

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

До захисту допущено:

Завідувач кафедри

_____ Владислав ШЛИКОВ

«___» _____ 2023 р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Медична інженерія»

спеціальності 163 «Біомедична інженерія»

**на тему: «Аналіз захворювань шкіри на основі медичних зображень з
використанням машинного навчання»**

Виконала:

студентка IV курсу, групи БМ-91

Кузьо Іванна Сергіївна _____

Керівник:

доцент кафедри БМІ, к.т.н.

Білошицька Оксана Костянтинівна _____

Консультант з охорони праці:

доц. каф. ОППЦБ, к. т. н.

Демчук Гліб Вікторович _____

Рецензент:

доцент кафедри ББЗЛ, к.т.н.

Антонова-Рафі Юлія Валеріївна _____

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студентка _____

Київ – 2023 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет біомедичної інженерії
Кафедра біомедичної інженерії

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 163 «Біомедична інженерія»

Освітньо-професійна програма «Медична інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Владислав ШЛИКОВ

«__» _____ 20__ р.

ЗАВДАННЯ
на дипломну роботу студенту
Кузьо Іванні Сергіївні

1. Тема роботи «Аналіз захворювань шкіри на основі медичних зображень з використанням машинного навчання», керівник роботи Білошицька Оксана Костянтинівна, доцент кафедри БМІ, к.т.н., затверджена наказом по університету від «31» травня 2023 р. № 2106-с.
2. Термін подання студентом роботи «09» червня 2023 р.
3. Вихідні дані до роботи: патенти, статті та публікації в електронному форматі, державні та міжнародні стандарти, середовище розробки Google Colab, мова програмування Python, бібліотеки для роботи з даними.
4. Зміст роботи: літературний та патентний пошук моделей машинного навчання в медицині та дерматовенерології, аналіз методів в існуючих моделях, створення вимог до нової моделі, її розробка та тестування, оцінка результатів роботи моделі.
5. Перелік ілюстративного матеріалу: медичні зображення захворювань шкіри, презентація.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Демчук Г.В., доцент кафедри охорони праці, промисловості та цивільної безпеки		

7. Дата видачі завдання «5» квітня 2023 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Літературний та патентний пошук	Квітень-травень 2023 р.	
2	Вибір бази даних	Квітень 2023 р.	
3	Налаштування обробки зображень перед навчанням	Квітень 2023 р.	
4	Створення архітектури моделі	Квітень-травень 2023 р.	
5	Реалізація та тестування	Травень 2023 р.	
6	Оцінка ефективності моделі	Травень 2023 р.	
7	Оформлення розділу з охорони праці	Квітень-травень 2023 р.	
8	Оформлення дипломної роботи	Травень-Червень 2023 р.	
9	Оформлення документації	Травень-Червень 2023 р.	
10	Здача роботи на нормоконтроль	Червень 2023 р.	
11	Захист дипломної роботи	Червень 2023 р.	

Студентка

Іванна КУСЬО

Керівник

Оксана БІЛОШИЦЬКА

АНОТАЦІЯ

Обсяг звіту становить 70 сторінок, міститься 35 ілюстрацій, 3 формули, 13 таблиць та 2 додатки. Загалом опрацьовано 46 джерел.

Модель для діагностики захворювань шкіри на основі медичних зображень може бути використана в якості раннього інструменту скринінгу захворювань шкіри. Це особливо корисно для ідентифікації потенційних проблем з шкірою на ранніх стадіях, коли вони найбільш ефективно лікуються. Сфери можливого використання: домашній моніторинг, телемедицина, використання в медичних установах та наукових дослідженнях.

Актуальність даної роботи полягає у використанні комбінації CNN та LSTM мереж, що може сприяти діагностиці та лікуванню пацієнтів.

Мета роботи є підвищення точності диференціювання мальформацій шкіри на основі медичних зображень з використанням згорткових та рекурентних нейронних мереж.

Для досягнення мети виконуються наступні задачі:

1. Проведення літературного та патентного пошуку мальформацій шкіри та існуючих моделей машинного навчання.
2. Аналіз методів в існуючих моделях.
3. Формування вимог до нової моделі на основі аналізу існуючих.
4. Реалізація моделі з шарами CNN і RNN.
5. Оцінка ефективності роботи моделі.
6. Тестування моделі.

Ключові слова: машинне навчання, нейронна мережа, CNN, RNN, LSTM, модель.

ABSTRACT

The volume of the report is 70 pages, containing 35 illustrations, 3 formulas, 13 tables, and 2 appendixs. A total of 46 sources were studied.

A model for the diagnosis of skin diseases based on medical images can be used as an early screening tool for skin diseases. This is especially useful for identifying potential skin problems in their early stages when they are most effectively treated.

Areas of possible use: home monitoring, telemedicine, use in medical institutions, and scientific research.

This work's relevance lies in using a combination of CNN and LSTM networks, which can contribute to the diagnosis and treatment of patients.

The aim of the work is to increase the accuracy of differentiating skin malformations based on medical images using convolutional neural networks and recurrent neural networks.

To achieve the goal, the following tasks are performed:

1. Conducting a literature and patent search for skin malformations and existing machine learning models.
2. Analysis of methods in existing models.
3. Formation of requirements for the new model based on the analysis of the existing ones.
4. Model implementation with CNN and RNN layers.
5. Evaluation of the effectiveness of the model.
6. Model testing.

Keywords: machine learning, neural network, CNN, RNN, LSTM model

ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	8
ВСТУП	9
РОЗДІЛ 1 ЛІТЕРАТУРНИЙ ТА ПАТЕНТНИЙ ПОШУК	11
1.1. Мальформації шкіри.....	11
1.2. Моделі машинного навчання в медицині	16
1.3. Моделі машинного навчання в дерматовенерології	17
Висновки до розділу 1	20
РОЗДІЛ 2 АНАЛІЗ МЕТОДІВ В ІСНУЮЧИХ МОДЕЛЯХ	22
2.1 Згорткові нейронні мережі (CNN)	22
2.2 Рекурентні нейронні мережі	25
2.3 Інструменти та бібліотеки	27
Висновки до розділу 2	28
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА	29
3.1 Формування основних вимог до нової моделі.....	29
3.2 Реалізація моделі	31
3.2 Результати навчання моделі.....	35
3.3 Тестування моделі.....	38
Висновки до розділу 3	45
РОЗДІЛ 4 ОХОРОНА ПРАЦІ	46
4.1 Характеристика приміщення.	46

					БМ91.16.3105.2106.ПЗ							
Вим	Лист	№ докум.	Підпис	Дата								
Розробив	Кусьо І.С.				Аналіз захворювань шкіри на основі медичних зображень з використанням машинного навчання				Літ.	Лист	Листів	
Перевірів	Білошицька О.К.										6	
Реценз.	Антонова-Рафі								КПІ ім. Ігоря Сікорського ФБМІ БМ-91			
Н. Контр.	Андреев П.І.											
Затвердив	Шликов В.В.											

4.2 Оцінка небезпечних та шкідливих факторів	48
4.2.1 Електронебезпека.....	49
4.2.2 Пожежна безпека.....	50
4.2.3 Мікроклімат	51
4.2.4 Біологічні фактори	52
Висновки до розділу 4	53
ВИСНОВКИ.....	54
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	55
ДОДАТОК А.....	60
ДОДАТОК Б	69

СПИСОК СКОРОЧЕНЬ

CNN – Згорткова нейронна мережа

RNN – Рекурентна нейронна мережа

LSTM – Мережа довгої короткочасної пам'яті

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		8

ВСТУП

Як найбільший орган тіла, шкіра має широкий спектр аномалій, які можуть виникнути. Існує близько 1500 різних видів шкірних захворювань. Переважна більшість цих захворювань мають симптоми, з якими люди не знайомі, але це ускладнює їхню точну ідентифікацію для дерматологів [1]. Шкіра виконує низку важливих функцій, зокрема захист від зовнішніх впливів та регуляцію температури. Захворювання шкіри можуть суттєво позначитися на якості життя людини, викликаючи дискомфорт, психологічне незадоволення та медичні проблеми.

Однак, діагностика захворювань шкіри може бути складною задачею та займати багато часу. Традиційні методи діагностики, такі як клінічний огляд та інструментальні дослідження, часто вимагають великого досвіду та експертної кваліфікації. Водночас, з огляду на зростання обсягу медичних даних та доступність цифрових зображень, з'являється потреба у розробці ефективних та автоматизованих методів діагностики захворювань шкіри.

В останні роки спостерігається швидкий прогрес у галузі медичного штучного інтелекту, машинного навчання та глибокого навчання. Виявлення різних шкірних захворювань вимагає як фізичних ознак, і аналізу зображень. Використовуючи аналіз зображень, дерматологи можуть з більшою точністю виявляти проблеми зі шкірою. Застосування методів машинного навчання для діагностики захворювань шкіри на основі медичних зображень відкриває нові можливості для швидкої, недорогої та надійної оцінки стану шкіри та вчасного виявлення різних хвороб.

Тема дипломної роботи: Аналіз захворювань шкіри на основі медичних зображень з використанням машинного навчання.

Мета дипломної роботи: Підвищення точності диференціювання мальформацій шкіри на основі медичних зображень з використанням

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		9

згорткових та рекурентних нейронних мереж.

Задачі дипломної роботи:

1. Проведення літературного та патентного пошуку мальформацій шкіри та існуючих моделей машинного навчання.
2. Аналіз методів в існуючих моделях.
3. Формування вимог до нової моделі на основі аналізу існуючих.
4. Реалізація моделі з шарами CNN і RNN.
5. Оцінка ефективності роботи моделі.
6. Тестування моделі.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		
						10

РОЗДІЛ 1

ЛІТЕРАТУРНИЙ ТА ПАТЕНТНИЙ ПОШУК

1.1. Мальформації шкіри

Мальформації шкіри — це вади шкіри, які можуть проявлятися у вигляді аномальних шрамів, пігментації, вузлів, розширених кровоносних судин, а також дефектів, пов'язаних з розміром, формою або положенням певних областей шкіри [2]. Вони можуть бути наслідком генетичних аномалій, зовнішніх впливів, або їх причина може залишатися невідомою.

Мальформації шкіри можуть бути ізольованими, вражаючи лише шкіру, або вони можуть бути частиною широкого спектра синдромів, які включають аномалії інших органів та систем.

Наприклад, актинічний кератоз (рис. 1.1) є найпоширенішим передраковим ураженням, яке утворюється на шкірі, пошкодженій хронічним впливом ультрафіолетових (УФ) променів від сонця та/або засмаги [3].



Рисунок 1.1 – Актинічний кератоз [4]

Захворювання часто проявляється у вигляді невеликих сухих, лускатих або покритих скоринкою ділянок шкіри. Вони можуть бути червоними, світло- або темно-коричневими, білими, рожевими, тілесного відтінку або поєднанням кольорів, іноді піднятими.

Незважаючи на те, що лише 5-10% актинічних кератозів перетворюються на рак шкіри, переважна більшість плоскоклітинних

карцином починаються як актинічний кератоз. Ось чому рання діагностика є дуже важливою.

Себорейний кератоз (рис. 1.2) — це поширене доброякісне новоутворення шкіри, схоже на родимку [5]. Вони нешкідливі і не потребують лікування, але їх можна видалити, якщо вони турбують пацієнта. Це плями округлої або овальної форми на шкірі, іноді їх описують як воскові або лускаті. Вони підняті над шкірою, і навіть якщо вони плоскі, їх можна відчутти пальцем. Зазвичай вони коричневі, але також можуть бути чорними або коричневими, рідше рожевими, жовтими або білими.



Рисунок 1.2 – Себорейний кератоз [4]

Якщо багато себорейних кератозів раптово спалахують разом, це може викликати певне занепокоєння. Це незвичайне явище іноді вважалося ознакою внутрішнього раку. Кореляція ще не доведена і може бути лише випадковим збігом. Але це може давати підстави для додаткової діагностики пацієнта.

Меланоцитарний невус (рис. 1.3) — це ще одне поширене доброякісне ураження шкіри [6]. Коричневий або чорний меланоцитарний невус містить пігмент меланін, тому його також можна назвати пігментним невусом. Меланоцитарний невус може бути присутнім при народженні (вроджений меланоцитарний невус) або з'явитися пізніше (набутий невус).



Рисунок 1.3 – Меланоцитарний невус [4]

Кількість меланоцитарних невусів у людини залежить від генетичних факторів, впливу сонця та імунного статусу.

Люди турбуються про свої родимки, тому що чули про меланому, злоякісну проліферацію меланоцитів, яка є найпоширенішою причиною смерті від раку шкіри. Спочатку меланома може виглядати схожою на нешкідливий меланоцитарний невус, але з часом вона стає більш невідповідною за структурою та має тенденцію до збільшення. Люди з більшою кількістю невусів мають вищий ризик розвитку меланоми, ніж ті, у кого їх мало, особливо якщо їх більше 100. Отже, меланоцитарні невуси потребують медичної оцінки та спостереження.

Дійсно небезпечною для людини є меланома (рис. 1.4). Меланома, найсерйозніший вид раку шкіри, розвивається в клітинах (меланоцитах), які виробляють меланін — пігмент, який надає шкірі колір [7]. Меланома також може утворюватися в очах і, рідко, всередині тіла людини, наприклад у носі або горлі. Точна причина всіх меланом не ясна, але вплив ультрафіолетового (УФ) випромінювання від сонячного світла або солярію підвищує ризик розвитку меланоми. Меланому можна успішно лікувати, якщо її виявити на ранній стадії.



Рисунок 1.4 – Меланома [4]

Базальноклітинна карцинома (рис.1.5) або базаліома є найпоширенішою формою раку шкіри та найпоширенішою формою з усіх видів раку [8]. Базаліоми виникають внаслідок аномального, неконтрольованого росту

базальних клітин. Базаліоми можуть виглядати як відкриті виразки, червоні плями, рожеві нарости, блискучі горбки, шрами або нарости з дещо піднятими, загорнутими краями та/або центральним поглибленням.



Рисунок 1.5 – Базальноклітинна карцинома [4]

За відсутності лікування базаліоми можуть проростати широко та глибоко в шкіру та руйнувати шкіру, тканини та кістки. Бувають дуже незвичайні, агресивні випадки, коли захворювання поширюється на інші частини тіла. Тому дуже важливо вчасно діагностувати захворювання і розпочати лікування пацієнта.

Дерматофіброми (рис.1.6) в основному зустрічаються у дорослих.



Рисунок 1.6 – Дерматофіброма [4]

Неясно, чи є дерматофіброма реактивним процесом чи справжнім новоутворенням. Іноді їх пов'язують із незначною травмою, зокрема укусами комах або ін'єкціями. Множинні дерматофіброми можуть розвинути у пацієнтів зі зміненим імунітетом, таким як ВІЛ, імуносупресія або аутоімунні захворювання [9].

Дерматофіброма зазвичай проявляється як одиночна тверда папула або вузлик на кінцівці. Розмір дерматофіброми коливається від 0,5 до 1,5 см в діаметрі; більшість пошкоджень мають діаметр 7–10 мм. Колір може бути від рожевого до світло-коричневого для білої шкіри та від темно-коричневого до чорного для темної шкіри; деякі виглядають блідіше в центрі.

Дерматофіброма не викликає раку. Але через постійне травмування може викликати інші проблеми зі здоров'ям.

Червонуваті або багрянні плями кровоносних судин, які лопнули під шкірою (наприклад, судинні вени, гемангіоми), називаються судинними ураженнями [10] (рис. 1.7). Деякі вади розвитку вен є вродженими або присутніми при народженні. Інші розвиваються і стають видимими десь між дитинством і дорослим життям.



Рисунок 1.7 – Судинні ураження шкіри [4]

Судинні ураження можуть бути будь-якого розміру, форми та інтенсивності кольору. Розташовані на обличчі, шії, голові та інших ділянках тіла, які важко приховати.

Судинні ураження шкіри представляють широкий спектр станів, від доброякісних до потенційно загрозливих для життя. Хоча більшість судинних уражень доброякісні, деякі можуть бути злоякісними та метастазувати в інші частини тіла.

В цілому, всі ці стани вимагають медичної оцінки та потенційного лікування, щоб зменшити ризик серйозних наслідків.

1.2. Моделі машинного навчання в медицині

Патент «Метод розробки моделі машинного навчання нейронної мережі для класифікації медичних зображень» [11] описує систему машинного навчання для аналізу медичних даних та прогнозування станів пацієнтів. Одним з ключових принципів роботи цієї системи є постійне навчання на нових даних, що дозволяє системі постійно оновлюватися та поліпшувати свої прогнози з часом. Однак, в системі є певні недоліки: зокрема, недостатня кількість даних може призвести до неточності результатів, а також відсутність інтеграції з іншими медичними системами може вимагати введення даних у систему виключно вручну.

Патент «Медичні системи та методи глибокого навчання для реконструкції зображення та оцінки якості» [12] описує систему, яка використовує нейронні мережі для автоматизації процесу діагностики та передбачення хвороб. Система навчається на медичних даних для виявлення пухлин, оцінки ступеню захворювання та передбачення результатів лікування. Одним з недоліків цієї системи є потреба у великій кількості медичних даних для навчання системи ефективного аналізу медичних зображень. Точність системи може залежати від якості вхідних даних та може бути обмежена здатністю нейронних мереж адаптуватися до нових типів захворювань або зображень.

Патент «Глибоке навчання медичних систем і методів медичних процедур» [13] використовує нейронні мережі, зокрема, глибокі згорткові мережі (Convolutional Neural Networks, CNN), рекурентні нейронні мережі (Recurrent Neural Networks, RNN) та автокодувальні мережі (Autoencoder Networks). Унікальність патенту полягає у використанні глибоких згорткових мереж для обробки медичних даних та автоматичної діагностики різних захворювань. При цьому мережі можуть бути навчені за допомогою невеликої

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		16

кількості зображень, що робить їх застосування більш ефективним у порівнянні з іншими методами. Недоліком цього патенту може бути відносно малий обсяг даних для навчання мереж, що може призвести до недостатньої точності діагностики при застосуванні до нових даних.

Патент «Медичні системи та методи глибокого навчання для отримання зображень» [14] описує методи та системи машинного навчання для обробки медичних даних. У патенті описано використання різних алгоритмів машинного навчання, таких як нейронні мережі, SVM (support vector machine), ансамблі дерев рішень та інші. Недоліками цього патенту можуть бути високі вимоги до обчислювальної потужності, необхідність великої кількості навчальних даних та складнощі у підтримці та налагодженні системи.

Патент «Системи та методи обробки медичних даних і машинного навчання для анатомічної оцінки» [15] описує методи та системи машинного навчання, які дозволяють розпізнавати зміни в зображеннях медичних даних та забезпечувати автоматизовану оцінку стану пацієнта. Було використано різні типи нейронних мереж, зокрема згорткові мережі та рекурентні мережі. Принцип роботи системи полягає у використанні нейронних мереж, які навчаються на великому обсязі медичних зображень для автоматичного виявлення патологічних змін у нових зображеннях. Система автоматично аналізує зображення та відзначає будь-які зміни, які можуть бути показниками патології. Серед недоліків патенту можна виділити високу складність процесу навчання нейронної мережі, а також високі вимоги до обчислювальної потужності для роботи з великим обсягом медичних даних.

1.3. Моделі машинного навчання в дерматовенерології

Дослідники вивчали корисність штучного інтелекту для покращення або доповнення поточних процесів скринінгу меланоми та немеланомного раку

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		
						17

шкіри. Наср–Есфакані та ін. [16] були першими, хто навчив нейронну мережу виявленню меланоми, а запропонований ними метод мав чутливість і специфічність 0,81 і 0,80 відповідно.

У 2017 році Стенфордський університет опублікував дослідження [17] про глибоке вивчення пухлин шкіри. Було продемонстровано класифікацію уражень шкіри за допомогою єдиної згорткової нейронної мережі, навченої наскрізно із зображень безпосередньо, використовуючи лише пікселі та мітки захворювання як вхідні дані. Згорткову нейронну мережу було навчено, використовуючи набір даних із 129 450 клінічних зображень 2032 різних захворювань. Ефективність оцінювалась у порівнянні з 21 сертифікованим дерматологом на клінічних зображеннях, підтверджених біопсією. Було виявлено, що апарат має компетенцію, яку можна порівняти з сертифікованими дерматологами в ідентифікації та класифікації раку шкіри.

Однак зовнішня достовірність цієї роботи невизначена, оскільки не було включено демографічну інформацію. Іншим обмеженням дослідження було те, що, кількість навчальних зображень, необхідних для такої системи надзвичайно велика.

Патент «Система і метод діагностики уражень і аномалій шкіри і тканин» [18] описує комп'ютеризовану систему та метод діагностики уражень шкіри. Використовуючи методи аналізу зображень, інформацію з анкети, інформацію з профілю користувача та, за бажанням, додаткову зовнішню інформацію, система встановлює діагноз ураження.

У патенті пропонується використання різних типів нейронних мереж, зокрема, згорткових нейронних мереж (Convolutional Neural Networks), рекурентних нейронних мереж (Recurrent Neural Networks) та глибоких нейронних мереж (Deep Neural Networks). Крім того, використовуються методи попередньої обробки та аугментації даних для поліпшення якості навчання нейронної мережі.

Одним з можливих недоліків даного патенту може бути складність налагодження та оптимізації системи, а також необхідність великої кількості даних для навчання нейронної мережі, що може бути складно забезпечити в деяких випадках.

Автори статті «Automated Skin Disease Identification Using Deep Learning Algorithm» [19] описують розроблення автоматизованої системи ідентифікації захворювань шкіри на основі алгоритму глибокого навчання. Дослідження проводилося на основі бази даних різних захворювань шкіри.

Система використовує три загальнодоступні архітектури розпізнавання зображень, а саме Inception V3, Inception Resnet V2, Mobile Net з модифікаціями для застосування шкірних захворювань і успішно прогнозує захворювання шкіри на основі максимального голосування з трьох мереж. Після попередньої обробки зображень, а саме зменшення розміру та нормалізації, зображення піддаються аналізу мережею. Після навчання мережі, система може визначати захворювання шкіри за зображеннями.

Одним з можливих застосувань системи є її використання для автоматичної діагностики та ідентифікації захворювань шкіри в реальному часі. Також система може використовуватися для навчання студентів медичних факультетів та лікарів - дерматологів.

До недоліків системи можна віднести те, що для навчання мережі було використано базу даних, що складається з великої кількості зображень з високою якістю, що може не відображати реальний стан речей в клінічній практиці. Крім того, навчання мережі виконувалося на обмеженій кількості типів захворювань шкіри, що може призвести до меншої точності визначення рідкісних захворювань. Також, для покращення роботи системи потрібне додаткове навчання на нових даних.

У статті «A Method Of Skin Disease Detection Using Image Processing And Machine Learning» [20] описано метод виявлення захворювань шкіри за допомогою обробки зображень та методів машинного навчання. Всю

архітектуру можна розділити на кілька модулів, що включають попередню обробку, виділення ознак і класифікацію.

У роботі використовували зображення з різних веб-сайтів, присвячених шкірним захворюванням. База даних має 80 зображень (20 звичайних зображень, 20 зображень меланоми, 20 зображень екземи та 20 зображень псоріазу). Для покращення точності діагностики, автори використали методи навчання глибоких нейронних мереж (deep learning), зокрема, архітектуру Convolutional Neural Network (CNN). Автори стверджують, що система успішно виявляє 3 різні типи шкірних захворювань з точністю 100%.

Недоліком цієї моделі є мала кількість даних для навчання, тому не можна однозначно оцінити її точність та ефективність.

Останнім часом мобільні застосунки для смартфонів стали доступними для діагностики меланоми [21]. Якщо вони ефективні, вони потенційно пропонують миттєву оцінку ризику ймовірності злоякісності, щоб люди зверталися за додатковою медичною допомогою до лікаря для більш детальної оцінки ураження. Однак існує ризик того, що меланоми будуть пропущені, а лікування відкладено, якщо програма запевнить користувача, що їхнє ураження є низьким.

Чутливість для цих застосунків коливається від 7% до 73%, а специфічність коливається від 37% до 94%. Можна зробити висновок, що в поточному стані ці програми потенційно можуть пропустити меланоми та є потенційно небезпечними, оскільки можуть вселити в користувача помилкове відчуття безпеки.

Висновки до розділу 1

У даному розділі було описано найбільш поширені мальформації шкіри, а також моделі на основі нейронних мереж для діагностики цих захворювань.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		
						20

Було визначено, що в моделях для діагностики захворювань шкіри використовують згорткові нейронні мережі та рекурентні нейронні мережі.

Однак, у розробці можуть виникнути певні проблемні фактори:

- наразі даних про різні захворювання шкіри недостатньо, рівень обміну інформацією між джерелами низький, а якість зображень шкіри неоднакова;
- необхідність великої кількості даних для навчання нейронної мережі;
- діагностика штучним інтелектом включає правові проблеми, етичні проблемита проблеми конфіденційності даних;
- діагностика та класифікація шкірних захворювань вимагають комплексного врахування анамнезу пацієнта, статі, віку та іншої інформації, крім фотографій. Тож у майбутньому буде потрібна інтеграція клінічних даних із цією інформацією.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		21

РОЗДІЛ 2

АНАЛІЗ МЕТОДІВ В ІСНУЮЧИХ МОДЕЛЯХ

2.1 Згорткові нейронні мережі (CNN)

Один з найпоширеніших підходів – це використання згорткових нейронних мереж (CNN).

Згорткова нейронна мережа (CNN) - це варіант моделі нейронної мережі, який був особливо розроблений для обробки двовимірних зображень [22], хоча він також може бути застосований для одно- і тривимірних даних. Ідея згорткового шару, який становить основу CNN, полягає в застосуванні операції, відомої як "згортка".

«Згортка» в контексті CNN – це лінійна дія, що включає перемноження ваги на вхідні дані, подібно до того, як це відбувається в звичайній нейронній мережі. Отже, існуючи в рамках традиційних нейронних мереж, згорткові мережі спеціалізуються на обробці більш складних даних, таких як зображення. На рисунку 2.1 зображено принцип роботи CNN.

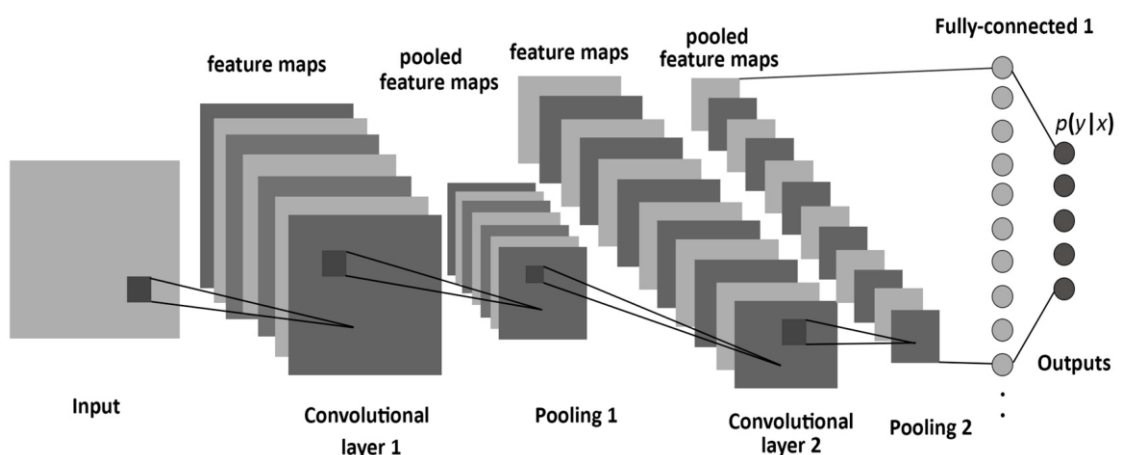


Рисунок 2.1 – Принцип роботи CNN [23]

Враховуючи, що метод був розроблений для двовимірного введення, множення виконується між масивом вхідних даних та двовимірним масивом ваг, званим фільтром або ядром.

У даному контексті "фільтр" є меншим, ніж самі вхідні дані, що дозволяє працювати з конкретними областями вхідних даних за один раз. Це виконується через процес, який називається скалярним добутком, що є типом множення, застосовуваного між фрагментом вхідних даних, який має розмір фільтра, та самим фільтром. Скалярний добуток – це поелементне множення між фрагментом вхідних даних розміром з фільтр і фільтром, яке підсумовується, що завжди призводить до одного значення [24]. Оскільки в результаті виходить одне значення, операцію часто називають скалярним добутком.

Стосовно фільтра зменшеного розміру у відношенні до вхідних даних – це навмисне рішення, яке дає змогу множинними способами пересувати однаковий фільтр (набір ваг) через вхідний масив, це відбувається в різних точках входу. У певному сенсі, цей фільтр систематично проходить через всі відповідні ділянки вхідних даних, рухаючись від лівої до правої сторони та зверху вниз. Ця стратегія дозволяє фільтру виявити конкретний об'єкт у будь-якому місці зображення, якщо цей фільтр був спеціально розроблений для виявлення такого об'єкту. Це властивість відома як трансляційна інваріантність [25], що означає фокус на виявленні присутності функції, а не її місцезнаходження.

Кожен раз, коли фільтр перемножується на вхідний масив, отримується одне значення. Але оскільки фільтр застосовується до вхідного масиву кілька разів, в результаті ми отримуємо двовимірний масив вихідних значень – це є відфільтрований вхід. Тому, цей двовимірний масив, що виходить з процесу, називається картою об'єктів.

Після формування карти об'єктів, можна обробити кожне значення на карті через нелінійну функцію, так, як це зазвичай відбувається з вихідними

даними у повнозв'язних шарах. Технічно згортка, описана під час використання згорткових нейронних мереж, насправді є «взаємною кореляцією» [26]. Тим не менш, у глибокому навчанні це називається операцією «згортки».

Таким чином, є вхідні дані, такі як зображення значень пікселів, і є фільтр, який є набором ваг, і фільтр систематично застосовується до вхідних даних для створення карти об'єктів [27].

Проте, використання лише CNN може бути обмеженим у діагностиці захворювань шкіри на основі медичних зображень з декількох причин:

- моделі, навчені на даних одного джерела, можуть показувати менш ефективні результати на даних з інших джерел;
- моделі, навчені на одному типі захворювання, можуть не бути ефективними для діагностики інших типів захворювань шкіри.

Можна навести приклад практичної перевірки одного з недоліків. Було знайдено навчальну модель нейронної мережі з використанням згорткових шарів (CNN) яка попередньо навчена на діагностику певних захворювань шкіри, таких як меланома, кератоз або карцинома [28]. Моделі було надано зображення шкіри з підозрою на акне. У цьому випадку, модель вивела ймовірності наявності хвороб, на які вона була навчена, не звертаючи увагу на те, що зображення відноситься до іншого типу захворювання шкіри (рис. 2.2).

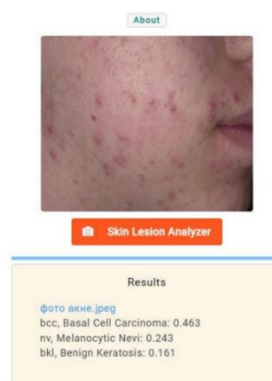


Рисунок 2.2 – Результат роботи моделі для діагностики меланом, карцином та кератозу

Для того, щоб зменшити можливі недоліки моделі CNN, можна виконати додатковий аналіз зображення, що виводиться. Цей аналіз може включати в себе різноманітні обробки та перевірки зображення, наприклад, класифікацію типів захворювань шкіри. Якщо навчальна модель не може знайти збіги з хворобами, на які вона була навчена, то можна вивести повідомлення про це.

2.2 Рекурентні нейронні мережі(RNN)

Рекурентні нейронні мережі (RNN) зазвичай використовуються для аналізу послідовностей даних.

Рекурентна нейронна мережа (рис. 2.3) – це тип алгоритму, орієнтованого на глибоке навчання, який слідує послідовному підходу [29]. У нейронних мережах завжди припускається, що кожен вхід і вихід залежить від інших верств. Ці типи нейронних мереж називаються рекурентними, тому що вони послідовно виконують математичні обчислення.

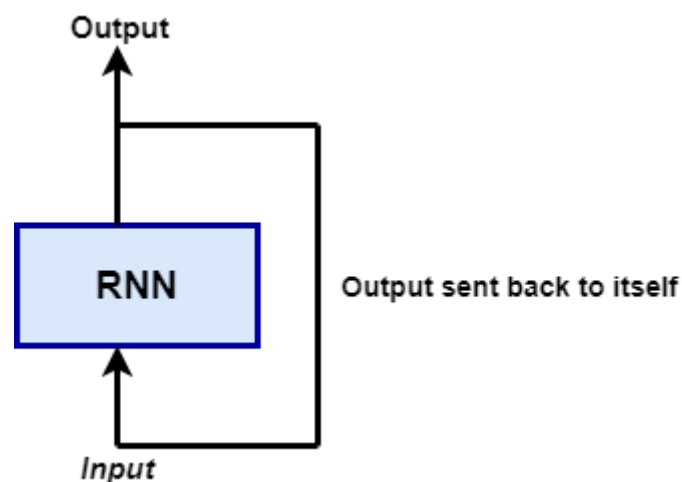


Рисунок 2.3 – Принцип роботи RNN [29]

RNN можуть бути корисними для аналізу інформації про розвиток захворювань шкіри в часі.

Теоретично, можна використовувати лише RNN для діагностики шкіри, адже RNN може аналізувати послідовність зображень і виявляти зміни в

пікселях, що можуть свідчити про розвиток хвороби. Однак, такий підхід може бути менш ефективним порівняно з CNN, які спеціалізуються на роботі з зображеннями.

Є деякі приклади використання RNN для аналізу медичних зображень, але вони в основному пов'язані з аналізом динаміки змін в зображеннях в часі (наприклад, зміна форми органу під час його функціонування) [30].

У контексті діагностики захворювань шкіри, можливо використати RNN для аналізу динаміки змін в розмірі і формі деяких дефектів, наприклад, для виявлення зміни розміру пухлини протягом часу. Проте, в цілому, CNN є більш популярним і ефективним підходом для роботи з медичними зображеннями.

Варто також згадати про LSTM (мережа довгої короткострокової пам'яті). Це різновид рекурентних нейронних мереж (RNN), які здатні вивчати довгострокові залежності, особливо в задачах прогнозування послідовності [31]. LSTM має зв'язку зі зворотним зв'язком, тобто здатна обробляти всю послідовність даних, крім окремих точок даних. LSTM – це особливий вид RNN, який показує видатну продуктивність під час вирішення великої кількості завдань.

Основна відмінність між RNN та LSTM полягає в тому, що інформація зберігається в пам'яті протягом тривалого часу. Тут LSTM має перевагу перед RNN [32].

RNN мають петлі зворотного зв'язку в рекурентному шарі. Це дозволяє їм зберігати інформацію в пам'яті з часом. Але може бути складно навчити стандартні RNN для вирішення завдань, які потребують вивчення довгострокових залежностей.

Блоки LSTM включають «осередок пам'яті», яка може зберігати інформацію в пам'яті протягом тривалих періодів часу. Цей осередок пам'яті дозволяє їм вивчати довгострокові залежності.

Отже, використання LSTM можна вважати кращим способом відслідковувати певні послідовності.

Також є моделі, які порівнюють роботу CNN так LSTM [32]. Але саме комбінація цих моделей може допомогти в покращенні точності діагностики захворювань шкіри на основі медичних зображень, оскільки вона поєднує переваги як CNN, так і RNN, і дозволяє використовувати інформацію з послідовностей ознак, що може бути важливим при діагностиці захворювань шкіри.

2.3 Інструменти та бібліотеки

Для реалізації моделей машинного навчання зазвичай використовують наступні інструменти та бібліотеки:

- Google Colab - середовище розробки та виконання коду [33];
- PyTorch: бібліотека машинного навчання, яка надає широкі можливості для побудови та навчання моделей глибокого навчання [34];
- OpenCV: бібліотека комп'ютерного зору, яка використовується для обробки зображень [35];
- NumPy: бібліотека для роботи з масивами та обчисленнями в наукових областях, включаючи обробку зображень [36];
- Pandas: бібліотека для аналізу та маніпулювання даними, використовується для роботи з даними з файлу CSV [37];
- Matplotlib: бібліотека для візуалізації даних, використовується для створення графіків та діаграм [38].

Висновки до розділу 2

У даному розділі було описано найбільш поширені класи нейронних мереж, а також інструменти та бібліотеки, що зазвичай використовуються для діагностики захворювань шкіри. Було виявлено, що використання певної однієї нейронної мережі не дасть достовірних результатів діагностики.

Зокрема, CNN зазвичай використовується для визначення характерних ознак на зображеннях, таких як форма, текстура, контур та інші, і не враховує послідовність зображень в часі.

З іншого боку, RNN може працювати з послідовністю зображень та розпізнавати зміни в хворобі з часом, але це може бути дуже складно, оскільки медичні зображення шкіри можуть бути дуже великими та важкими для обробки. LSTM має перевагу перед RNN, оскільки здатна зберігати інформацію в пам'яті на довший термін.

Отже, поєднання CNN та LSTM може забезпечити більш точну діагностику шкіри.

РОЗДІЛ 3

ПРАКТИЧНА ЧАСТИНА

3.1 Формування основних вимог до нової моделі

В попередньому розділі було визначено, що для розробки нової моделі варто використовувати комбінацію нейронних мереж, а саме згорткову та LSTM мережі. Також, спираючись на недоліки моделей, які були розглянуті у попередніх розділах, можна сформувати основні вимоги до нової моделі:

1. Модель повинна бути навчена на великій кількості якісних фото захворювань шкіри. Для роботи було знайдено базу даних [39], що налічує близько 10015 дерматоскопічних зображень, які випущено як навчальний набір для цілей машинного навчання. Випадки включають репрезентативну колекцію всіх важливих діагностичних категорій у сфері пігментних уражень. Понад 50% уражень було підтверджено патологією, тоді як для решти випадків було або подальше спостереження консенсус експертів, або підтвердження конфокальною мікроскопією *in vivo*. База даних включає зображення таких станів шкіри людини: меланоцитарні невуси, меланома, доброякісний кератоз, базаліома, актинічний(сонячний) кератоз, судинні ураження шкіри та дерматофіброма.

2. Повинна бути попередня обробка даних. Перед використанням зображень для навчання моделі потрібно провести попередню обробку, наприклад змінити розмір зображень, виконати нормалізацію. Також, для коректного навчання моделі всі фото мають бути розподілені на три категорії: навчальні, валідаційні і тестові (техніка Train-Valid-Test) [40].

Навчальна категорія – набір даних, які використовуються для навчання (моделлю), тобто для підгонки параметрів до моделі машинного навчання.

Валідаційна категорія – набір даних, який використовується для надання неупередженої оцінки моделі, підігнаної до навчального набору даних під час налаштування гіперпараметрів моделі.

Тестова категорія – набір даних, який використовується для забезпечення неупередженої оцінки остаточної моделі, підігнаної до навчального набору даних.

3. Модель повинна мати визначену архітектуру. Модель повинна включати комбінацію CNN і LSTM для ефективної обробки зображень шкіри та аналізу послідовностей. CNN використовується для виявлення характерних ознак на зображеннях, тоді як LSTM забезпечує аналіз змін в часі та послідовностях зображень.

4. Після розробки повинна бути оцінена ефективність моделі. Після навчання моделі, її ефективність повинна бути перевірена за допомогою валідаційних даних та оцінена за допомогою метрик, таких як точність (precision), повнота (recall) та F-міра.

Для оцінки метрик використовуються такі змінні [41]:

- TP (True Positive) – кількість правильно класифікованих позитивних зразків (захворювання шкіри в даному випадку);
- TN (True Negative) – кількість правильно класифікованих негативних зразків (здорова шкіра);
- FP (False Positive) – кількість неправильно класифікованих позитивних зразків (хибно-позитивні результати);
- FN (False Negative) – кількість неправильно класифікованих негативних зразків (хибно-негативні результати).

Точність (precision) в контексті моделі визначається як відсоток коректно визначених випадків з усіх, які модель віднесла до конкретного класу

(формула 3.1). Тобто, це частка випадків, що дійсно належать визначеному класу відносно всіх, що були віднесені моделлю до цього класу.

$$Precision = \frac{TP}{TP+FR} \quad (3.1)$$

Повнота (recall) є мірою істинно позитивних класифікацій. Вона демонструє, яку частину об'єктів, які насправді входять в позитивний клас, модель здатна правильно визначити (формула 3.2):

$$Recall = \frac{TP}{TP+FN} \quad (3.2)$$

F1 показник (f1-score) – це гармонічне середнє між точністю і повнотою. Він забезпечує збалансовану оцінку ефективності моделі (формула 3.3):

$$F = \frac{2 \times precision \times recall}{precision + recall} \quad (3.3)$$

5. Модель повинна бути експлуатаційно простотою. Розроблена модель повинна бути легкою у використанні для лікарів та медичного персоналу. Інтерфейс користувача повинен бути зрозумілим та інтуїтивно зрозумілим, щоб спростити процес діагностики та зробити його доступним для фахівців з різним рівнем.

6. Модель повинна надавати достовірні та точні результати, які відповідають дійсному стану захворювання шкіри на основі медичних зображень.

7. Модель повинна бути швидкою та ефективною, щоб забезпечити швидку діагностику та рішення щодо захворювань шкіри.

Загалом, очікуваний від роботи результат – точна, ефективна та надійна система, яка може допомогти у ранньому виявленні та класифікації захворювань шкіри, сприяючи швидкому та ефективному лікуванню.

3.2 Реалізація моделі

Основні кроки, виконані під час реалізації моделі наведені на рисунку 3.1.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		31

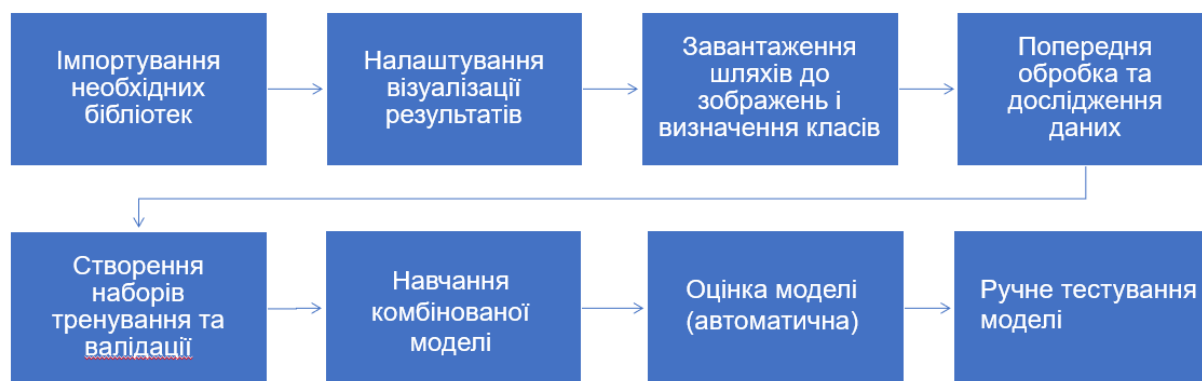


Рисунок 3.1 – Кроки реалізації моделі

Спочатку було проведено підготовку даних для навчання моделі. В якості бази зображень було використано «The HAM10000 dataset» [42]. Проте, цей набір включав розподіл фотографій лише на дві вибірки: тренувальну та валідаційну. Тому, для безпосереднього тестування моделі тестовий набір фото був зібраний вручну.

Зображення завантажувалися з використанням бібліотеки OpenCV, а метадані про класи захворювань були збережені у файлі CSV. На основі цих метаданих, модель обчислює середнє значення та стандартне відхилення каналів RGB для всіх зображень для подальшої нормалізації. Результат завантаження зображень можна побачити на рисунку 3.2.

	lesion_id	image_id	dx	dx_type	age	sex	localization	path	cell_type	cell_type_idx
0	HAM_0000118	ISIC_0027419	bkl	histo	80.0	male	scalp	gdrive/MyDrive/Диплом/HAM10000_images_part_1/I...	Benign keratosis-like lesions	2
1	HAM_0000118	ISIC_0025030	bkl	histo	80.0	male	scalp	gdrive/MyDrive/Диплом/HAM10000_images_part_1/I...	Benign keratosis-like lesions	2
2	HAM_0002730	ISIC_0026769	bkl	histo	80.0	male	scalp	gdrive/MyDrive/Диплом/HAM10000_images_part_1/I...	Benign keratosis-like lesions	2
3	HAM_0002730	ISIC_0025661	bkl	histo	80.0	male	scalp	gdrive/MyDrive/Диплом/HAM10000_images_part_1/I...	Benign keratosis-like lesions	2
4	HAM_0001466	ISIC_0031633	bkl	histo	75.0	male	ear	gdrive/MyDrive/Диплом/HAM10000_images_part_2/I...	Benign keratosis-like lesions	2

Рисунок 3.2 – Результат завантаження даних з CSV файлу

Кодування діагнозів здійснено наступним чином:

‘nv’: ‘Melanocytic nevi’ – Меланоцитарні невуси

‘mel’: ‘Melanoma’ – Меланома,

‘bkl’: ‘Benign keratosis-like lesions’ – Доброякісні ураження, схожі на кератоз,

‘bcc’: ‘Basal cell carcinoma’ – Базальноклітинна карцинома,

‘akiec’: ‘Actinic keratoses’ - Актинічний кератоз,
‘vasc’: ‘Vascular lesions’ – Ураження судин,
‘df’: ‘Dermatofibroma’ – Дерматофіброма.

У файлі CSV класи захворювань були представлені у вигляді текстових міток. Для полегшення класифікації класи було перекодовано у числовий формат, де кожному класу було присвоєно унікальне числове значення.

Нижче представлено розшифровку міток захворювань та числові значення які були присвоєні кожному захворюванню (рис. 3.3).

cell_type_idx	cell_type
0	Actinic keratoses
1	Basal cell carcinoma
2	Benign keratosis-like lesions
3	Dermatofibroma
4	Melanocytic nevi
5	Melanoma
6	Vascular lesions

Рисунок 3.3 –Числові значення які були присвоєні кожному захворюванню

Зображення шкіри було піддано попередній обробці та нормалізації. Зображення було зменшено до однакового розміру та перетворено у векторизовані, щоб їх можна було використовувати для навчання моделі.

Розмір зображення змінюється до певного розміру (input_size). Потім застосовуються випадкові горизонтальні та вертикальні перевороти, випадкове обертання та перетворення тремтіння кольорів, щоб збільшити навчальні дані.

Потім модель виконує розпізнавання дублікатів серед зображень і відокремлює унікальні зображення. Ці унікальні зображення потім розподіляються на набори для тренування і валідації. Набір для валідації формується так, щоб він мав пропорційний розподіл класів, як і вихідний датасет.

Далі модель балансує набір даних для тренування. Це означає, що класи з меншою кількістю зображень копіюються до більшої кількості, щоб уникнути упередженості під час тренування.

Нарешті, набори даних для тренування та валідації обробляються і готові для подальшого використання в навчанні моделі.

Для побудови моделі було використано два основні типи нейронних мереж: CNN та LSTM. Архітектура моделі зображена на рисунку 3.4.

```
class PretrainedResNextLSTM(nn.Module):
    def __init__(self, num_class=7, hidden_size=128, num_layers=1):
        super().__init__()
        self.resNext = models.resnext101_32x8d(pretrained=True)
        self.channels = self.resNext.fc.in_features # Оновлено тут
        for params in self.resNext.parameters():
            params.requires_grad_(False)
        self.features = nn.Sequential(*list(self.resNext.children())[:-1])
        self.lstm = nn.LSTM(self.channels + 1, hidden_size, num_layers, batch_first=True)
        self.fc1 = nn.Linear(hidden_size, num_class)
        self.softmax = nn.Softmax(dim=1)

    def forward(self, x, previous_diagnosis):
        x = self.features(x)
        x = torch.mean(x, dim=(2, 3))
        x_and_previous_diagnosis = torch.cat([x, previous_diagnosis.view(-1, 1)], dim=1)
        x = x_and_previous_diagnosis.unsqueeze(0)
        lstm_out, _ = self.lstm(x)
        x = self.fc1(lstm_out)
        out = self.softmax(x)
        return out.squeeze(0)
```

Рисунок 3.4 – Архітектура моделі

Специфічно, тут використовується ResNext101_32x8d. ResNext [43] – це тип згорткової нейронної мережі, що включає в себе модулі з «наступними» з'єднаннями, що покращують процес навчання. Зображення вводяться в модель, і вихід моделі потім використовується як вхідний вектор для LSTM-мережі.

LSTM використовується для обробки вихідних даних від CNN разом з попередніми діагнозами. Вихід LSTM потім передається в повністю з'єднаний шар для остаточної класифікації.

Цей підхід, який поєднує CNN та LSTM, дозволяє моделі використовувати як просторову інформацію з зображень, так і послідовність попередніх діагнозів для передбачення типу клітини. Кількість шарів та їх параметри були обрані на основі експериментів та оптимізації моделі.

Вихід з LSTM-модуля передається в FC-шар для виконання остаточної класифікації. У цьому випадку, FC-шар має таку ж кількість вихідних одиниць, як кількість класів у задачі класифікації (`num_class`).

Після FC-шару, вихід пропускається через `softmax`-функцію, яка перетворює вихід моделі в ймовірності для кожного класу. Це допомагає при інтерпретації результатів моделі.

Для навчання моделі використовується алгоритм оптимізації Adam і функція втрат. Оптимізатор використовується для оновлення параметрів моделі на основі градієнтів, вирахованих з функції втрат. Функція втрат, широко використовується для задач класифікації, оскільки вона враховує ймовірність кожного класу, а не лише мітку класу. Функція втрати визначає, як модель оцінює свою точність, а оптимізатор визначає, як модель навчається на основі вхідних даних та втрати. Було створено оптимізатор Adam, де `model.parameters()` передає параметри моделі для оптимізації, а `lr=1e-3` встановлює швидкість навчання рівною 0,001.

Модель була навчена на тренувальних даних. Для цього були використані тренувальний та валідаційний набір зображень та їх мітки.

Повний код моделі наведено у Додатку А.

3.3 Результати навчання моделі

По завершенню роботи з моделлю було отримано інформацію про ефективність даної моделі, а також деякі інші графіки.

Після навчання модель будує два графіки (рис. 3.5). Перший показує динаміку значень втрат і точності моделі під час навчання для кожної епохи. Другий графік показує динаміку втрат і точності на валідаційному наборі даних за кожну епоху. Ці графіки допомагають у підтвердженні того, чи модель вчиться правильно, і дозволяють виявити, чи модель перенавчилася

(overfitting), коли точність на тренувальному наборі продовжує зростати, а на валідаційному – знижується.

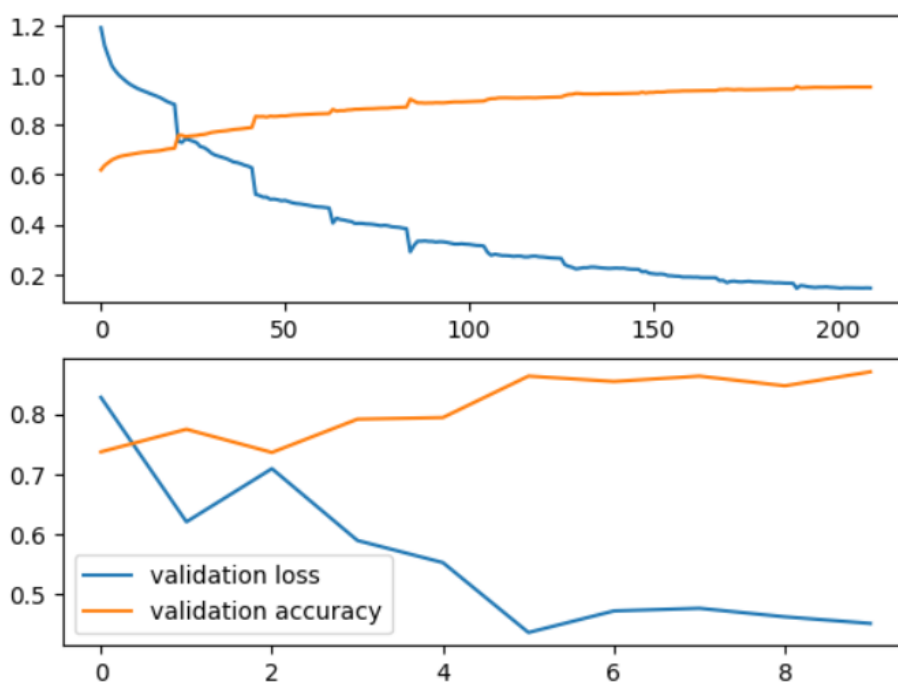


Рисунок 3.5 – Динаміка значень втрат і точності на тренувальному та валідаційному наборі даних

З наведеного графіку можна зробити висновок, що перенавчання моделі не відбулося, оскільки точність на валідаційному наборі збільшується.

Оцінки метрик, таких як точність (precision), повнота (recall), f1 показник (f1 score) та підтримка (support), допомогли оцінити ефективність моделі в розпізнаванні та класифікації захворювань шкіри. Отримано показники, як для кожного класу, так і загальні (рис. 3.6).

	precision	recall	f1-score	support
akiec	0.52	0.70	0.59	30
bcc	0.74	0.83	0.78	35
bkl	0.80	0.73	0.76	88
df	0.70	0.70	0.70	8
nv	0.99	0.94	0.96	883
vasc	0.38	0.79	0.52	46
mel	1.00	0.77	0.87	13
accuracy				1103
macro avg	0.73	0.78	0.74	1103
weighted avg	0.94	0.91	0.92	1103

Рисунок 3.6 – Метрики оцінки ефективності моделі

Загальні показники:

- «ассигасу» (Точність): Вказує на загальну точність моделі, яка розраховується як відношення кількості правильно класифікованих екземплярів до загальної кількості екземплярів [44]. У даному випадку, модель правильно класифікувала 91% екземпляр в тестовому наборі даних;
- «масго avg» (Середнє значення): Представляє середні значення показників (precision, recall, f1-score) для всіх класів, обчислені без залежності від розміру класу [45];
- «weighted avg» (Зважене середнє значення): Представляє середні значення показників (precision, recall, f1-score) для всіх класів, обчислені з урахуванням розміру класу [46].

Загальною тенденцією є те, що модель має високі показники точності, повноти і f1-показника для класу "nv" (Меланоцитарні невуси), що свідчить про її ефективність у класифікації меланоцитарних невусів. Однак, модель має нижчі показники для деяких інших класів. Для покращення ефективності моделі можна звернути увагу на ці класи та розглянути можливі покращення у тренуванні моделі для цих класів або збільшення обсягу даних цих класів для кращого навчання моделі.

Ще один спосіб оцінки моделі – матриця помилок (рис. 3.7).

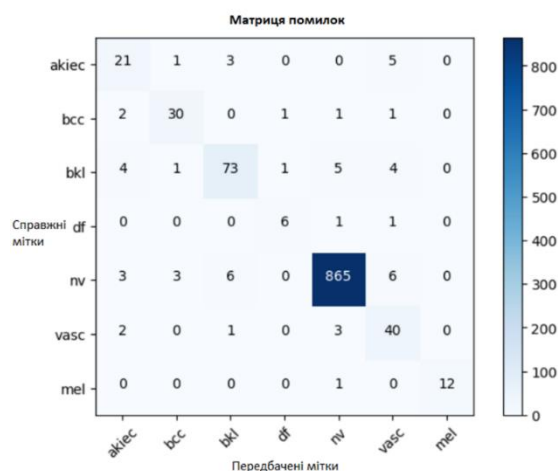


Рисунок 3.7 – Матриця помилок

Матриця помилок дозволила визначити, які класи були правильно класифіковані, а які призвели до помилок. Чим більше значень на головній діагоналі (від верхнього лівого до нижнього правого кута), тим краще. Ці значення відповідають кількості правильно класифікованих екземплярів для кожного класу. Значення поза головною діагоналлю вказують на помилки.

3.4 Тестування моделі

Для тестування моделі було створено окрему функцію `test_model(model, image_path, transform, user_id)`. Дана функція наведена у Додатку Б. Вона приймає набір параметрів: модель для тестування, шлях до зображення, яке має бути аналізоване, трансформацію, яка має бути застосована до зображення для підготовки його до вхідних даних моделі, та ідентифікатор користувача, який використовується для зв'язку з даними, що зберігаються.

Функція `test_model` починається з встановлення моделі в режим оцінки, що означає, що вона готова прийняти вхідні дані та виробляти передбачення. Зображення відкривається, трансформується і перетворюється на тензор для подачі в модель.

Зображення потім передається моделі, яка виробляє вихідні дані, що відображають передбачувану ймовірність кожної категорії діагнозу. Проходиться максимізація по осі та отримується індекс найвищого значення, який відповідає передбаченому діагнозу.

Функція `analyze_and_compare_diagnosis(predicted_label, predicted_prob, user_id)` використовується для порівняння нового діагнозу з останнім записаним діагнозом у файлі CSV.

Дані записуються в CSV файл. Кожен новий діагноз, отриманий в результаті аналізу фотографії, буде записано в цей файл. Приклад записаних даних у файл зображено на рисунку 3.8.

← 1_diagnoses.csv			
	A	B	C
1	datetime	diagnosis	probability
2	2023-06-04 0:01:37	Dermatofibroma	99.89808202

Рисунок 3.8 – Збережені дані пацієнта

Є кілька сценаріїв, які могли б зустрітися при аналізі діагнозів:

1. Якщо попередній діагноз існує (тобто це не перше завантаження фотографії), то новий діагноз порівнюється з попереднім.

— Якщо ймовірність нового діагнозу вища за попередній, виводиться повідомлення «Стан пацієнта покращився» (рис. 3.9 – 3.14).



Рисунок 3.9 – Тестування покращення актинічного кератозу: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.10 – Тестування покращення базаліоми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.11 – Тестування покращення судинних уражень шкіри: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.12 – Тестування покращення дерматофіброми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі

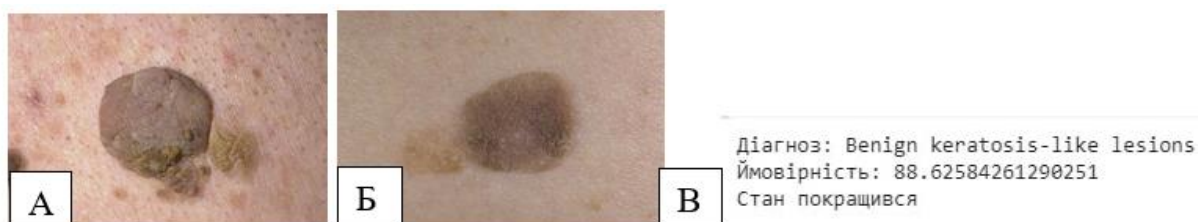


Рисунок 3.13 – Тестування покращення доброякісних кератозів: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.14 – Тестування покращення меланоми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.15 – Тестування покращення меланоцитарних невусів: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі

Якщо ймовірність нового діагнозу нижча за попередній, виводиться повідомлення «Стан пацієнта погіршився» (рис. 3.16– 3.22).



Рисунок 3.16 – Тестування погіршення актинічного кератозу: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.17 – Тестування погіршення базаліоми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.18 – Тестування погіршення судинних уражень шкіри: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.19 – Тестування погіршення дерматофіброми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.20– Тестування погіршення доброякісних кератозів: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі

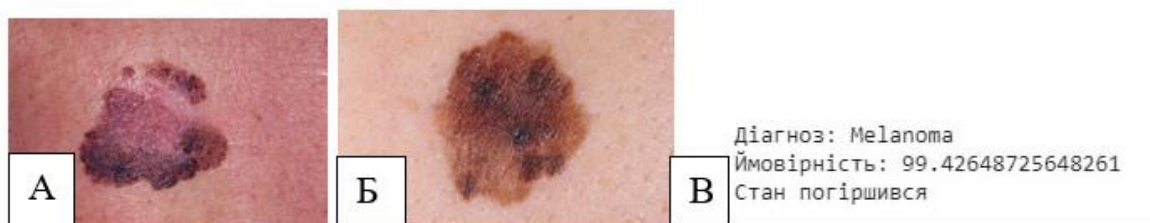


Рисунок 3.21 – Тестування погіршення меланоми: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі



Рисунок 3.22 – Тестування погіршення меланоцитарних невусів: А – попереднє завантажене зображення; Б – наступне завантажене зображення; В – результат роботи моделі

Якщо ймовірність нового діагнозу дорівнює попередньому, виводиться повідомлення «Стан пацієнта залишився незмінним». Такий варіант можливий якщо помилково було завантажено одне й те саме фото.

2. Якщо попереднього діагнозу не існує (тобто це перше завантаження фотографії):

— Якщо ймовірність нового діагнозу менша за визначений поріг, виводиться повідомлення «Модель не здатна проаналізувати ваш стан шкіри або ви здорові». В цьому випадку новий діагноз не записується в CSV файл (рис. 3.23). Цей сценарій був створений для того, щоб модель не виводила помилкові діагнози для захворювань, на які не була навчена, і для здорової шкіри.

```
# Аналізуємо фотографію шкіри користувача
image_path = 'herpes1.jpg'
user_id = 4
threshold = 60
predicted_label, predicted_prob = test_model(model, image_path, val_transform, user_id)
health_status = analyze_and_compare_diagnosis(predicted_label, predicted_prob, user_id)

# Виводимо результати
print(f"Діагноз: {predicted_label}")
print(f"Ймовірність: {predicted_prob}")
print(health_status)
```

Модель не здатна діагностувати ваш тип захворювання або ви здорові

Рисунок 3.23 – Приклад запобігання помилкового діагнозу

— Якщо ймовірність нового діагнозу перевищує визначений поріг, новий діагноз записується в CSV файл. Виводиться повідомлення «Це перше завантаження фото» (рис. 3.24).

```
# Аналізуємо фотографію шкіри користувача
image_path = 'dermatofibroma-1.jpg'
user_id = 1
threshold = 60
predicted_label, predicted_prob = test_model(model, image_path, val_transform, user_id)
health_status = analyze_and_compare_diagnosis(predicted_label, predicted_prob, user_id)

# Виводимо результати
print(f"Діагноз: {predicted_label}")
print(f"Ймовірність: {predicted_prob}")
print(health_status)
```

Діагноз: Dermatofibroma
Ймовірність: 99.99550580978394
Це перше завантаження фото

Рисунок 3.24 – Вивід моделі при першому завантаженні зображення

Цей процес дозволяє відслідковувати динаміку стану хвороби користувача за допомогою моделі та зберігати історію діагнозів для подальшого аналізу.

На завершення, результати діагнозу та стан здоров'я виводяться на екран. Крім того, користувачу надається інструкція про те, як завантажити фото для аналізу (рис. 3.25).

```
[ ] # Надаємо інструкції користувачу
print("Завантажте фото ділянки шкіри, зроблене на відстані від 5 до 20 см.")
```

Завантажте фото ділянки шкіри, зроблене на відстані від 5 до 20 см.

Рисунок 3.25 – Інструкція для користувача

Ця інструкція важлива, оскільки відстань між камерою та ділянкою шкіри, яку фотографує користувач, може суттєво вплинути на якість отриманого зображення, а отже, і на точність діагнозу, який видає модель.

Якщо фотографія буде зроблена на надто великій або надто малій відстані, це може призвести до різноманітних проблем. Занадто близька відстань може викликати розмиття або спотворення деталей шкіри, що ускладнить виявлення ознак захворювання. Занадто велика відстань може

призвести до того, що потрібні деталі будуть недостатньо видимі для аналізу моделлю.

Таким чином, якщо користувач завантажить фото, зроблене на відстані, що не відповідає рекомендованому діапазону, це може вплинути на точність виявлення захворювань шкіри моделлю і в результаті вона може показати надто низьку ймовірність, пропустити захворювання або навіть видає невірний діагноз. Отже, ця інструкція важлива для забезпечення найкращих умов для аналізу моделлю.

Висновки до розділу 3

У даному розділі було описано основні вимоги до нової моделі, а також наведено результати її реалізації.

В процесі реалізації моделі було виконано наступні дії: імпортування необхідних бібліотек, налаштування візуалізації результатів, завантаження шляхів до зображень і визначення класі, попередня обробка та дослідження даних, створення наборів тренування та валідації, навчання комбінованої моделі, оцінка ефективності моделі (автоматична), ручне тестування моделі.

Створена модель досягла точності в 91%, а також показала хороші результати в ручному тестуванні з фотографіями з інших джерел. Модель враховує різні сценарії, які можуть відбуватися при аналізі зображень, а також здатна відслідковувати зміни в часі. Крім того, в моделі прописана інструкція для користувача щодо завантаження зображення.

Всі вимоги до моделі були виконані.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		45

РОЗДІЛ 4

ОХОРОНА ПРАЦІ

4.1 Характеристика приміщення

Розроблену модель пропонується застосовувати в медичному кабінеті.

Медичний кабінет представляє собою просторе, освітлене приміщення з антибактеріальним лінолеумом на підлозі. Стіни приміщення покриті водостійкою фарбою світлих відтінків, що сприяє збільшенню рівня освітлення. Приміщення оснащено природним та штучним освітленням, створюючи комфортні умови для праці в будь-який час доби.

Медичний кабінет має природну вентиляцію, а також може бути обладнаний вентиляційною системою для підтримання оптимального клімату. Меблі та обладнання розміщені таким чином, щоб забезпечити достатньо простору для безпечного руху і роботи.

План приміщення зображений на рисунку 4.1, параметри кабінету та перелік обладнання і оснащення наведено в таблиці 4.1.

Таблиця 4.1 – Характеристика медичного кабінету. Параметри приміщення, обладнання і оснащення.

№	Найменування	Основні характеристики	Кіль- сть	Позиція на рисунку
1	Параметри всього кабінету	5000×5000×3000 мм; S=25 м ² ; V=75 м ³	—	—
2	Кількість працюючих	Лікар-дерматовенеролог	1	—
3	Природне освітлення	Пластикове вікно з 2-х частин Rehau, 1200 х 1400 мм	1	—
4	Штучне освітлення	Світильник ЛПО-01	1	—

Продовження таблиці 4.1

№	Найменування	Основні характеристики	Кіль- сть	Позиція на рисунок
5	Ноутбук HP 250 G8 (45S01ES)	Розміри: 358×242×20 мм Температура: 0...70 °C Матеріал: метал, пластик	1	1
6	Стіл письмовий двотумбовий ПС-2100	Розміри: 750×1500×600 мм Матеріал: дерево	1	2
7	Стілець офісний ISO	Розміри: 530×435×475 мм Матеріал: тканина, метал	1	3
8	Стелаж для архіву SA039499	Розміри: 2200×1200×600 мм Матеріал: метал	1	4
9	Раковина підвісна STATUS GLOV	Розміри: 600×400×120 мм Матеріал: кераміка	1	5
10	Кушетка медична Амед К – 1.0	Розміри: 1980×600×500 мм Матеріал: сталь, шкірозамінник	1	6
11	Фотоапарат Canon EOS 4000D BK 18-55	Розміри: 101×129×77 мм Матеріал: метал, пластик	1	7
12	Тумба мобільна Roko БЮ301	Розміри: 406×584×423 мм Матеріал: дерево	1	8
13	Порошкові вогнегасники, ВП-5	Розміри: Ø160 мм, висота – 505 мм Матеріал: метал	2	9
14	Протипожежна система ДИП-1	Димова	1	10

В таблиці 4.2 наведено порівняльну характеристику основних нормативних значень медичного кабінету з реальними значеннями.

Таблиця 4.2 – Порівняння нормативних параметрів

№	Параметр кабінету	Реальне значення	Нормативне значення
1	Площа на 1 працюючого	25 м ²	4,5 м ²
2	Об'єм на 1 працюючого	75 м ³	15 м ³
3	Мінімальна ширина проходу	1,5 м	1,5 м
4	Ширина дверного проходу	1 м	Не менше 1 м

На основі результатів, представлених у таблиці 4.2, можна зробити висновок, що розглянутий медичний кабінет відповідає всім встановленим нормативам [46].

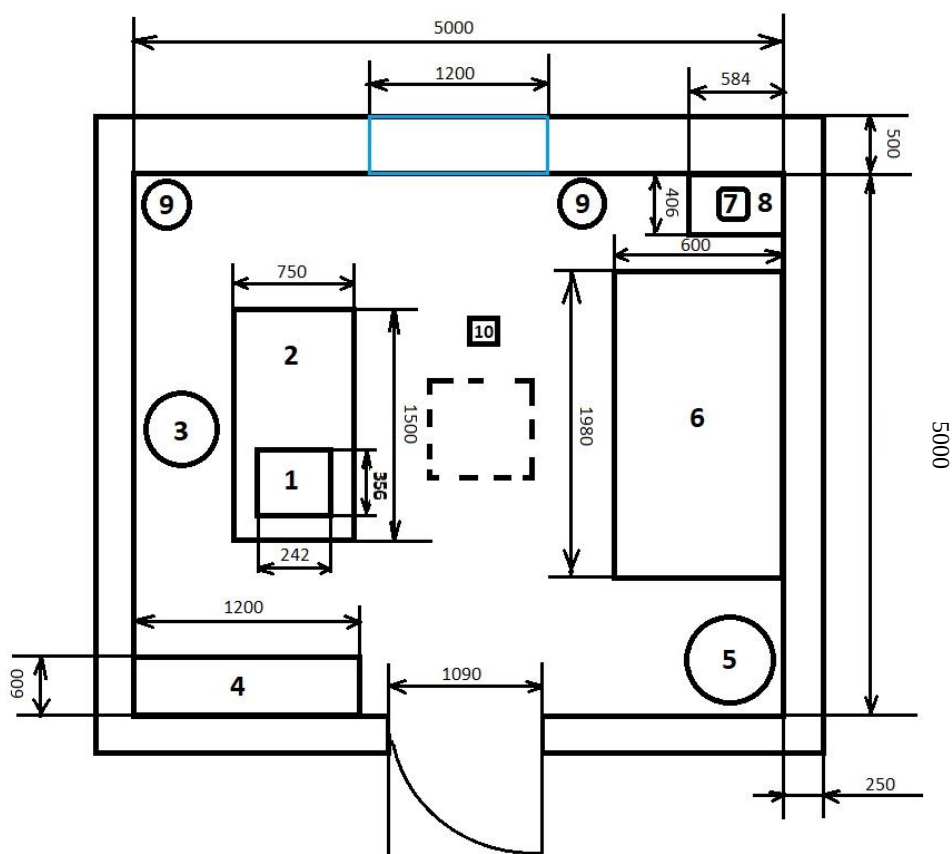


Рисунок 4.1 – Схема медичного кабінету

4.2 Оцінка небезпечних та шкідливих факторів

Згідно з ГОСТ 12.0.003-74 [46], всі потенційно небезпечні та шкідливі фактори класифікуються на фізичні, хімічні, біологічні, та психофізіологічні групи. Висновки про наявність або відсутність цих факторів під час проведення тестування моделі для діагностики шкірних захворювань в медичному кабінеті представлені в таблиці 4.3.

Таблиця 4.3 – Основні небезпечні та шкідливі фактори

Тип небезпеки	Основні фактори
Фізичні	Електробезпека, пожежна безпека, мікроклімат
Хімічні	Відсутні
Біологічні	Ризик контакту з вірусами, бактеріями, грибками під час проведення медичних процедур
Психофізіологічні	Відсутні

4.2.1 Електронебезпека

Інформацію про джерела небезпеки та заходи для захисту, що стосуються електробезпеки наведено в таблицях 4.4-4.6.

Таблиця 4.4 – Джерела небезпеки

Джерело небезпеки	Причина небезпеки	Наслідки небезпеки
Ноутбук HP 250 G8 (45S01ES)	Пошкодження шнуру живлення, коротке замикання, неправильна експлуатація	Електричний удар, пожежа, збій в роботі обладнання
Світильник ЛПО-01	Пошкодження проводки, коротке замикання, перегрів	Електричний удар, пожежа, втрата освітлення
Фотоапарат Canon EOS 4000D BK 18-55	Пошкодження батареї, неправильне заряджання, коротке замикання	Електричний удар, пожежа, збій в роботі обладнання
Протипожежна система ДИП-1	Несправність системи, коротке замикання	Неможливість виявлення та гасіння пожежі, електричний удар

Таблиця 4.5 – Реальні та нормативні фактори небезпеки

Фактор небезпеки	Реальне значення	Нормативне значення
Максимальний струм	> 1 А	1 А

Таблиця 4.6 – Заходи та засоби захисту

Вид захисту	Засоби подолання небезпеки	Критерії вибору
Технічні заходи	Установка автоматичних вимикачів, використання захисних кабелів, регулярна перевірка і заміна пошкоджених шнурів та кабелів, використання надійних та сертифікованих пристроїв заряджання, встановлення системи гасіння пожежі.	Попередження перевантаження, короткого замикання, пожежі.
Експлуатаційні заходи	Використання обладнання відповідно до інструкцій, уникнення перегріву пристроїв, регулярна перевірка стану батарей, уникнення контакту з вологою, регулярне відключення обладнання від електромережі після закінчення роботи.	Збереження життєдіяльності обладнання, попередження випадків перегріву, уникнення замикання через вологу та зниження ризику несподіваних аварійних ситуацій.

Продовження таблиці 4.6

Вид захисту	Засоби подолання небезпеки	Критерії вибору
Організаційні заходи	Проведення регулярних навчань з питань безпеки, встановлення правил та процедур безпеки, регулярна перевірка приміщення та обладнання, впровадження плану евакуації на випадок надзвичайних ситуацій.	Гарант планування та виконання ефективних дій у випадку надзвичайних ситуацій.
Режимні заходи	Обмеження використання обладнання під час грози, дотримання правил роботи з електричними пристроями, установлення графіку технічного обслуговування та моніторингу стану обладнання.	Безпека користування електрообладнанням.

Реальне значення струму не перевищує стандартні нормативні значення, тому приміщення можна вважати достатньо безпечним для роботи.

4.2.2 Пожежна безпека

Інформацію про джерела небезпеки та заходи для захисту, що стосуються пожежної безпеки наведено в таблицях 4.7 – 4.9.

Таблиця 4.7 – Джерела небезпеки

Джерело небезпеки	Причина небезпеки	Наслідки
Ноутбук HP 250 G8 (45S01ES)	Перегрів, коротке замикання, несправність живлення	Пожежа, збій в роботі обладнання
Світильник ЛПО-01	Перегрів, коротке замикання	Пожежа, втрата освітлення
Фотоапарат Canon EOS 4000D BK 18-55	Перегрів, коротке замикання, неправильне заряджання	Пожежа, збій в роботі обладнання
Протипожежна система ДИП-1	Несправність системи, коротке замикання	Неможливість виявлення та гасіння пожежі

Таблиця 4.8 – Характеристика пожежної небезпеки приміщення

Категорія пожежної небезпеки	Клас пожежонебезпечної зони	Клас пожежної небезпеки
D(низька пожежна небезпека)	П-Па	Е (горіння електроустановок під напругою)

Таблиця 4.9 – Заходи та засоби захисту

Вид захисту	Засоби подолання небезпеки	Критерії вибору
Технічні	Наявність вогнегасників, протипожежна система	Припинення вогню в разі пожежі
Експлуатаційні	Правила користування електроприладами, регулярна перевірка технічного стану обладнання.	Попередження пожежі
Організаційні	План евакуації, регулярні навчання та інструктажі з пожежної безпеки, правила поведінки при виявленні диму чи вогню відповідальні особи за пожежну безпеку	Гарант коректних дій в разі виникнення пожежі
Режимні	Відповідність режиму роботи вимогам пожежної безпеки, режим контролю за дотриманням вимог пожежної безпеки	Запобігання виникнення пожежі

За допомогою цих заходів можна зменшити ймовірність виникнення та розповсюдження пожежі.

4.2.3 Мікроклімат

Інформацію про джерела небезпеки та заходи для захисту, що стосуються проблем з мікрокліматом в приміщенні наведено в таблицях 4.10 – 4.11.

Таблиця 4.10 – Джерела небезпеки

Джерело небезпеки	Причина	Наслідки
Недостатній контроль за температурою в приміщенні	Поганий мікроклімат або неефективне управління кліматичною системою	Перегрів або переохолодження пацієнта, що може призвести до теплового удару або гіпотермії
Проблеми з вентиляцією	Еефективна система вентиляції або відсутність регулярного обслуговування	Підвищений ризик розповсюдження мікроорганізмів, можливість виникнення проблем з дихальною системою у пацієнтів або персоналу
Відсутність контролю за вологістю	Поганий мікроклімат, неефективне управління системою кондиціювання повітря	Пересушена або надмірно волога шкіра може сприяти появі шкірних захворювань або посилити вже існуючі проблеми

Таблиця 4.11 – Засоби захисту

Вид захисту	Засоби подолання небезпеки	Критерії вибору
Технічний	Система клімат-контролю, яка здатна регулювати температуру та вологість	Забезпечує комфортний мікроклімат, запобігає перегріву або переохолодженню, пересушуванню або надмірному зволоженню повітря
Експлуатаційний	Регулярне обслуговування і перевірка систем вентиляції та освітлення	Забезпечує ефективне функціонування обладнання, зменшує ризик виникнення проблем з дихальною системою

Зазначені вище заходи дозволять мінімізувати негативний вплив джерел порушення мікроклімату в медичному кабінеті.

4.2.4 Біологічні фактори

Інформацію про джерела небезпеки та заходи для захисту, що стосуються біологічної небезпеки наведено в таблицях 4.12 – 4.13.

Таблиця 4.12 – Небезпека, яку представляють біологічні фактори

Біологічний фактор	Шлях передачі та зараження
Бактерії	Повітряно-краплинний шлях, контактний шлях
Віруси	Повітряно-краплинний шлях, контактний шлях, через кров, через секрети та відділення організму
Грибки	Контактний шлях, через пошкодження шкіри, через воду, повітря

Таблиця 4.13 – Заходи та засоби захисту

Вид захисту	Засоби подолання небезпеки	Критерії вибору
Технічні	Використання спеціалізованого обладнання та пристроїв для запобігання інфекцій (наприклад, стерилізатори, ультрафіолетові лампи)	Зниження ризику передачі інфекцій
Організаційні	Розробка і впровадження протоколів і стандартів безпеки, включаючи санітарні норми та процедури	Підвищення стерильності приміщення
Засоби індивідуального захисту	Використання персонального захисного обладнання, такого як рукавички, маски, захисні окуляри, халати	Забезпечення безпеки пацієнтів і персоналу

За допомогою цих заходів можна зменшити вплив біологічних небезпек.

Висновки до розділу 4

В даному розділі було розглянуто план медичного кабінету для діагностики захворювань шкіри. Проаналізовано можливі фізичні та біологічні у відповідності до існуючих стандартів. Приміщення медичного кабінету відповідає вимогам електробезпеки, пожежної безпеки. Запропоновано засоби для мінімізації впливу біологічних факторів впливу.

Приміщення відповідає загальним вимогам безпеки.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		53

ВИСНОВКИ

В результаті літературного та патентного пошуку визначено, що більшість моделей для діагностики захворювань шкіри використовують згорткові нейронні мережі та рекурентні нейронні мережі.

В результаті аналізу роботи існуючих моделей було виявлено, що використання однієї нейронної мережі не дасть достатньо достовірних результатів діагностики. Тому було прийнято рішення про створення нової моделі на основі комбінацій CNN та LSTM мереж.

На основі літературного пошуку та аналізу роботи існуючих моделей машинного навчання було сформовано основні вимоги до нової моделі та реалізовано її.

Прописано та виконано функції тестування моделі за різними користувацькими сценаріями.

Проведено оцінку ефективності моделі. Загальний показник точності диференціювання мальформацій шкіри на основі медичних зображень з використанням згорткових та рекурентних нейронних мереж склав 91%.

Розглянуто можливість застосування розробленої моделі в медичному кабінеті з точки зору охорони праці.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		54

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Pathan S. Techniques and algorithms for computer aided diagnosis of pigmented skin lesions - A review [Електронний ресурс] / S. Pathan, K. Prabhu, P. Siddalingaswamy. – 2018. – Режим доступу до ресурсу: <https://www.semanticscholar.org/paper/>.
2. Skin Malformation [Електронний ресурс] // Fundamentals of Toxicologic Pathology (Second Edition). – 2010. – Режим доступу до ресурсу: <https://www.sciencedirect.com/topics/medicine-and-dentistry/skin-malformation>.
3. Mark L. Actinic Keratosis Overview [Електронний ресурс] / L. Mark, G. Leonard. – 2022. – Режим доступу до ресурсу: <https://www.skincancer.org/skin-cancer-information/actinic-keratosis/>.
4. DermNet [Електронний ресурс] – Режим доступу до ресурсу: <https://dermnetnz.org/>.
5. Seborrheic Keratosis [Електронний ресурс] // Cleveland Clinic medical professional. – 2021. – Режим доступу до ресурсу: <https://my.clevelandclinic.org/health/diseases/21721-seborrheic-keratosis>.
6. Oakley A. Melanocytic naevus [Електронний ресурс] / Amanda Oakley. – 2016. – Режим доступу до ресурсу: <https://dermnetnz.org/topics/melanocytic-naevus>.
7. Mayo Clinic Staff. Melanoma [Електронний ресурс] / Mayo Clinic Staff. – 2022. – Режим доступу до ресурсу: <https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884>.
8. Karen J. Basal Cell Carcinoma Overview [Електронний ресурс] / J. Karen, R. Moy. – 2022. – Режим доступу до ресурсу: <https://www.skincancer.org/skin-cancer-information/basal-cell-carcinoma/>.
9. Oakley A. Dermatofibroma [Електронний ресурс] / Amanda Oakley. – 2020. – Режим доступу до ресурсу: <https://dermnetnz.org/topics/dermatofibroma>.

10. Vascular Lesion [Електронний ресурс] // Handbook of Clinical Neurology. – 2020. – Режим доступу до ресурсу: <https://www.sciencedirect.com/topics/medicine-and-dentistry/vascular-lesion>.

11. Buras W. Method for developing a machine learning model of a neural network for classifying medical images [Електронний ресурс] / William Buras. – 2021. – Режим доступу до ресурсу: <https://patents.google.com/patent/US11017695B2/en>.

12. Hsieh J. Deep learning medical systems and methods for image reconstruction and quality evaluation [Електронний ресурс] / Jiang Hsieh. – 2021. – Режим доступу до ресурсу: <https://patents.google.com/patent/US10896352B2/en>.

13. Hsieh J. Deep learning medical systems and methods for medical procedures [Електронний ресурс] / Jiang Hsieh. – 2019. – Режим доступу до ресурсу: <https://patents.google.com/patent/US10438354B2/en>+

14. Hsieh J. Deep learning medical systems and methods for image acquisition [Електронний ресурс] / Jiang Hsieh. – 2020. – Режим доступу до ресурсу: <https://patents.google.com/patent/US10628943B2/en>.

15. Grady L. Systems and methods for medical acquisition processing and machine learning for anatomical assessment [Електронний ресурс] / Leo Grady. – 2020. – Режим доступу до ресурсу: <https://patents.google.com/patent/US10789706B2/en>.

16. Melanoma detection by analysis of clinical images using convolutional neural network. [Електронний ресурс] – Режим доступу до ресурсу: <https://s3-us-west-2.amazonaws.com/>.

17. Esteva A. Dermatologist-level classification of skin cancer with deep neural networks [Електронний ресурс] / A. Esteva, B. Kuprel. – 2017. – Режим доступу до ресурсу: <https://www.nature.com/articles/nature21056>.

18. Steuer G. A system and method of diagnosis skin and tissue lesions and abnormalities [Електронний ресурс] / G. Steuer, A. Ingber. – 2018. – Режим доступу до ресурсу: <https://patents.google.com/patent/WO2018146688A1/en>.

19. Patnaik S. Automated Skin Disease Identification using Deep Learning Algorithm [Електронний ресурс] / S. Patnaik, M. Sidhu. – 2018. – Режим доступу до ресурсу: <https://biomedpharmajournal.org/vol11no3/>.

20. Nawal Soliman ALKolifi ALEnezi. A Method Of Skin Disease Detection Using Image Processing And Machine Learning [Електронний ресурс] / Nawal Soliman ALKolifi ALEnezi. – 2019. – Режим доступу до ресурсу: <https://www.sciencedirect.com>.

21. Chuchu N. Smartphone applications for triaging adults with skin lesions that are suspicious for melanoma [Електронний ресурс] / Naomi Chuchu Chuchu. – 2018. – Режим доступу до ресурсу: <https://www.cochranelibrary.com/cdsr/>.

22. Brownlee J. How Do Convolutional Layers Work in Deep Learning Neural Networks? [Електронний ресурс] / Jason Brownlee // Deep Learning for Computer Vision. – 2019. – Режим доступу до ресурсу: <https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/>.

23. Albelwi S. A Framework for Designing the Architectures of Deep Convolutional Neural Networks [Електронний ресурс] / S. Albelwi, A. Mahmood. – 2017. – Режим доступу до ресурсу: <https://www.mdpi.com/1099-4300/19/6/242>.

24. Волянський І. A Deep Dive into the Basics of Deep Learning [Електронний ресурс] / Іван Волянський. – 2021. – Режим доступу до ресурсу: <https://mspsss.org.ua/index.php/journal/article/view/582>.

25. What is translation invariance in computer vision and convolutional neural network? [Електронний ресурс] – Режим доступу до ресурсу: <https://stats.stackexchange.com/questions/208936/what-is-translation-invariance-in-computer-vision-and-convolutional-neural-netwo>.

26. Convolution and Cross Correlation in CNN [Електронний ресурс] – Режим доступу до ресурсу:

<https://datascience.stackexchange.com/questions/>.

27. Awati R. Convolutional neural network (CNN) [Електронний ресурс] / Rahul Awati. – 2023. – Режим доступу до ресурсу: <https://www.techtarget.com/>.

28. Skin Lesion Analyzer [Електронний ресурс] – Режим доступу до ресурсу: <http://skin.test.woza.work/>.

29. Recurrent Neural Network (RNN) in TensorFlow [Електронний ресурс] – Режим доступу до ресурсу: <https://www.javatpoint.com/recurrent-neural-network-in-tensorflow>.

30. Hesamian M. Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges [Електронний ресурс] / Mohammad Hesamian. – 2019. – Режим доступу до ресурсу: <https://www.ncbi.nlm.nih.gov/>.

31. What is LSTM? Introduction to Long Short Term Memory [Електронний ресурс] // Intellipaat. – 2023. – Режим доступу до ресурсу: <https://intellipaat.com/blog/what-is-lstm/?US>.

32. Tripathi A. What is the main difference between RNN and LSTM [Електронний ресурс] / Ashutosh Tripathi. – 2021. – Режим доступу до ресурсу: <https://ashutoshtripathi.com/2021/07/02/>.

33. Середовище Google Colab [Електронний ресурс] – Режим доступу до ресурсу: <https://colab.research.google.com/>.

34. Jwalapuram N. A Gentle Introduction to PyTorch Library for Deep Learning [Електронний ресурс] / Narasimha Jwalapuram. – 2021. – Режим доступу до ресурсу: <https://www.analyticsvidhya.com/blog/2021/04/>.

35. OpenCV [Електронний ресурс] – Режим доступу до ресурсу: <https://opencv.org/>.

36. NumPy [Електронний ресурс] – Режим доступу до ресурсу: <https://numpy.org/>.

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		58

37. Pandas [Електронний ресурс] – Режим доступу до ресурсу: <https://pandas.pydata.org/>.

38. Matplotlib: Visualization with Python [Електронний ресурс] – Режим доступу до ресурсу: <https://matplotlib.org/>.

39. Kasiotis A. Skin Cancer MNIST: HAM10000 disease Classification (Extended Project) [Електронний ресурс] / Antreas Kasiotis. – 2022. – Режим доступу до ресурсу: <https://github.com/kasiotis/Skin-Cancer-MNIST-HAM10000>.

40. Tschandl P. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions [Електронний ресурс] / Philipp Tschandl. – 2018. – Режим доступу до ресурсу: <https://arxiv.org/abs/1803.10417>.

41. Agrawal S. How to split data into three sets (train, validation, and test) And why? [Електронний ресурс] / Samarth Agrawal. – 2021. – Режим доступу до ресурсу: <https://towardsdatascience.com/>.

42. Машинне навчання в задачах кібербезпеки [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.ztu.edu.ua/>.

43. ResNeXt [Електронний ресурс] – Режим доступу до ресурсу: <https://paperswithcode.com/lib/torchvision/resnext>.

44. Classification: Accuracy [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/machine-learning/>.

45. Machine learning model evaluation metrics part 2: multi-class classification [Електронний ресурс] – Режим доступу до ресурсу: <https://www.mariakhalusova.com/>.

46. Методичні вказівки до виконання розділу: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою «Регенеративна та біофармацевтична інженерія» спеціальності 163 Біомедична інженерія / КПІ ім. Ігоря Сікорського; уклад.: Г.В. Демчук, С.Ф. Каштанов, В.В. Мотроненко – Київ: КПІ ім. Ігоря Сікорського, 2022, - 33 с.

ДОДАТОК А

Лістинг коду моделі

```
#Прив'язка до Гугл Диску
from google.colab import drive
drive.mount('/content/gdrive')
#Імпорт бібліотек, необхідних для роботи
import os
from tqdm import tqdm
from glob import glob
import cv2
import pandas as pd
import numpy as np
from PIL import Image
import matplotlib.image as mpimg
import matplotlib.pyplot as plt
%matplotlib inline
plt.ion()
import torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torch.autograd import Variable
import torchvision
from torch.utils.data import DataLoader, Dataset
from torchvision import transforms, models
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
#Зчитування вмісту директорії
os.listdir('gdrive/MyDrive/Диплом')
base_dir = os.path.join('gdrive', 'MyDrive', 'Диплом' )
all_image_path = glob(os.path.join(base_dir, '*', '*.jpg'))
imageid_path_dict = {os.path.splitext(os.path.basename(x))[0]: x for x in
all_image_path}
lesion_type_dict = {
    'nv': 'Melanocytic nevi',
    'mel': 'Melanoma',
    'bkl': 'Benign keratosis-like lesions ',
    'bcc': 'Basal cell carcinoma',
    'akiec': 'Actinic keratoses',
    'vasc': 'Vascular lesions',
    'df': 'Dermatofibroma'
}
# Перевірка розміру та кількості каналів на зображенні
```

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		60

```

arr = np.asarray(Image.open(all_image_path[48]))
arr.shape
#Зчитування CSV-файлу з метаданими зображень, створення шляху до
зображень,
# надання назв класів та числових індекси для класів у метаданих зображень
df_original = pd.read_csv(os.path.join(base_dir, 'HAM10000_metadata.csv'))
df_original['path'] = df_original['image_id'].map(imageid_path_dict.get)
df_original['cell_type'] = df_original['dx'].map(lesion_type_dict.get)
df_original['cell_type_idx'] =
pd.Categorical(df_original['cell_type']).codes
df_original.head()
df = pd.read_csv(os.path.join(base_dir, 'HAM10000_metadata.csv'))
df = df.sort_values(['lesion_id', 'image_id'])
df['previous_diagnosis'] = df.groupby('lesion_id')['dx'].shift(1)
# Замінюємо NaN на "no previous diagnosis" або будь-яку іншу відповідну
df['previous_diagnosis'].fillna('no previous diagnosis', inplace=True)
df.to_csv(os.path.join(base_dir, 'HAM10000_metadata.csv'), index=False)
print(df.columns)
#Цей код отримує унікальні класи "cell_type" та їх числові індекси
"cell_type_idx"
# з DataFrame "df_original", відсортовані за зростанням значення
"cell_type_idx".
df_original[['cell_type_idx',
'cell_type']].sort_values('cell_type_idx').drop_duplicates()
#Підрахунок кількості входжень кожного унікального значення в стовпці
df_original['cell_type'].value_counts()
#Обчислення середнього значення та стандартного відхилення для
#трьох каналів (RGB) на основі зображень, поданих списком шляхів до
зображень.
def compute_img_mean_std(image_paths):
    # Задамо розмір зображення.
    img_h, img_w = 224, 224
    # Ініціалізуємо списки для зберігання зображень та статистик.
    imgs = []
    channel_means, channel_stdevs = [], []
    # Перебираємо по всіх шляхах до зображень.
    for img_path in tqdm(image_paths):
        # Зчитуємо зображення.
        img = cv2.imread(img_path)
        # Змінюємо розмір зображення.
        resized_img = cv2.resize(img, (img_h, img_w))
        # Додаємо зображення до списку.
        imgs.append(resized_img)
    # Створюємо масив зі всіх зображень.
    imgs_array = np.stack(imgs, axis=3)
    # Нормалізуємо зображення, перетворивши їх на float32 та поділивши на
255.
    normalized_imgs = imgs_array.astype(np.float32) / 255.

```

```

# Перебираємо канали у кольоровому просторі та розраховуємо середнє
значення та стандартне відхилення.
for i in range(3):
    # Перетворюємо масив у одновимірний.
    channel_pixels = normalized_imgs[:, :, i, :].ravel()
    # Обчислюємо середнє та стандартне відхилення та додаємо їх до
    відповідних списків.
    channel_means.append(np.mean(channel_pixels))
    channel_stdevs.append(np.std(channel_pixels))
# Обертаємо списки, щоб вони відповідали порядку каналів BGR.
channel_means.reverse()
channel_stdevs.reverse()
# Повертаємо результат.
return channel_means, channel_stdevs
print("normMean = {}".format(channel_means))
print("normStd = {}".format(channel_stdevs))
return channel_means, channel_stdevs
normMean = [0.7630358, 0.54564357, 0.5700475]
normStd = [0.14092763, 0.15261263, 0.16997081]
# це скаже нам, скільки зображень пов'язано з кожним lesion_id
df_undup = df_original.groupby('lesion_id').count()
# фільтрація lesion_id, які мають лише одне зображення, пов'язане з ним
df_undup = df_undup[df_undup['image_id'] == 1]
df_undup.reset_index(inplace=True)
df_undup.head()
# ідентифікація lesion_id, які мають повторювані зображення, і ті, які
мають лише одне зображення
def check_duplicate_status(lesion_id, unique_id_set):
    # Перевірка, чи є 'lesion_id' унікальним.
    if lesion_id in unique_id_set:
        return 'unduplicated'
    else:
        return 'duplicated'
# створення нового стовпця, який є копією стовпця lesion_id
df_original['duplicates'] = df_original['lesion_id']
# застосування функції до нової колонки
df_original['duplicates'] =
df_original['duplicates'].apply(check_duplicate_status)
df_original.head()
df_original['duplicates'].value_counts()
#фільтрація зображень, які не мають дублікатів
df_undup = df_original[df_original['duplicates'] == 'unduplicated']
df_undup.shape
# створення набору val за допомогою df, оскільки жодне з цих зображень не
має доповнених дублікатів у тренувальному наборі
y = df_undup['cell_type_idx']
_, df_val = train_test_split(df_undup, test_size=0.2, random_state=101,
stratify=y)
df_val.shape

```

						Лист
Изм.	Лист	№ докум.	Підпис	Дата		62

```

df_val['cell_type_idx'].value_counts()
# Ця функція визначає, чи є зображення частиною набору train або val.
def get_val_rows(x):
# створення списку всіх lesion_id у наборі val
    val_list = list(df_val['image_id'])
    if str(x) in val_list:
        return 'val'
    else:
        return 'train'
df_original['train_or_val'] = df_original['image_id']
df_original['train_or_val'] =
df_original['train_or_val'].apply(get_val_rows)
df_train = df_original[df_original['train_or_val'] == 'train']
print(len(df_train))
print(len(df_val))
df_train['cell_type'].value_counts()
df_val['cell_type'].value_counts()
#Копіювання меншої кількості класів, щоб збалансувати кількість 7 класів
data_aug_rate = [15,10,5,50,0,40,5]
for i in range(7):
    if data_aug_rate[i]:
        df_train=df_train.append([df_train.loc[df_train['cell_type_idx'] ==
i,:]]*(data_aug_rate[i]-1), ignore_index=True)
df_train['cell_type'].value_counts()
df_train = df_train.reset_index()
df_val = df_val.reset_index()
df_train.sample(5)
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
device
class PretrainedResNextLSTM(nn.Module):
    def __init__(self, num_class=7, hidden_size=128, num_layers=1):
        super().__init__()
        self.resNext = models.resnext101_32x8d(pretrained=True)
        self.channels = self.resNext.fc.in_features # Оновлено тут
        for params in self.resNext.parameters():
            params.requires_grad_(False)
        self.features = nn.Sequential(*list(self.resNext.children())[:-1])
        self.lstm = nn.LSTM(self.channels + 1, hidden_size, num_layers,
batch_first=True)
        self.fc1 = nn.Linear(hidden_size, num_class)
        self.softmax = nn.Softmax(dim=1)
    def forward(self, x, previous_diagnosis):
        x = self.features(x)
        x = torch.mean(x, dim=(2, 3))
        x_and_previous_diagnosis = torch.cat([x, previous_diagnosis.view(-1,
1)], dim=1)
        x = x_and_previous_diagnosis.unsqueeze(0)
        lstm_out, _ = self.lstm(x)
        x = self.fc1(lstm_out)

```

```

        out = self.softmax(x)
        return out.squeeze(0)
model = PretrainedResNextLSTM().to(device)
optimizer = optim.Adam(model.parameters(), lr=1e-3)
criterion = nn.CrossEntropyLoss().to(device)
#створення моделі model_ft на основі попередньо навченої архітектури
model = PretrainedResNext()
model_ft = models.resnext101_32x8d(pretrained=True)
model_ft.fc = nn.Linear(in_features=2048, out_features=7)
model_ft
model = model_ft
class CustomDataset(Dataset):
    def __init__(self, df, transform=None):
        self.df = df
        self.transform = transform
    def __len__(self):
        return len(self.df)
    def __getitem__(self, index):
        # Load data and get label
        X = Image.open(self.df['path'][index])
        y = torch.tensor(int(self.df['cell_type_idx'][index]))
        # get the previous diagnosis
        previous_diagnosis = self.df['previous_diagnosis'][index]
        # Map the previous diagnosis to a numerical identifier
        diagnosis_dict = {
            'nv': 0,
            'mel': 1,
            'bkl': 2,
            'bcc': 3,
            'akiec': 4,
            'vasc': 5,
            'df': 6,
            'no previous diagnosis': -2
        }
        if previous_diagnosis in diagnosis_dict:
            previous_diagnosis_code = diagnosis_dict[previous_diagnosis]
        else:
            previous_diagnosis_code = -1
        previous_diagnosis = torch.tensor(previous_diagnosis_code)
        if self.transform:
            X = self.transform(X)
        return X, y, previous_diagnosis
#визначення перетворень для підготовки даних перед навчанням та перевіркою
моделі
input_size = 224
# визначаємо трансформації для навчання
train_transform = transforms.Compose([
    transforms.RandomResizedCrop(input_size),
    transforms.RandomHorizontalFlip(),

```

```

        transforms.RandomVerticalFlip(),
        transforms.RandomRotation(20),
        transforms.ColorJitter(brightness=0.1, contrast=0.1, hue=0.1),
        transforms.RandomGrayscale(p=0.2),
        transforms.ToTensor(),
        transforms.Normalize(normMean, normStd)
    ])
# визначаємо трансформації для валідації
val_transform = transforms.Compose([
    transforms.Resize((input_size,input_size)),
    transforms.CenterCrop(input_size),
    transforms.ToTensor(),
    transforms.Normalize(normMean, normStd)
])
# Визначення навчального набору за допомогою таблиці train_df і
використання визначених нами переходів (train_transform)
training_set = CustomDataset(df_train, transform=train_transform)
train_loader = DataLoader(training_set, batch_size=32, shuffle=True,
num_workers=4)
# те ж само для валідаційного набору
validation_set = CustomDataset(df_val, transform=train_transform)
val_loader = DataLoader(validation_set, batch_size=32, shuffle=False,
num_workers=4)
for i, data in enumerate(train_loader):
    images, labels, prev_diagnosis = data
    print(images.shape) # Додайте цей рядок для перевірки розміру зображення
# ця функція використовується в процесі навчання, для розрахунку втрат і
точності
class AverageMeter(object):
    def __init__(self):
        self.reset()
    def reset(self):
        self.val = 0
        self.avg = 0
        self.sum = 0
        self.count = 0
    def update(self, val, n=1):
        self.val = val
        self.sum += val * n
        self.count += n
        self.avg = self.sum / self.count
total_loss_train, total_acc_train = [],[]
# LSTM треба передати попередній діагноз як вхідні дані, використовуючи
зміннені функції тренування та перевірки.
def calculate_accuracy(outputs, labels):
    prediction = outputs.max(1, keepdim=True)[1]
    acc = prediction.eq(labels.view_as(prediction)).sum().item()
    return acc

```

```

def calculate_loss(criterion, outputs, labels):
    return criterion(outputs, labels).item()
def train(train_loader, model, criterion, optimizer, scheduler, epoch):
    model.train()
    train_loss = AverageMeter()
    train_acc = AverageMeter()
    for i, data in enumerate(train_loader):
        try:
            images, labels, previous_diagnosis = data
            images = Variable(images).to(device)
            labels = Variable(labels).to(device)
            previous_diagnosis =
Variable(previous_diagnosis.float()).to(device)
            optimizer.zero_grad()
            outputs = model(images, previous_diagnosis)
            loss = calculate_loss(criterion, outputs, labels)
            acc = calculate_accuracy(outputs, labels)
            train_loss.update(loss)
            train_acc.update(acc / images.size(0))
            loss.backward()
            optimizer.step()
            scheduler.step()
            if (i + 1) % 100 == 0:
                print(f'Epoch: {epoch}, Iteration: {i +
1}/{len(train_loader)}, Training Loss: {train_loss.avg:.5f}, Training
Accuracy: {train_acc.avg:.5f}')
        except Exception as e:
            print(f'Error occurred at iteration {i+1}: {str(e)}')
    return train_loss.avg, train_acc.avg
#процес перевірки (валідації) моделі на наборі даних для визначення
#середнього значення втрат та точності моделі на валідаційному наборі
даних
def validate(val_loader, model, criterion, optimizer, epoch):
    model.eval()
    val_loss = AverageMeter()
    val_acc = AverageMeter()
    with torch.no_grad():
        for i, data in enumerate(val_loader):
            images, labels, previous_diagnosis = data
            N = images.size(0)
            images = Variable(images).to(device)
            labels = Variable(labels).to(device)
            previous_diagnosis = Variable(previous_diagnosis.float()).to(device)
            outputs = model(images, previous_diagnosis)
            prediction = outputs.max(1, keepdim=True)[1]
            val_acc.update(prediction.eq(labels.view_as(prediction)).sum().item() / N)
            val_loss.update(criterion(outputs, labels).item())

```

```

    print('[epoch %d], [val loss %.5f], [val acc %.5f]' % (epoch,
val_loss.avg, val_acc.avg))
    return val_loss.avg, val_acc.avg
model.to(device)
optimizer = optim.Adam(model.parameters(), lr=1e-3)
criterion = nn.CrossEntropyLoss().to(device)
#процес тренування моделі протягом заданої кількості епох
epoch_num = 10
best_val_acc = 0
total_loss_val, total_acc_val = [],[]
for epoch in tqdm(range(1, epoch_num+1)):
    loss_train, acc_train = train(train_loader, model, criterion, optimizer,
epoch)
    loss_val, acc_val = validate(val_loader, model, criterion, optimizer,
epoch)
    total_loss_val.append(loss_val)
    total_acc_val.append(acc_val)
    if acc_val > best_val_acc:
        best_val_acc = acc_val
        print('best record: [epoch %d], [val loss %.5f], [val acc %.5f]' %
(epoch, loss_val, acc_val))
fig = plt.figure(num = 2)
fig1 = fig.add_subplot(2,1,1)
fig2 = fig.add_subplot(2,1,2)
fig1.plot(total_loss_train, label = 'training loss')
fig1.plot(total_acc_train, label = 'training accuracy')
fig2.plot(total_loss_val, label = 'validation loss')
fig2.plot(total_acc_val, label = 'validation accuracy')
plt.legend()
plt.show()
#створення матриці помилок
def plot_confusion_matrix(cm, classes,
                        normalize=False,
                        title='Матриця помилок',
                        cmap=plt.cm.Blues):
    plt.figure(figsize=(10,10))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)
    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                horizontalalignment="center",
                color="white" if cm[i, j] > thresh else "black")
    plt.tight_layout()

```

```

plt.ylabel('True label')
plt.xlabel('Predicted label')
def predict(model, val_loader, device):
    model.eval()
    y_label = []
    y_predict = []
    with torch.no_grad():
        for i, data in enumerate(val_loader):
            images, labels, previous_diagnosis = data
            images = images.to(device)
            previous_diagnosis = previous_diagnosis.to(device)
            outputs = model(images, previous_diagnosis)
            prediction = outputs.argmax(dim=1)
            y_label.extend(labels.cpu().numpy())
            y_predict.extend(prediction.cpu().numpy())
    return y_label, y_predict
# compute the confusion matrix
confusion_mtx = confusion_matrix(y_label, y_predict)
# plot the confusion matrix
plot_labels = ['akiec', 'bcc', 'bkl', 'df', 'nv', 'vasc', 'mel']
plot_confusion_matrix(confusion_mtx, plot_labels)
import itertools
plot_confusion_matrix(confusion_mtx, plot_labels)
# Генерація звіту класифікації
report = classification_report(y_label, y_predict,
target_names=plot_labels)
print(report)

```

ДОДАТОК Б

```

# Тестування моделі за різними сценаріями
from datetime import datetime
def save_diagnosis_to_csv(user_id, diagnosis, probability):
    file_name = f"{user_id}_diagnoses.csv"
    file_exists = os.path.isfile(file_name)
    with open(file_name, 'a', newline='') as csvfile:
        writer = csv.writer(csvfile)
        if not file_exists:
            writer.writerow(["datetime", "diagnosis", "probability"])
            current_datetime = datetime.now()
            writer.writerow([current_datetime, diagnosis, probability])
def get_last_diagnosis_from_csv(user_id):
    file_name = f"{user_id}_diagnoses.csv"
    if os.path.isfile(file_name):
        with open(file_name, 'r', newline='') as csvfile:
            reader = csv.reader(csvfile)
            rows = list(reader)

```

```

        if len(rows) > 1:
            last_row = rows[-1]
            last_diagnosis = last_row[1]
            last_probability = float(last_row[2])
            return last_diagnosis, last_probability
    return None, None

def analyze_and_compare_diagnosis(predicted_label, predicted_prob,
user_id):
    last_diagnosis, last_probability = get_last_diagnosis_from_csv(user_id)
    if last_diagnosis is not None and last_probability is not None:
        if predicted_prob > last_probability:
            return "Стан хвороби покращився"
        elif predicted_prob < last_probability:
            return "Стан хвороби погіршився"
        else:
            return "Стан хвороби залишився незмінним"
    else:
        if predicted_prob < threshold:
            return "Модель не здатна діагностувати ваш тип захворювання або ви здорові"
        else:
            save_diagnosis_to_csv(user_id, predicted_label, predicted_prob)
            return "Це перше завантаження фото"

def test_model(model, image_path, transform, user_id):
    model.eval()
    image = Image.open(image_path)
    image = transform(image).unsqueeze(0).to(device)
    output = model(image)
    probabilities = torch.softmax(output, dim=1)[0]
    _, predicted_idx = torch.max(output, 1)
    predicted_label = df_train[df_train['cell_type_idx'] ==
predicted_idx.item()][ 'cell_type'].values[0]
    predicted_prob = probabilities[predicted_idx.item()].item() * 100
    return predicted_label, predicted_prob

# Надаємо інструкції користувачу
print("Завантажте фото ділянки шкіри, зроблене на відстані від 5 до 20
см.")
# Аналізуємо фотографію шкіри користувача
image_path = 'Melanoma.jpg'
user_id = 6
threshold = 60
predicted_label, predicted_prob = test_model(model, image_path,
val_transform, user_id)
health_status = analyze_and_compare_diagnosis(predicted_label,
predicted_prob, user_id)
# Виводимо результати
print(f"Діагноз: {predicted_label}")
print(f"Ймовірність: {predicted_prob}")
print(health_status)

```