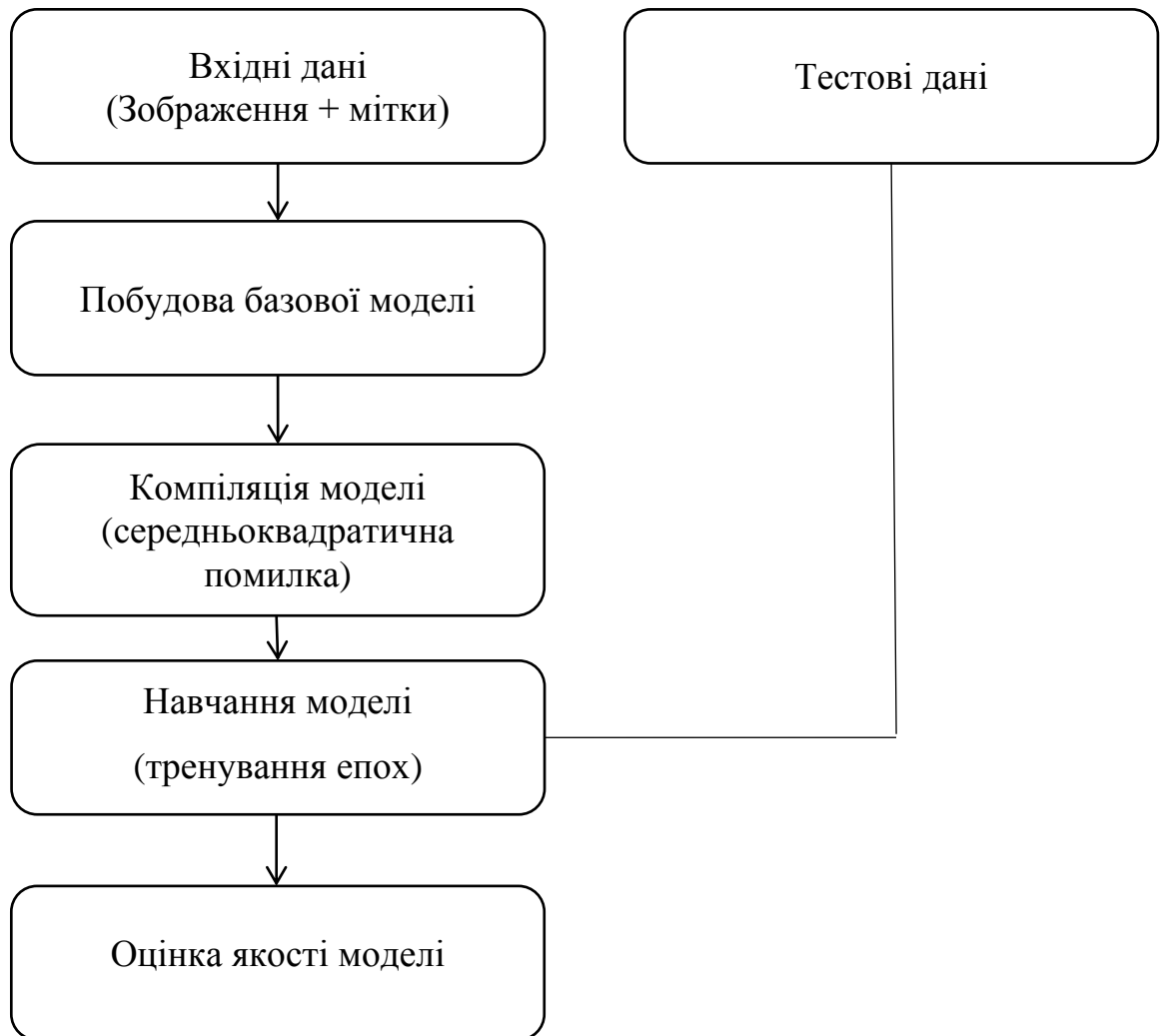


Рисунок 3.1 – Схема проведення експерименту

Ця діаграма вказує на послідовність дій у методології експерименту.

1. Вхідні дані, що складаються з зображень листя рослин з відповідними мітками, використовуються для навчання моделі.
2. Базова модель ResNet будується з підготовленими вагами.
3. Модель компілюється з використанням функції втрат середньоквадратичної помилки.
4. Модель навчається за допомогою генератора пакетів зображень протягом декількох епох.

5. Після навчання модель оцінюється за допомогою тестового набору даних для визначення її якості точність, втрати, F1-оцінка.

Діаграма дає загальне уявлення про процес експерименту з використання моделі ResNet для ідентифікації хвороб листя рослин на зображеннях. Здійснимо опис методології експерименту з використання моделі ResNet для ідентифікації хвороб листя рослин на зображеннях.

Починаємо зі збору колекції зображень листя рослин разом з відповідними файлами анотацій, які містять інформацію про класи або категорії хвороб. Ці дані попередньо обробляються для виділення шляхів до зображень, координат обмежувальних рамок та міток класів. Це набір зображень, які використовуються для тренування моделі. Кожне зображення містить листя рослини, яке може бути здоровим або хворим, і має відповідний файл анотацій, що містить інформацію про клас наприклад, хвороба, а також координати обмежувальних рамок для виділення регіону інтересу.

Використовуємо базову модель ResNet, яка має підготовлені ваги з набору даних. Додаткові шари будуються поверх цієї архітектури, щоб адаптувати модель для виявлення хвороб листя рослин та їх класифікації на зображеннях. Базова модель ResNet використовується як основа для виявлення об'єктів у зображеннях. Ця модель має підготовлені ваги, що допомагає у виявленні загальних функцій в зображеннях.

На базову архітектуру додаються додаткові шари, які адаптовані для конкретної задачі виявлення хвороб листя рослин. Ці шари допомагають використовувати вихідні дані для визначення наявності і типу хвороби на зображенні.

Модель компілюється з використанням функції втрат, такої як середньоквадратична помилка Це готує модель до фази навчання. Під час фази навчання модель тренується за допомогою спеціального генератора пакетів зображень. Цей генератор створює пакети зображень разом з їхніми відповідними мітками, і модель навчається на цих даних протягом декількох епох. Під час навчання модель оптимізує свої ваги, використовуючи градієнти,

обчислені на основі втрат. Модель компілюється з використанням оптимізатора функції втрат, такої як середньоквадратична помилка. Оптимізатор використовується для налаштування ваг моделі, а функція втрат визначає, як оцінюється різниця між прогнозами моделі та справжніми мітками. Під час навчання моделі використовується спеціальний генератор даних, який генерує пакети зображень та їхні мітки і використовується для навчання моделі.

Модель навчається протягом кількох епох, де кожна епоха включає подачу пакетів даних у модель, обчислення втрат і корекцію ваг моделі на основі градієнтів. Після завершення навчання моделі її якість оцінюється за допомогою тестового набору даних, який містить зображення листя рослин разом з відповідними мітками. Модель використовується для передбачення міток на цих зображеннях, і результати порівнюються з істинними мітками для визначення точності, втрат та інших показників ефективності моделі.

Після навчання модель зберігається у файл для подальшого використання. Збережена модель використовується для оцінки якості на тестовому наборі даних. Результати прогнозів порівнюються зі справжніми мітками для обчислення різних показників ефективності моделі.

Кроки дозволяють систематично провести експерименти з моделлю для ідентифікації та класифікації хвороб листя рослин на зображеннях. Кожен крок є важливим для успішного тренування та оцінки моделі, щоб досягти мети ідентифікації хвороб рослин з використанням методів машинного навчання.

Методологія експерименту дозволяє систематично досліджувати та оцінювати ефективність моделі у виявленні та класифікації хвороб листя рослин на зображеннях. Процес включає підготовку даних, побудову моделі, її навчання та оцінку результатів, що дозволяє зробити висновки про якість моделі та ефективність використаних підходів.

Нейронна мережа є згортковою нейронною мережею, яка призначена для обробки зображень. Розглянемо її структуру більш детально.

1. Вхідний шар.

Вхідні зображення мають три кольорові канали (RGB), оскільки перший згортковий шар має вхідний канал з розміром 3.

2. Перший згортковий шар.

Згортковий шар з 64 фільтрами ядрами розміром 3×3 з кроком 1 і додаванням обмежувачів розміром 1.

Після згорткового шару застосовується функція активації ReLU.

Після цього застосовується нормалізація по партіях.

3. Другий згортковий шар з пулінгом.

Згортковий шар з 128 фільтрами розміром 3×3 з ReLU.

Після згорткового шару використовується операція підвибірки з ядром розміром 4×4 і кроком 4.

4. Блок "Residual" у вигляді послідовності.

Послідовність двох згорткових шарів з 128 фільтрами із ReLU, кожен з яких застосовується після нормалізації.

5. Третій згортковий шар з пулінгом.

Згортковий шар з 256 фільтрами розміром 3×3 з ReLU.

Після згорткового шару використовується операція підвибірки з ядром розміром 4×4 і кроком 4.

6. Четвертий згортковий шар з пулінгом.

Згортковий шар з 512 фільтрами розміром 3×3 з ReLU.

Після згорткового шару використовується операція підвибірки з ядром розміром 4×4 і кроком 4.

7. Другий блок "residual" у вигляді послідовності.

Послідовність двох згорткових шарів з 512 фільтрами із ReLU, кожен з яких застосовується після нормалізації.

8. Класифікатор.

Після останнього згорткового шару застосовується операція підвибірки.

Після цього застосовується функція flatten для перетворення тензора у вектор.

На виході використовується повністю зв'язаний шар з 512 вхідними нейронами і 38 вихідними нейронами, що відповідає кількості масивів для класифікації зображень.

Нейронна мережа складається з декількох згорткових шарів з активаціями ReLU, операціями підвибірки, блоками "residual" для поліпшення навчання та підключеним класифікатором на основі повністю зв'язаних шарів для кінцевої обробки і класифікації зображень. Така структура дозволяє мережі ефективно вчитися розпізнавати патерни та особливості вхідних зображень.

3.2 Реалізація методу для ідентифікації хвороб рослин

У цьому дослідженні було проведено експеримент, щоб оцінити вплив різних темпів навчання на якість моделі для класифікації хвороб листя рослин. Було проведено аналіз різних показників навчання від 0,1 до 0,01 протягом різних епох. Головною метою було визначити оптимальну швидкість навчання, яка максимізує точність, відтворюваність та оцінку F1 моделі. Результати цього експерименту дають важливі висновки щодо впливу швидкості навчання на процес тренування моделі та загальну її якість.

Нижче представлені результати оцінки якості моделі, яка була навчена з використанням швидкості навчання 0,1 та різною кількістю епох. У таблиці 3.1 показані метри якості для кожної з епох:

Таблиця 3.1 – Показники оцінки якості моделі

Епоха	Accuracy	Recall	F1-показник
5	0,78	0,80	0,78
10	0,83	0,81	0,79
20	0,84	0,82	0,82
30	0,86	0,82	0,83
40	0,84	0,83	0,83

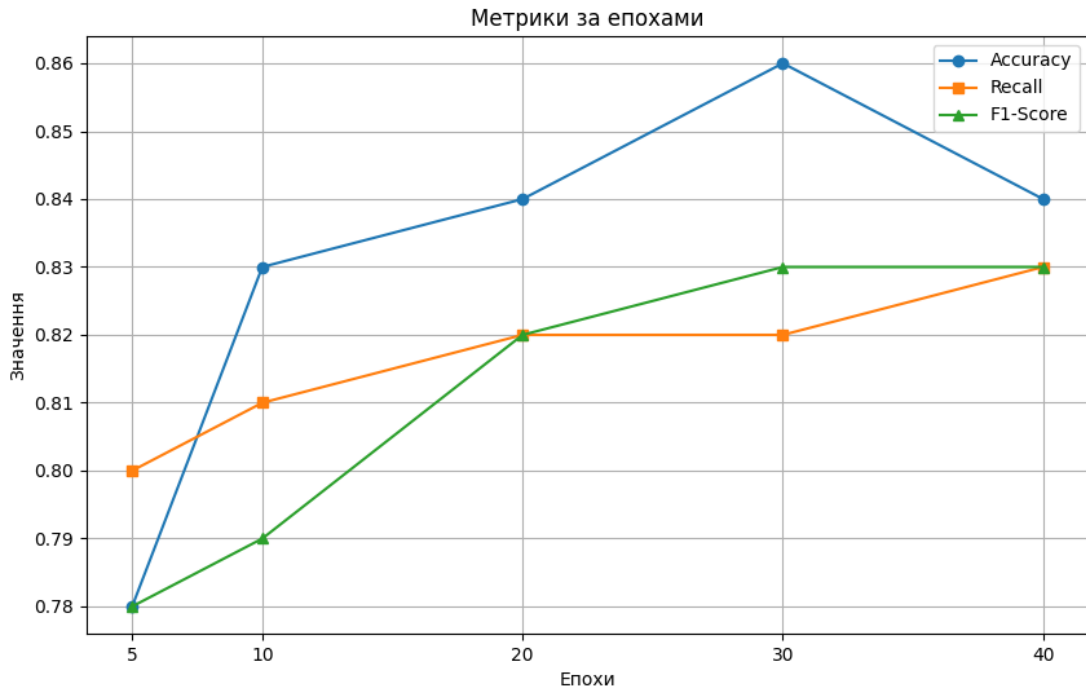


Рисунок 3.2 – Результати оцінки якості моделі

Аналіз результатів полягає в наступному. Середня точність збільшується зі зростанням кількості епох до 30, але потім трохи знижується на 40 епохі. Максимальне значення досягається на 30 епохах 0.86. Загалом, точність збільшується зі збільшенням кількості епох, але це може сигналізувати про перенавчання на більш високих епохах.

Recall також покращується зі збільшенням кількості епох і досягає максимуму на 30 епохах 0.82. Зниження recall на 40 епохах може також бути зв'язане з перенавчанням. F1-показник показує стабільне покращення зі зростанням кількості епох і досягає максимуму на 40 епохах 0.83.

Загалом, з даних можна зробити висновок, що оптимальним значенням для більшості метрик є 30-40 епох, коли модель досягає більшої загальної якості. Однак може бути корисно використовувати регуляризацію або інші методи контролю перенавчання для поліпшення стійкості моделі на високих епохах.

Точність показує позитивну тенденцію зі збільшенням кількості епох, що може свідчити про покращення роботи моделі з тривалістю навчання. Recall також покращується зі зростанням кількості епох і досягає максимуму на 30

епохах 0.82, після чого стабілізується. Це означає, що модель стає більш чутливою до виявлення позитивних класів зі збільшенням кількості епох навчання. F1-показник показує стабільне покращення зі зростанням кількості епох і досягає максимуму на 30-40 епохах 0.83. Цей показник об'єднує точність і recall відгук, відображаючи баланс між ними, і показується стабільним на високих епохах. Зменшення ефективності на 40 епохах може бути пов'язане з перенавчанням моделі через занадто довгий час навчання. Це підкреслює важливість вибору оптимального числа епох для навчання моделі з урахуванням показників ефективності. Оптимальним числом епох для навчання моделі, з точки зору максимальної точності, recall та F1-показника, здається бути в діапазоні 30-40 епох. Важливо підтримувати баланс між підвищенням ефективності і уникненням перенавчання моделі з подовженням тривалості навчання.

Таблиця 3.2 – Показники оцінки якості моделі

Епоха	Accuracy	Recall	F1-показник
5	0,81	0,79	0,80
10	0,85	0,83	0,81
20	0,86	0,85	0,82
30	0,86	0,83	0,83
40	0,85	0,83	0,83

Проведемо аналіз даних з таблиці метрик accuracy, recall, F1-показник для різних епох. Тенденції зміни метрик зі зростанням епох. Accuracy збільшується протягом перших епох, але стабілізується навколо 0.86 після 30-40 епох. Recall також збільшується і потім стабілізується на рівні близько 0.83-0.85. F1-показник показує певне покращення протягом тренування і стабілізується навколо 0.83 після 20-30 епох.

Accuracy має найвищі значення серед усіх метрик і стабілізується на високому рівні після певної кількості епох. Recall і F1-показник, хоча і

підвищуються зі зростанням епох, не досягають таких великих значень, як Accuracy.

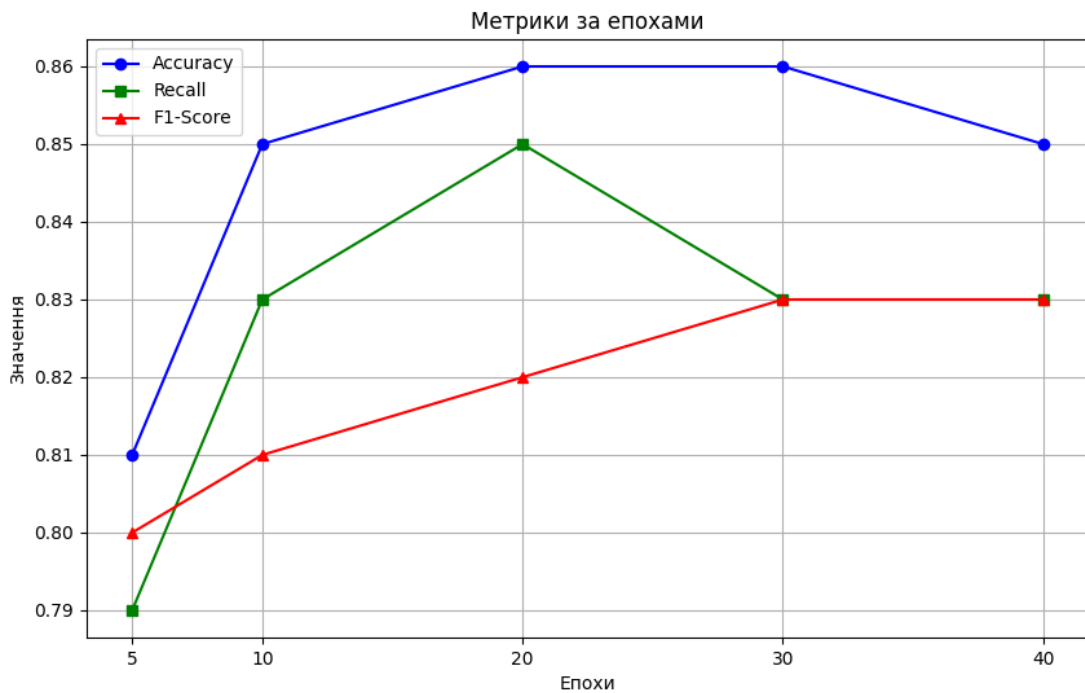


Рисунок 3.3 – Результати оцінки якості моделі

Модель має стабільні значення Accuracy після 30-40 епох, що може свідчити про її навчання без перенавчання. Recall і F1-показник також стабілізуються, хоча їхні значення не досягають пікових значень Accuracy.

Оптимальна кількість епох для цих даних може бути між 30 і 40, де метрики досягають стабільного рівня ефективності без великої зміни в подальшому навчанні. Для підвищення метрик можна спробувати збільшити розмір навчального набору даних, використати оптимізатори зі зменшеною швидкістю навчання або використовувати техніки регуляризації.

Ці дані показують, як ефективність моделі змінюється зі зростанням епох і як важливо вибрати оптимальну кількість епох для досягнення максимальної ефективності. Стабілізація метрик після певної кількості епох може свідчити про досягнення точки, коли додаткове навчання не приносить значного покращення в якості моделі. Цей аналіз допомагає зрозуміти динаміку тренування моделі і

можливі шляхи покращення її ефективності на основі навчальних даних і вибору параметрів навчання.

У результаті аналізу даних для різних епох можна виявити тенденції зміни метрик. По-перше, спостерігається загальне покращення всіх трьох метрик зі зростанням числа епох. Це означає, що модель навчається краще і показує вищу точність, відновлення і гармонічне середнє зі збільшенням кількості навчальних ітерацій. Ассигасу збільшується на близько 6-7% протягом тренування, що свідчить про те, що модель стає більш точною у передбаченнях зі зростанням числа епох.

Recall також показує помітне підвищення на 5-8%, що означає, що модель стає краще розпізнавати позитивні класи зі збільшенням кількості навчальних ітерацій. F1-показник, хоча й демонструє менше збільшення порівняно з ассигасу і recall, все ж відзначається зростанням на 1-4%, що свідчить про покращення узгодженості між точністю і повнотою моделі зі зростанням тренувальних ітерацій. Зауважимо також, що після певної кількості епох, наприклад, після 30-40 епох значення метрик стабілізуються, що може свідчити про досягнення певного рівня ефективності моделі. Це важливо для визначення оптимальної кількості навчальних ітерацій.

Для порівняльного аналізу двох моделей з різними значеннями швидкості навчання $lr = 0.1$ та $lr = 0.01$, оцінюємо їх за такими метриками: ассигасу, recall, і F1-показник на кожній з п'яти епох навчання 5, 10, 20, 30, 40.

$lr = 0.1$:

– Ассигасу підвищується протягом всього навчання, з 0.78 на початку до 0.85 на 10-й епохі;

– Recall також зростає, але не досягає значення 0.85, стабілізуючись на 0.83;

– F1-показник зменшується невеликою мірою після 20-ї епохи, стабілізуючись на 0.83.

$lr = 0.01$:

- Ассурасу починає з 0.81 і також зростає з кожною епохою, досягаючи 0.86 на 20-й епохі;

- Recall постійно підвищується до 0.85 на 20-й епохі, після чого залишається стабільним;

- F1-показник зростає до 0.83 на 20-й епохі, після чого залишається на тому рівні.

Модель з $lr = 0.01$ має більш високі значення ассурасу, recall, і F1-показник порівняно з моделлю з $lr = 0.1$ після 20-ї епохи навчання.

Модель з $lr = 0.01$ досягає стабільного значення метрик швидше на 20-й епохі, тоді як у моделі з $lr = 0.1$ це відбувається більш повільно.

Модель з меншим значенням швидкість навчання $lr = 0.01$ виявилася ефективнішою за розглянутими метриками порівняно з моделлю з $lr = 0.1$.

У порівняльному аналізі моделей з різними значеннями швидкість навчання $lr = 0.1$ та $lr = 0.01$ за допомогою метрик ассурасу, recall і F1-показник на п'яти епохах навчання 5, 10, 20, 30, 40 було виявлено наступне.

Починаючи зі значення ассурасу на рівні 0.78 на першій епохі, модель показує поступове покращення, досягаючи максимуму 0.85 на 10-й епохі, після чого стабілізується. Проте значення recall і F1-показник досягає піку на 10-й епохі 0.83 і залишається стабільним пізніше.

Початкове значення ассурасу у цій моделі становить 0.81 і постійно зростає до максимуму 0.86 на 20-й епохі, після чого залишається стабільним. Значення recall досягає свого піка також на 20-й епохі 0.85 і залишається на цьому рівні, так само як і F1-показник, який досягає значення 0.83.

Порівнюючи обидві моделі, можна зробити наступні спостереження:

- модель з $lr = 0.01$ показує кращі результати з метриками ассурасу, recall і F1-показник, досягаючи більш високих значень на більш ранній етап 20 епоха, ніж модель з $lr = 0.1$;

- значення метрик у моделі з $lr = 0.01$ стабілізуються на вищому рівні, що може вказувати на більш стабільну і ефективну збіжність моделі;

– модель з $lr = 0.1$ потребує більшого числа епох для досягнення підйому точності і стабільності, хоча вона також досягає досить прийнятних значень метрик.

Отже, за умов даної порівняльної аналізу, модель з меншим значенням швидкості навчання $lr = 0.01$ виявилася більш ефективною за метриками *accuracy*, *recall* і F1-показник порівняно з моделлю з $lr = 0.1$.

3.3 Експериментальна перевірка ефективності додавання додаткових шарів

Був проведений експеримент з додаванням додаткових шарів у модель. Тестували різні комбінації шарів, таких як об'єднання згорткового і щільного шарів. Метою було визначити оптимальні комбінації шарів, які найефективніше підвищують результати моделі за різними метриками. Нижче наведено отримані результати для різних комбінацій шарів у таблиці 3.3.

Таблиця 3.3 – Показники оцінки якості моделі з додатковим згортковим шаром

Епоха	Accuracy	Recall	F1-показник
5	0,83	0,82	0,81
10	0,88	0,83	0,81
20	0,90	0,85	0,84
30	0,93	0,86	0,86
40	0,91	0,85	0,82

Таблиця представляє результати оцінки моделі за допомогою метрик *accuracy*, *recall* і F1-показник на різних епохах? число ітерацій під час навчання моделі. *Accuracy* вимірює загальну точність моделі у класифікації. З таблиці видно, що точність моделі зростає зі збільшенням числа епох навчання до певного моменту 30 епох, після чого вона знижується. Найвища досягнута точність складає 93% при 30 епохах.

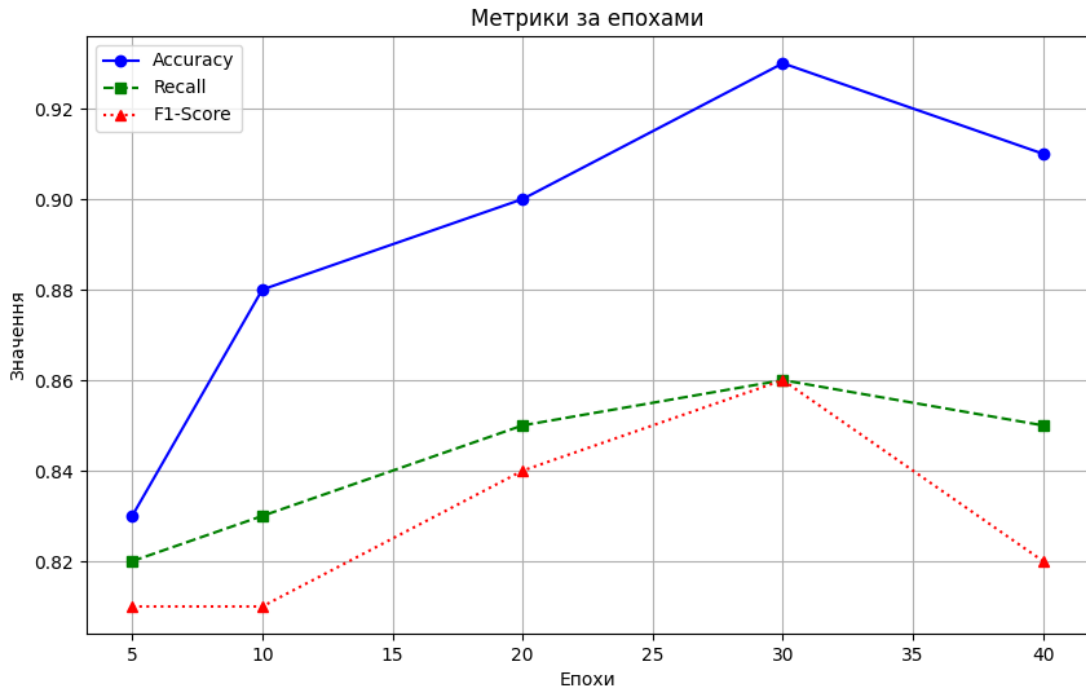


Рисунок 3.4 – Результати оцінки якості моделі

Recall вимірює спроможність моделі відшукати всі релевантні екземпляри даних true positives у класі. Тут також спостерігається підвищення recall зі збільшенням числа епох. Найвищий recall досягається на 30 епохах і становить 0.86, що означає, що модель правильно виявляє 86% екземплярів певного класу.

F1-показник - це гармонічне середнє між точністю і повнотою. Подібно до інших метрик, F1-показник також підвищується зі збільшенням числа епох. Найкращий результат F1-показник 0.86 також досягається на 30 епохах.

З 5 до 30 епох відбувається помітне покращення всіх метрик, що може свідчити про ефективне навчання моделі. Однак, після 30 епох Accuracy знижується, хоча recall і F1-показник залишаються високими. Це може вказувати на перенавчання моделі після 30 епох навчання. Найбільш оптимальним вибором є навчання моделі протягом 30 епох, оскільки на цьому етапі досягається хороший баланс між усіма метриками. Для підтримки кращої узагальнюючої здатності моделі слід уникати перенавчання, яке може відбутися при подовженні тривалості навчання після 30 епох.

При зростанні числа епох з 5 до 30 точність моделі значно покращується з 0.83 до 0.93, що свідчить про ефективність процесу навчання протягом цього періоду. Проте на 40 епохах точність знижується до 0.91. Це може бути знаком перенавчання моделі на навчальних даних після певної точки.

Подібно до точності, повнота збільшується зі зростанням числа епох з 0.82 до 0.86. Це означає, що модель стає кращою в ідентифікації позитивних випадків зі збільшенням кількості епох.

F1-показник також показує схожі тенденції з точністю і повнотою. Він збільшується з 0.1 при 5 епохах до 0.86 при 30 епохах. Після 30 епох F1-показник зменшується до 0.82 на 40 епохах, що може бути знаком недооцінки на валідаційних даних.

Зниження точності, recall і F1-показник після досягнення піку на 30 епохах може бути пов'язане з перенавчанням моделі. Модель може стати занадто витонченою під час тренування на навчальних даних, що впливає на її здатність узагальнювати на нових даних (наприклад, на валідаційному наборі).

Для досягнення кращої узагальнюючої здатності моделі рекомендується обирати оптимальну кількість епох, яка дозволяє досягти найвищих показників метрик на валідаційному наборі без перенавчання. Також може стати корисним застосування методів регуляризації або ранньої зупинки навчання для уникнення перенавчання.

Найкращий результат за метриками асигасу, recall і F1-показник показала модель з додатковим згортковим шаром. Вплив швидкості навчання відображається на швидкості збільшення метрик моделі під час навчання, де більше значення lr сприяє швидшій збірці метрик, але з моменту підходу до оптимальної точки знижується.

Це відображає вплив гіперпараметрів на якість моделі під час навчання і вказують на те, що додаткові згорткові шари можуть покращити якість моделі, а оптимізація швидкості навчання є ключовою для досягнення кращих результатів на тренувальних даних.

Порівняння моделей з різними гіперпараметрами. Модель з швидкість навчання $lr = 0.1$ досягла найвищої точності (ассигасу) після 30 епох тренування, однак метрики recall і F1-показник також показували стійке покращення з кожною епохою. Модель з меншим швидкість навчання $lr = 0.01$ показала добрі результати, але її збільшення в точності та інших метриках було менш помітним у порівнянні з моделлю з більшою швидкістю навчання. Модель з додатковим згортковим шаром показала кращі результати, особливо після 30 епох тренування, порівняно з базовою моделлю без цього шару. Вона досягла найвищих значень точності і F1-показник.

Вибір оптимальної швидкості навчання залежить від конкретних умов задачі. Загалом, більша швидкість навчання може пришвидшити збір метрик під час навчання, але може призвести до менш стабільної конвергенції. Менша швидкість навчання може потребувати більше епох для досягнення максимальної ефективності, але зазвичай має стабільніший процес навчання. Додавання додаткових шарів у цьому випадку згорткового шару може покращити якість моделі, особливо в задачах з багатошаровими даними або складними шаблонами. Однак це може збільшити складність моделі і час тренування.

При виборі оптимальних гіперпараметрів для моделі слід враховувати специфіку завдання, обсяг даних, а також бажані критерії ефективності моделі точність, recall, F1-показник. Експериментування з різними значеннями швидкості навчання і архітектурою моделі дозволить знайти оптимальне рішення для конкретного завдання машинного навчання.

Під час аналізу результатів можна помітити, що модель з додатковим згортковим шаром досягає найкращої стабільності та консистентності у всіх метриках протягом тренування на різних епохах. Це свідчить про те, що додавання додаткових шарів може допомогти знизити перенавчання та покращити загальну якість.

Крім швидкості навчання і архітектури моделі, варто також розглядати інші гіперпараметри, такі як кількість шарів, розмір пакету, метод оптимізації тощо. Ці фактори можуть також суттєво впливати на якість моделі.

На підставі аналізу наведених таблиць можна зробити висновок, що оптимальною конфігурацією для даної задачі може бути модель з більшою швидкістю навчання, наприклад, $lr = 0.1$ та з додатковим згортковим шаром. Ця конфігурація забезпечує баланс між швидкістю збірки метрик та стабільністю під час тренування.

В цілому, аналіз таблиць оцінки якості моделей дозволяє зробити висновки щодо впливу гіперпараметрів на якість моделі та визначити оптимальну конфігурацію для конкретної задачі машинного навчання. За отриманими результатами можна зробити наступні спостереження, щодо зміни швидкості навчання та кількості епох. Вибір оптимальної швидкості навчання та кількості епох під час тренування моделі може значно вплинути на її якість. Високі швидкості навчання можуть підвищувати точність, але можуть призводити до менш ефективного запам'ятовування. Збільшення кількості епох також може підвищити точність до певного рівня, але далі це може викликати перенавчання моделі.

При швидкості навчання 0.1 найкращі результати були отримані після 20 епох з точністю 0.88 і значенням F1 0.87. Збільшення числа епох після 20 призводить до зниження продуктивності.

Для швидкості навчання 0.01 найкращий результат був досягнутий на 30-й епохі з точністю 0.88 і значенням F1 0.88. Подальше збільшення числа епох після 30 не суттєво впливає на якість.

Таким чином, швидкість навчання 0.1 та 0.01 дозволяють досягти приблизно однакових значень Ассигасу і F1-показник після достатньої кількості епох близько 30-40. Модель з нижчою швидкістю навчання 0.01 має перевагу у високому значенні recall на ранніх етапах навчання до 20-ї епохи, але потім вже не покращується. Модель з більшою швидкістю навчання 0.1 досягає аналогічних значень F1-показник та ассигасу, але потребує трохи більше епох

для досягнення максимального потенціалу. Ці дані підкреслюють важливість вибору оптимальної швидкості навчання залежно від конкретної задачі і датасету, а також необхідність стежити за динамікою метрик під час навчання моделі.

Якщо важлива рання висока точність і повнота, то рекомендується використовувати швидкість навчання 0.01. Це дозволить моделі швидше наблизитися до бажаних результатів на ранніх етапах навчання. Якщо важлива максимальна точність і F1-показник, і може тривати навчання до 40 епох для досягнення максимального потенціалу моделі, то швидкість навчання 0.1 також може бути ефективним вибором.

Вибір оптимальної швидкості навчання залежить від конкретних вимог до моделі, обсягу даних, а також можливостей обчислювальних ресурсів.

Висновки до розділу 3

Додавання додаткових шарів до моделі може покращити її ефективність порівняно з базовою версією. Це пояснюється тим, що додаткові шари допомагають у вилученні більш складних ознак і категоризації, що зазвичай призводить до покращення точності, відтворюваності та загальної ефективності моделі.

Базова модель без додаткових шарів не показує високої ефективності, її максимальна точність становить лише 0.86. Це свідчить про те, що модель може бути недостатньо складною для взяття в розгляд усіх необхідних аспектів для точної класифікації.

Додавання повнозв'язного шару до базової моделі значно покращує її якість, досягаючи максимальної точності на рівні 0.89. Це підтверджує, що збільшення кількості параметрів, які можна навчити в моделі, сприяє кращому захопленню необхідних функцій для більш точної класифікації.

Додавання шару об'єднання до базової моделі покращує її загальну якість, хоча не настільки помітно, як додавання щільного шару. Об'єднання допомагає

моделі збирати більш деталізовані аспекти, хоча це може бути менш корисним у порівнянні з додатковими параметрами, які можна навчити.

Додавання згорткового шару до базової моделі додатково підвищує загальну якість з максимальною точністю 0.93. Це означає, що додаткові згорткові шари допомагають моделі засвоювати більш складні характеристики та покращують її загальну якість. Отже, додавання згорткового шару до базової моделі може значно підвищити її якість, що робить це однією з оптимальних конфігурацій параметрів.

Висновок

У кваліфікаційній роботі розроблено ефективний метод аналізу зображень рослин для ідентифікації хвороб, що дозволяє точно класифікувати зразки з хворобами та здоровими рослинами. Створено та навчено модель глибокого навчання, яка виявилась досить точною в класифікації хвороб за зображеннями, зокрема завдяки застосуванню передових методів обробки зображень та попередньої обробки даних.

Оцінено ефективність розробленої моделі на тестовому наборі даних, що підтвердило її високу точність і надійність у роботі з реальними зображеннями рослин.

Отже, розроблений метод і модель глибокого навчання успішно реалізовано для ідентифікації хвороб рослин за їх зображеннями з високим рівнем точності та надійності.

Модель на основі архітектури ResNet дозволяє ефективно виявляти та класифікувати хвороби листя рослин за зображеннями. Результати моделі показали високу якість розпізнавання на тестових зображеннях.

Оптимальний вибір гіперпараметрів, таких як швидкість навчання та кількість епох, дозволяє покращити загальну якість результатів. Найбільш ефективними виявилися швидкість 0.01 та 30-40 епох навчання, при яких модель досягає найкращих показників точності, повноти та F1-оцінки.

Збільшення архітектури моделі шляхом додавання згорткових та повністю зв'язаних шарів дозволяє підвищити потенціал моделі та покращити її здатність навчатися складних характеристик зображень.

При оптимізації гіперпараметрів та архітектури моделі важливо враховувати збалансований підхід для досягнення високої якості при збереженні обмежень на обчислювальні ресурси.

Загалом, запропонований метод CNN з результатами експериментів дозволяє ефективно виконувати задачу автоматичного розпізнавання хвороб рослин за зображеннями. Це створює передумови для розробки практичних

рішень у галузі сільського господарства. У цілому, було показано, що модель на основі архітектури ResNet дозволяє ефективно розпізнавати та класифікувати хвороби листя рослин проводячи точний аналіз зображень. Оптимізація архітектури та гіперпараметрів дозволяє поліпшити якість моделі. Розроблений підхід може бути застосований для автоматизації діагностики рослин в сільському господарстві.

Перелік посилань

1. Prabha K. Disease sniffing robots to apps fixing plant diseases: applications of artificial intelligence in plant pathology—a mini review. *Indian Phytopathology*. 2021. Vol. 74, No. 1. Pp. 13–20.
2. Sankaran S., Mishra A., Ehsani R., Davis C. A review of advanced techniques for detecting plant diseases. *Computers and electronics in agriculture*. 2010. Vol. 72, No. 1. Pp. 1–13.
3. Anand R., Mishra R. K., Khan R. Plant diseases detection using artificial intelligence: *Application of Machine Learning in Agriculture*. Elsevier, 2022.
4. Poornappriya T. S., Gopinath R. Rice plant disease identification using artificial intelligence approaches. *International Journal of Electrical Engineering and Technology*. 2020. Vol. 11, No. 10. Pp. 392–402.
5. Albattah W., Javed A., Nawaz M., Masood M., Albahli S. Artificial intelligence-based drone system for multiclass plant disease detection using an improved efficient convolutional neural network. *Frontiers in Plant Science*. 2022. Vol. 13. Pp. 808380.
6. Orchi H., Sadik M., Khaldoun M. On using artificial intelligence and the internet of things for crop disease detection: A contemporary survey. *Agriculture*. 2021. Vol. 12, No. 1. Pp. 9.
7. Yağ İ., Altan A. Artificial intelligence-based robust hybrid algorithm design and implementation for real-time detection of plant diseases in agricultural environments. *Biology*. 2022. Vol. 11, No. 12. Pp. 1732.
8. Ghorbanpour M., Omidvari M., Abbaszadeh-Dahaji P., Omidvar R., Kariman K. Mechanisms underlying the protective effects of beneficial fungi against plant diseases. *Biological Control*. 2018. Vol. 117. Pp. 147–157.
9. Koch E., Ole Becker J., Berg G., Hauschild R., Jehle J., Köhl J., Smalla K. Biocontrol of plant diseases is not an unsafe technology! *Journal of Plant Diseases and Protection*. 2018. Vol. 125. Pp. 121–125.

10. Thomas S., Kuska M. T., Bohnenkamp D., Brugger A., Alisaac E., Wahabzada M., Behmann J., Mahlein A.-K. Benefits of hyperspectral imaging for plant disease detection and plant protection: a technical perspective. *Journal of Plant Diseases and Protection*. 2018. Vol. 125. Pp. 5–20.
11. Devaraj A., Rathan K., Jaahnavi S., Indira K. Identification of plant disease using image processing technique: *2019 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 2019. Pp.0749–0753.
12. Iqbal Z., Khan M. A., Sharif M., Shah J. H., Rehman M. H. ur, Javed K. An automated detection and classification of citrus plant diseases using image processing techniques: A review. *Computers and electronics in agriculture*. 2018. Vol. 153. Pp. 12–32.
13. Liu J., Wang X. Plant diseases and pests detection based on deep learning: a review. *Plant Methods*. 2021. Vol. 17. Pp. 1–18.
14. Mohammed L., Yusoff Y. Detection and classification of plant leaf diseases using digital image processing methods: a review. *ASEAN Engineering Journal*. 2023. Vol. 13, No. 1. Pp. 1–9.
15. Vishnoi V. K., Kumar K., Kumar B. Plant disease detection using computational intelligence and image processing. *Journal of Plant Diseases and Protection*. 2021. Vol. 128. Pp. 19–53.
16. Abade A., Ferreira P. A., Barros Vidal F. de. Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*. 2021. Vol. 185. Pp. 106125.
17. Boulent J., Foucher S., Théau J., St-Charles P.-L. Convolutional neural networks for the automatic identification of plant diseases. *Frontiers in plant science*. 2019. Vol. 10. Pp. 464450.
18. Dhingra G., Kumar V., Joshi H. D. Study of digital image processing techniques for leaf disease detection and classification. *Multimedia Tools and Applications*. 2018. Vol. 77. Pp. 19951–20000.
19. Singh V., Sharma N., Singh S. A review of imaging techniques for plant disease detection. *Artificial Intelligence in Agriculture*. 2020. Vol. 4. Pp. 229–242.

20. Albattah W., Nawaz M., Javed A., Masood M., Albahli S. A novel deep learning method for detection and classification of plant diseases. *Complex & Intelligent Systems*. 2022. Vol. 8, No. 1. Pp. 507–524. URL: <https://doi.org/10.1007/s40747-021-00536-1>.
21. Ferentinos K. P. Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*. 2018. Vol. 145. Pp. 311–318. URL: <https://doi.org/10.1016/j.compag.2018.01.009>.
22. Saleem M. H., Potgieter J., Arif K. M. Plant disease detection and classification by deep learning. *Plants*. 2019. Vol. 8, No. 11. Pp. 468.
23. Too E. C., Yujian L., Njuki S., Yingchun L. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*. 2019. Vol. 161. Pp. 272–279. URL: <https://doi.org/10.1016/j.compag.2018.03.032>.
24. Mohameth F., Bingcai C., Sada K. A. Plant Disease Detection with Deep Learning and Feature Extraction Using Plant Village. *Journal of Computer and Communications*. 2020. Vol. 8, No. 6. Pp. 10–22. URL: <https://doi.org/10.4236/jcc.2020.86002>.
25. Hassan S. M., Maji A. K. Plant Disease Identification Using a Novel Convolutional Neural Network. *IEEE Access*. 2022. Vol. 10. Pp. 5390–5401. URL: <https://doi.org/10.1109/ACCESS.2022.3141371>.
26. Chen J., Zhang D., Suzauddola M., Nanekaran Y. A., Sun Y. Identification of plant disease images via a squeeze-and-excitation MobileNet model and twice transfer learning. *IET Image Processing*. 2021. Vol. 15, No. 5. Pp. 1115–1127. URL: <https://doi.org/10.1049/ipr2.12090>.
27. Kumar M. S., Ganesh D., Turukmane A. V., Batta U., Sayyadliyakat K. K. Deep Convolution Neural Network Based solution for Detecting Plant Diseases. *Journal of Pharmaceutical Negative Results*. 2022. Pp. 464–471. URL: <https://doi.org/10.47750/pnr.2022.13.S01.57>.
28. Shoaib M., Shah B., EI-Sappagh S., Ali A., Ullah A., Alenezi F., Gechev T., Hussain T., Ali F. An advanced deep learning models-based plant disease

detection: A review of recent research. *Frontiers in Plant Science*. 2023. Vol. 14. URL: <https://doi.org/10.3389/fpls.2023.1158933>.

29. Liu L., Ouyang W., Wang X., Fieguth P., Chen J., Liu X., Pietikäinen M. Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*. 2020. Vol. 128, No. 2. Pp. 261–318. URL: <https://doi.org/10.1007/s11263-019-01247-4>.

30. Padilla R., Netto S. L., Silva E. A. B. da. A Survey on Performance Metrics for Object-Detection Algorithms: 2020 *International Conference on Systems, Signals and Image Processing (IWSSIP)*, July 2020. Pp.237–242. URL: <https://doi.org/10.1109/IWSSIP48289.2020.9145130>.

31. Zhao Z.-Q., Zheng P., Xu S.-T., Wu X. Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*. 2019. Vol. 30, No. 11. Pp. 3212–3232. URL: <https://doi.org/10.1109/TNNLS.2018.2876865>.

32. Zou Z., Chen K., Shi Z., Guo Y., Ye J. Object Detection in 20 Years: A Survey. *Proceedings of the IEEE*. 2023. Vol. 111, No. 3. Pp. 257–276. URL: <https://doi.org/10.1109/JPROC.2023.3238524>.

33. A Survey of Deep Learning-Based Object Detection | IEEE Journals & Magazine | IEEE Xplore: URL: <https://ieeexplore.ieee.org/abstract/document/8825470>(дата звернення: 05.05.24).

34. Brahim M., Arsenovic M., Laraba S., Sladojevic S., Boukhalfa K., Moussaoui A. Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation: *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*: J. Zhou, F. Chen. Cham, Springer International Publishing, 2018. https://doi.org/10.1007/978-3-319-90403-0_6.

35. Guo Y., Zhang J., Yin C., Hu X., Zou Y., Xue Z., Wang W. Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming. *Discrete Dynamics in Nature and Society*. 2020. Vol. 2020. Pp. e2479172. URL: <https://doi.org/10.1155/2020/2479172>.

36. Jackulin C., Murugavalli S. A comprehensive review on detection of plant disease using machine learning and deep learning approaches. *Measurement: Sensors*. 2022. Vol. 24. Pp. 100441. URL: <https://doi.org/10.1016/j.measen.2022.100441>.
37. Madhulatha G., Ramadevi O. Recognition of Plant Diseases using Convolutional Neural Network: *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, October 2020. Pp.738–743. URL: <https://doi.org/10.1109/I-SMAC49090.2020.9243422>.
38. Alzubaidi L., Zhang J., Humaidi A. J., Al-Dujaili A., Duan Y., Al-Shamma O., Santamaría J., Fadhel M. A., Al-Amidie M., Farhan L. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*. 2021. Vol. 8, No. 1. Pp. 53. URL: <https://doi.org/10.1186/s40537-021-00444-8>.
39. Bhatt D., Patel C., Talsania H., Patel J., Vaghela R., Pandya S., Modi K., Ghayvat H. CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope. *Electronics*. 2021. Vol. 10, No. 20. Pp. 2470. URL: <https://doi.org/10.3390/electronics10202470>.
40. Hossain M. A., Sajib M. S. A. Classification of Image using Convolutional Neural Network (CNN). *Global Journal of Computer Science and Technology*. 2019. Vol. 19, No. D2. Pp. 13–18.
41. Kattenborn T., Leitloff J., Schiefer F., Hinz S. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2021. Vol. 173. Pp. 24–49. URL: <https://doi.org/10.1016/j.isprsjprs.2020.12.010>.

ДОДАТКИ

Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Валентин Соколовський,
студент 4 курсу, група КН-20-1

Хмельницький - 2024



Актуальність

- Автоматизація діагностики хвороб рослин підвищує ефективність аграрного сектору.
- Штучний інтелект забезпечує швидке виявлення хвороб, дозволяючи своєчасне.
- Ідентифікація хвороб за зображенням допомагає зменшити втрати врожаю.
- Використання машинного навчання покращує точність діагностики хвороб.
- Штучний інтелект у сільському господарстві підвищує якість та кількість врожаю.
- Своєчасне виявлення хвороб допомагає адаптуватися до змін клімату.



Об'єкт дослідження – процес ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Предмет дослідження – методи та технології машинного навчання для роботи із зображеннями

Мета кваліфікаційної роботи бакалавра полягає в покращенні якості ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Завдання кваліфікаційної роботи

- зробити аналіз методів аналізу набору зображень рослин з різними хворобами та здорових екземплярів;
- визначити послідовність застосування метода аналізу із застосуванням обробки зображень;
- сформулювати модель глибокого навчання для класифікації хвороб за зображеннями;
- здійснити тренування моделі з використанням навчального набору зображень для навчання моделі на виявлення хвороб;
- провести оцінювання ефективності моделі на тестовому наборі даних для визначення її точності та надійності.



Схематичне представлення архітектури моделі ResNet



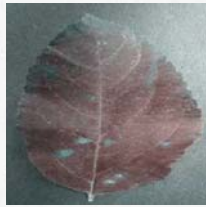
Архітектура мережі з коригуванням ваг





Зображення листя яблуні навчального набору даних з обертанням

Зображення листя яблуні з навчального набору даних (зеркальне відображення)



Зображення зі зміною яскравості

Розширення даних для визначення хвороб

Процес експерименту.

Крок 1: Вхідними даними для методу є набір зображень листя рослин із відповідним файлом анотацій, в яких містяться шляхи до зображень, координати обмежувальних рамок та класифікаційні мітки.

Крок 2: Базова модель виявлення об'єктів ResNet побудована на основі вагових коефіцієнтів. Додаткові шари додаються поверх базової архітектури моделі для адаптації її до виявлення хвороб листя рослин і класифікації зображень листя рослин.

Крок 3: Модель готується до навчання шляхом компіляції з використанням функції втрат середньоквадратичної помилки.

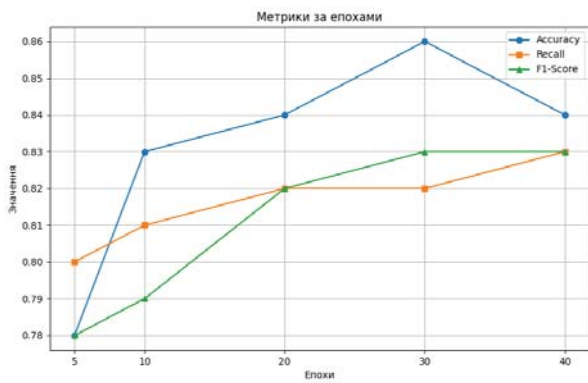
Крок 4: Під час навчання модель тренується за допомогою спеціального генератора, який створює пакети зображень і відповідні мітки. Процес навчання включає декілька епох, де кожна епоха включає подачу пакетів зображень до моделі, обчислення втрат і вдосконалення ваг моделі з використанням градієнтів.

Крок 5: Після навчання модель прогнозує мітки на основі тестових даних, і результати порівнюються з реальними мітками для обчислення показників оцінки, таких як втрати, точність, відтворюваність та оцінка F1.

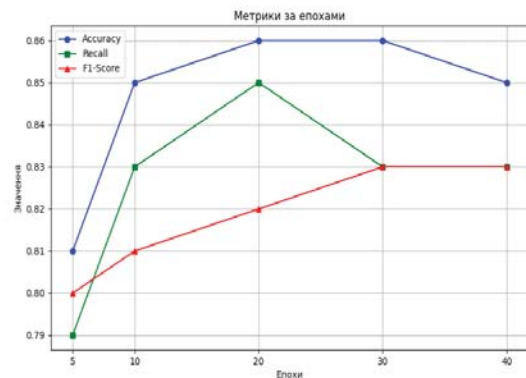




Результати оцінки якості моделі

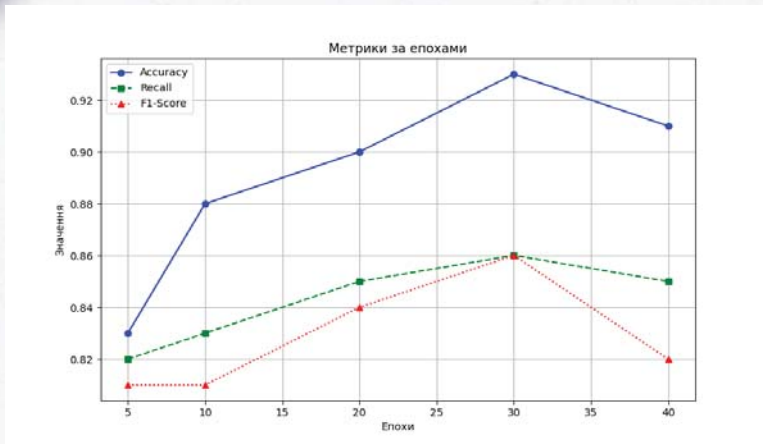


Швидкість навчання 0,1



Швидкість навчання 0,01

Експериментальна перевірка ефективності додавання додаткових шарів



Результати експериментів на наборі даних PlantVillage показали, що accuracy ідентифікації хвороб рослин для базової моделі становить 89,5%, а для модифікованої моделі - 93,2%. Recall розпізнавання хвороб для базової моделі - 88,1%, для модифікованої - 92,4%. F1-міра для базової моделі - 88,8%, для модифікованої - 92,8%.



Висновки

За результатами виконання кваліфікаційної роботи бакалавра було розроблено **метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту.**

- Проведено аналіз предметної області ідентифікації хвороб рослин за зображенням та розглянуто методи обробки зображень з використанням штучного інтелекту.
- Визначено оптимальні параметри, як кількість епох та швидкість навчання.
- Для покращення ефективності моделі запропоновано модифікацію архітектури шляхом додавання додаткових шарів.
- Проведено експериментальні дослідження ефективності розробленого методу. Базова модель без додаткових шарів не показує високої ефективності, її максимальна точність становить лише 0.86. Додавання згорткового шару до базової моделі додатково підвищує загальну якість з максимальною точністю 0.93. Це означає, що додаткові згорткові шари допомагають моделі засвоювати більш складні характеристики та покращують її загальну якість.

Таким чином, розроблений метод продемонстрував високу ефективність у задачі ідентифікації хвороб рослин за зображенням, що підтверджує досягнення мети кваліфікаційної роботи бакалавра.

Дякую за увагу



Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилки в документах: 7%**

ID: 130709 Назва: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту Додано в БД: 2024-06-15 Автора: Валентин СОКОЛОВСЬКИЙ Керівники: Едуард МАНЗІЮК Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	91192	1377	2313 (3%)	37 (3%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

Ім'я користувача:
Кафедра КН

ID перевірки:
1016363367

Дата перевірки:
15.06.2024 15:55:39 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
15.06.2024 16:00:51 EEST

ID користувача:
100005671

Назва документа: КН-20-1_Соколовський_ЗАПИСКА

Кількість сторінок: 73 Кількість слів: 14943 Кількість символів: 115252 Розмір файлу: 1.09 MB ID файлу: 1016168773

10.5% Схожість

Найбільша схожість: 1.29% з джерелом з Бібліотеки (ID файлу: 1016162677)

9.59% Джерела з Інтернету

884

Сторінка 75

3.81% Джерела з Бібліотеки

101

Сторінка 86

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

1

**РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованою системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

Автор: студент групи КН-20-1 Валентин СОКОЛОВСЬКИЙ

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: освітньо-професійна

Науковий керівник: д.т.н., професор кафедри Манзюк Е.А.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	<i>відповідає</i>
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі Валентина СОКОЛОВСЬКОГО, не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти програмного коду, що не мають авторства і містять поширені конструкції; серед запозичень знаходяться загальновідомі терміни, скорочення.

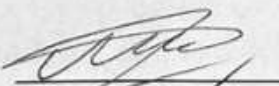
Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості, складає:

- за системою Anti-Plagiarism: 1%;

- за системою Unicheck: 10,5%.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості є допустимим.

Керівник роботи



Едуард МАНЗЮК

Гарант ОП



Олександр МАЗУРЕЦЬ

Завідувач кафедри КН



Олександр БАРМАК



ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук



**ВІДГУК НАУКОВОГО КЕРІВНИКА
на кваліфікаційну роботу бакалавра**

студента *гр. КН-20-1* Валентин СОКОЛОВСЬКИЙ

за темою Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

1. Актуальність теми

Тема дослідження, присвячена ідентифікації захворювань рослин за допомогою штучного інтелекту, є надзвичайно актуальною. Це обумовлено необхідністю підвищення ефективності аграрного сектору шляхом автоматизації процесів діагностики та моніторингу стану рослин. Сільське господарство постійно ризикує втратити врожай через швидке поширення хвороб, які важко виявити за звичайних умов. Інтеграція штучного інтелекту для аналізу зображень дозволить швидко і точно виявляти захворювання, що забезпечить своєчасні заходи для їх лікування та попередження подальшого розповсюдження.

2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки

Відповідно до встановлених стандартів, досліджувані об'єкти та сфера діяльності охоплюють математичні, інформаційні та симуляційні моделі реальних явищ, об'єктів, систем і процесів. Також включаються методи та технології для збору, зберігання, обробки, передачі та використання інформації. Основною метою роботи є створення методу для визначення захворювань рослин на основі зображень із застосуванням штучного інтелекту. Для досягнення цієї мети використовуються математичні моделі, методи та алгоритми, які дозволяють вирішувати як теоретичні, так і практичні завдання, пов'язані з розробкою методів машинного навчання. Результати цієї бакалаврської роботи відповідають стандартам спеціальності 122 – Комп'ютерні науки.

3. Професійні та особистісні якості бакалавра

Під час виконання бакалаврської роботи Валентин Соколовський проявив глибокі знання та високий рівень навичок, своєчасно виконуючи всі завдання. У процесі створення пояснювальної записки та розробки методу він показав свої професійні компетенції та

досягнення в навчанні. Валентин успішно оволодів професійними навичками у сфері "Комп'ютерні науки".

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Одержані в роботі результати є наслідком особистої діяльності студента, який самостійно виконував усі поставлені задачі.

5. Ступінь оволодіння методами дослідження

При реалізації кваліфікаційної роботи показав належний рівень компетентностей та володіння необхідними методами, методиками та технологіями предметної області комп'ютерних наук.

6. Повнота та якість розкриття теми роботи

Тема роботи повною мірою обґрунтована та розкрита належним чином. Проведено аналіз відомих досліджень відповідно до обраної теми. Поставлені завдання, реалізовані та розроблено програмне забезпечення для реалізації запропонованого метода.

7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу

Структура роботи та послідовність викладення логічні та відповідають поставленій меті. Викладення матеріалу послідовне, аргументоване, літературно грамотне.

8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин

Розроблений у роботі метод може бути використаний в автоматичних системах.

9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота

Враховуючи належний рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Керівник



д.т.н., професор каф. КН Едуард МАНЗЮК



РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра

студента *гр. КН-20-1 Валентина СОКОЛОВСЬКОГО*

за темою: Метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту

1. Актуальність обраної теми

Визначення захворювань рослин за допомогою штучного інтелекту відіграє ключову роль у сільському господарстві, оскільки ефективність та швидкість діагностики можуть суттєво впливати на врожайність і якість вирощених культур. Тому розробка систем штучного інтелекту для ідентифікації хвороб рослин є актуальним завданням.

2. Повнота розкриття мети та завдань роботи

Мета кваліфікаційної роботи бакалавра полягає в покращенні ідентифікації хвороб рослин за зображенням засобами штучного інтелекту. Для досягнення поставленої мети виконано наступні задачі дослідження: визначено послідовність застосування методу ідентифікації хвороб рослин за зображенням засобами штучного інтелекту; розроблено метод ідентифікації хвороб рослин за зображенням засобами штучного інтелекту; проведено експериментальні дослідження ефективності застосування розробленого методу.

3. Зміст кожного розділу роботи

Записка кваліфікаційної роботи бакалавра містить три розділи. У першому розділі проведено аналіз предметної області, досліджено відомі роботи та визначено актуальність теми. У другому розділі представлено метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту. Третій розділ присвячено експериментальній перевірці його ефективності.

4. Оцінка розробленої інформаційної системи, її практична цінність

Розроблений метод для ідентифікації хвороб рослин за зображенням засобами штучного інтелекту дозволяє ефективно ідентифікувати захворювання рослин за зображеннями, що має високу практичну цінність

5. Якість оформлення кваліфікаційної роботи бакалавра

Записка оформлена відповідно до вимог та правил. Викладення матеріалу логічне та послідовне.

6. Недоліки кваліфікаційної роботи бакалавра

Рекомендовано вдосконалити систему адаптуючи її до використання на мобільних пристроях.

7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Рецензент доц. каф. ІТБ Димитро О.В.