

Міністерство освіти і науки України  
Університет митної справи та фінансів

Факультет інноваційних технологій  
Кафедра комп'ютерних наук та інженерії програмного забезпечення

## Кваліфікаційна робота магістра

на тему: «Розроблення та машинне навчання нейронної моделі для  
аналізу медичних даних»

Виконав: студент групи К22-1М  
Спеціальність 122 «Комп'ютерні науки»  
Гулін Володимир Володимирович  
(прізвище та ініціали)

Керівник д.е.н., проф. Корнєєв М.В.  
(науковий ступінь, вчене звання, прізвище та ініціали)

Рецензент Університет митної справи та фінансів  
(місто роботи)

Професор кафедри кібербезпеки та  
(посада)  
інформаційних технологій  
(посада)

д.е.н., проф. Паршина О.А.  
(науковий ступінь, вчене звання, прізвище та ініціали)

## АНОТАЦІЯ

*Гулін В.В.* Розроблення та машинне навчання нейронної моделі для аналізу медичних даних.

Дипломна робота на здобуття освітнього ступеня магістр за спеціальністю 122 «Комп'ютерні науки» – Університет митної справи та фінансів, Дніпро, 2024.

Розробка і машинне навчання нейронних моделей для аналізу медичних даних стали ключовими напрямками досліджень, спрямованих на поліпшення якості діагностики, прогнозування захворювань і розробки ефективних методів лікування. Цей амбітний та інноваційний підхід не тільки надає лікарям більш точні інструменти для прийняття рішень, а й відкриває нові горизонти в розумінні медичних даних.

Об'єктом дослідження є процес розроблення та машинного навчання нейронних моделей, призначених для аналізу медичних даних. Дослідження спрямоване на розуміння того, які особливості та виклики існують у сфері медичних даних, а також яким чином нейронні моделі можуть бути найефективніше застосовані для їхнього аналізу.

Предметом дослідження є сутність і характер процесу розроблення та машинного навчання нейронних моделей, що застосовуються для аналізу медичних даних.

Метою магістерської роботи є глибоке дослідження процесу розроблення та застосування нейронних моделей у контексті аналізу медичних даних з використанням методів машинного навчання. Мета полягає у виявленні ключових аспектів цього інноваційного підходу, спрямованого на поліпшення діагностики, прогнозування та лікування захворювань.

Ключові слова: нейронні мережі, нейронні моделі, SciPy, Scikit-learn, Breast Cancer Wisconsin.

## ABSTRACT

Gulin V.V. Development and machine learning of a neural model for medical data analysis.

Diploma thesis for a master's degree in speciality 122 «Computer Science» – University of Customs and Finance, Dnipro, 2024.

The development and machine learning of neural models for medical data analysis have become key areas of research aimed at improving the quality of diagnosis, predicting diseases and developing effective treatments. This ambitious and innovative approach not only provides doctors with more accurate decision-making tools, but also opens up new horizons in understanding medical data.

The object of the study is the process of developing and machine learning neural models designed to analyse medical data. The research aims to understand the peculiarities and challenges of medical data and how neural models can be most effectively applied to analyse it.

The subject of the research is the nature and nature of the process of developing and machine learning neural models used for medical data analysis.

The purpose of the master's thesis is to conduct an in-depth study of the process of developing and applying neural models in the context of medical data analysis using machine learning methods. The goal is to identify the key aspects of this innovative approach aimed at improving the diagnosis, prognosis and treatment of diseases.

Keywords: neural networks, neural models, SciPy, Scikit-learn, Breast Cancer Wisconsin.

## ЗМІСТ

ВСТУП .....	6
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ .....	15
1.1 Аналіз предметної області .....	15
1.2 Огляд загальних понять про штучні нейронні мережі.....	17
1.3 Нейронні моделі .....	21
1.3.1 Переваги нейронних моделей.....	21
1.3.2 Недоліки нейронних моделей.....	23
1.4 Сучасний стан використання нейронних моделей в медичній галузі ....	24
1.5 Висновки до першого розділу .....	36
РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ НЕЙРОННИХ МОДЕЛЕЙ ДЛЯ АНАЛІЗУ МЕДИЧНИХ ДАНИХ .....	38
2.1 Збір даних та їх попереднє опрацювання .....	38
2.2 Згорткові нейронні мережі .....	40
2.3 Рекурентні нейронні мережі .....	42
2.4 Короткострокова пам'ять.....	45
2.5 Глибокі нейронні мережі прямого розповсюдження .....	47
2.6 Глибокі згорткові нейронні мережі.....	49
2.7 Проблема оверфітінгу.....	52
2.8 Методи регуляризації .....	54
2.9 Відбір ознак .....	56
2.10 Передача навчання.....	58
2.11 Огляд програмних засобів для машинного навчання нейронних моделей.....	61
2.11.1 Бібліотека Scikit-learn .....	61

2.11.2 Бібліотека SciPy .....	63
2.11.3 Бібліотека Joblib .....	66
2.12 Висновки до другого розділу .....	68
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ.....	70
3.1 Постановка задачі .....	70
3.2 Процес виконання програмного коду .....	71
3.3 Використані технології.....	72
3.4 Аналіз отриманих результатів .....	73
3.5 Висновки до третього розділу .....	75
ВИСНОВКИ.....	77
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	79
ДОДАТКИ.....	82

## ВСТУП

Розробка і машинне навчання нейронних моделей для аналізу медичних даних стали ключовими напрямками досліджень, спрямованих на поліпшення якості діагностики, прогнозування захворювань і розробки ефективних методів лікування. Цей амбітний та інноваційний підхід не тільки надає лікарям більш точні інструменти для прийняття рішень, а й відкриває нові горизонти в розумінні медичних даних. Розробка нейронних моделей для аналізу медичних даних включає в себе синтез сучасних методів машинного навчання, глибокого навчання та обчислювальної біології з метою створення інтелектуальних систем, здатних аналізувати складні та багатовимірні дані, отримані з різних медичних досліджень. Це забезпечує можливість виявлення прихованих закономірностей, даючи змогу ефективніше виявляти ризики захворювань, робити точніші діагнози та передбачати перебіг хвороб. Актуальність роботи зумовлена різким збільшенням обсягу медичних даних, а також необхідністю вилучення значущої інформації з цього величезного масиву даних. Машинне навчання і розробка нейронних моделей стали ефективними інструментами для аналізу складних і багатовимірних даних у медичній сфері. Ці методи надають можливість виявлення патернів, які можуть залишатися непоміченими при традиційних методах аналізу. З урахуванням динамічного характеру біологічних процесів і багатьох факторів, що впливають на здоров'я людини, потрібен сучасний і адаптивний підхід до аналізу медичних даних. Саме в цьому контексті машинне навчання та нейронні моделі виступають у ролі інноваційного інструменту, здатного автоматично виділяти закономірності та взаємозв'язки, які можуть бути ключовими для розуміння хвороб, їх діагностики та прогнозування. Сьогодні, коли медицина стикається з викликами, такими як персоналізована медицина і передбачення ризиків

різних захворювань, розробка нейронних моделей для аналізу медичних даних набуває стратегічного значення. Ці моделі можуть обробляти різні типи даних, включно з медичними зображеннями, генетичними даними, клінічними записами тощо, інтегруючи їх для створення більш повного і комплексного уявлення про стан пацієнта. Крім того, такий підхід відкриває нові горизонти для інновацій у галузі медичних досліджень, прискорюючи процес відкриття нових медичних знань і розробки перспективних методів лікування. Усе це зумовлює високу актуальність роботи, присвяченої розробці та машинному навчанню нейронних моделей для аналізу медичних даних.

Метою дослідницької роботи є глибоке дослідження процесу розроблення та застосування нейронних моделей у контексті аналізу медичних даних з використанням методів машинного навчання. Мета полягає у виявленні ключових аспектів цього інноваційного підходу, спрямованого на поліпшення діагностики, прогнозування та лікування захворювань. Необхідно не лише провести огляд актуальних досліджень у цій галузі, а й внести власний внесок у розвиток методології, шляхом розгляду та аналізу нових технологічних і наукових підходів. Конкретні завдання роботи включають в себе аналіз існуючих методів машинного навчання, що застосовуються в медичній сфері, виявлення їхніх переваг і обмежень. Необхідно також розглянути останні тенденції в галузі нейронних моделей, зокрема використання глибокого навчання, рекурентних нейронних мереж і технік передачі навчання. Робота буде прагнути до розуміння того, як ці методи можуть бути ефективно адаптовані і застосовані до різноманітних типів медичних даних. У рамках дослідження необхідно також поставити перед собою завдання вивчити реальні приклади успішної реалізації нейронних моделей у клінічній практиці, щоб виявити конкретні

випадки їхнього застосування, а також з'ясувати результати та переваги, досягнуті лікарями та дослідниками.

Для досягнення поставленої мети дослідницької роботи необхідно вирішити низку ключових завдань. Насамперед потрібно провести ретельний огляд літератури, спрямований на виявлення основних тенденцій у розробленні нейронних моделей для аналізу медичних даних. Це включає в себе аналіз різних методів машинного навчання, що застосовуються в медицині, і визначення їхніх сильних і слабких сторін. Додатково, необхідно вивчити сучасні технології в галузі нейронних мереж, включно з глибоким навчанням і рекурентними нейронними мережами, з метою оцінки їхньої застосовності до аналізу різноманітних медичних даних. Це вимагає глибокого розуміння особливостей даних, які використовуються в медичних дослідженнях, і розроблення стратегій адаптації наявних методів машинного навчання до специфіки медичних завдань. Далі, на основі отриманих знань, необхідно визначити оптимальні методи навчання моделей і підходи до їхньої валідації в контексті медичних даних. Акцент буде зроблено на виявленні принципів, що сприяють стійкості та надійності результатів, а також можливості їх перенесення на практику. Крім того, дослідження спрямоване на аналіз успішних кейсів застосування нейронних моделей у клінічній практиці, з метою виокремлення конкретних сценаріїв, у яких ці моделі проявили видатну ефективність. Це вимагає систематичного аналізу клінічних даних, результатів лікування та зворотного зв'язку лікарів. Нарешті, в рамках дослідження необхідно виявити і проаналізувати ключові фактори, що впливають на успішне впровадження нейронних моделей у медичну практику. Це включає в себе розгляд технічних, етичних і регуляторних аспектів, а також оцінку потенційних перешкод і шляхів їх подолання. Таким чином, для досягнення поставленої мети роботи необхідне розв'язання складних завдань, що включають у себе аналіз літератури,



розробку методології, адаптацію технологій, аналіз клінічних випадків і виділення чинників успішної інтеграції нейронних моделей у медичну практику.

Об'єктом дослідження поточної роботи є процес розроблення та машинного навчання нейронних моделей, призначених для аналізу медичних даних. У центрі уваги перебуває складний і багатогранний характер даних, притаманних медичним дослідженням, таких як медичні зображення, генетичні дані, клінічні записи та інші. Дослідження спрямоване на розуміння того, які особливості та виклики існують у сфері медичних даних, а також яким чином нейронні моделі можуть бути найефективніше застосовані для їхнього аналізу. У процесі дослідження розглядаються різні аспекти розроблення нейронних моделей, починаючи з вибору методів машинного навчання, включно з глибоким навчанням і рекурентними нейронними мережами, і закінчуючи оптимізацією цих моделей для конкретних медичних завдань. Об'єктом уваги також є адаптація наявних методів машинного навчання до особливостей медичних даних з огляду на їхню високу розмірність, динамічність і специфіку біологічних процесів. У рамках дослідження об'єктом також є успішні кейси використання нейронних моделей у клінічній практиці. Аналізується, як ці моделі впроваджуються в реальне медичне обслуговування, яких результатів вони досягають і як вони взаємодіють із лікарською спільнотою. Крім того, об'єктом уваги дослідження є різні фактори, що впливають на успішне інтегрування нейронних моделей у медичну практику. Це включає в себе аналіз технічних, етичних і регуляторних аспектів, а також виявлення можливих перешкод і шляхів їх подолання. Таким чином, об'єктом дослідження є весь комплекс процесів, пов'язаних із розробкою та застосуванням нейронних моделей для аналізу медичних даних, з

урахуванням їхньої специфіки та потенціалу в контексті поліпшення діагностики, прогнозування та лікування захворювань.

Предметом дослідження поточної роботи є сутність і характер процесу розроблення та машинного навчання нейронних моделей, що застосовуються для аналізу медичних даних. У центрі уваги перебуває не тільки технічний бік створення цих моделей, а й особливості даних, що використовуються в медичних дослідженнях. Предметом аналізу є структура і різноманітність медичних даних, таких як зображення, генетичні послідовності, клінічні записи та інші, і їхня специфіка в контексті застосування методів машинного навчання. У межах дослідження розглядається процес адаптації наявних методів машинного навчання, включно з глибоким навчанням і рекурентними нейронними мережами, до особливостей медичних даних. Предметом уваги також є вибір та оптимізація методів навчання з огляду на вимоги точності, інтерпретованості та застосовності результатів у медичній практиці. У процесі аналізу успішних кейсів застосування нейронних моделей у клінічній практиці предметом дослідження стає не тільки сам факт успішної інтеграції, а й результати, досягнуті в контексті поліпшення діагностики, прогнозування захворювань та ефективності лікування. Лікарський зворотний зв'язок, взаємодія з системами охорони здоров'я і пацієнтами також входять у предмет аналізу. Крім цього, предметом дослідження є фактори, що впливають на успішність інтеграції нейронних моделей у медичне середовище. Це охоплює технічні аспекти, етичні питання, регулювання та взаємодію з медичною спільнотою. Таким чином, предметом дослідження є комплекс феноменів, пов'язаних із використанням нейронних моделей для аналізу медичних даних, включно з технологічними, методологічними та соціальними аспектами, спрямованими на підвищення ефективності та точності медичної діагностики та лікування.

Методи дослідження в даній роботі будуються на комплексному підході, що охоплює як теоретичні, так і практичні аспекти аналізу процесу розроблення та машинного навчання нейронних моделей для аналізу медичних даних. На початку дослідження застосовано метод літературного огляду для систематизації та аналізу наявних наукових публікацій, статей і досліджень у галузі медичного машинного навчання. Цей метод дозволяє отримати огляд сучасних тенденцій, проблем і досягнень у цьому контексті, виявивши ключові методи та підходи. Далі, використовується аналітичний метод для детального розгляду технічних характеристик нейронних моделей, їх застосовності до різних типів медичних даних і можливостей адаптації під конкретні завдання. Аналіз також включає в себе оцінку переваг і обмежень обраних методів машинного навчання в контексті медичних даних. Основним методом дослідження є емпіричний підхід, що включає в себе проведення експериментів з різними моделями машинного навчання на реальних медичних даних. Це включає в себе розробку прототипів нейронних моделей, їх навчання на різноманітних наборах даних і ретельний аналіз результатів. Метод емпіричного дослідження також включає в себе аналіз успішних кейсів використання нейронних моделей у реальних клінічних сценаріях. Додатково, в процесі дослідження застосовуються методи порівняльного аналізу для оцінки ефективності різних підходів до розробки та використання нейронних моделей у медичній діагностиці та прогнозуванні. Ці методи взаємодіють з метою забезпечення всеосяжного розуміння технологічних, методологічних і прикладних аспектів застосування нейронних моделей у медичних дослідженнях. Такий комплексний підхід забезпечує ґрунтовне дослідження, спрямоване на досягнення поставлених цілей і розв'язання поставлених завдань.

Практична значимість дослідницької роботи проявляється в кількох аспектах, орієнтованих на поліпшення ефективності та результативності застосування нейронних моделей в аналізі медичних даних.

По-перше, результати дослідження нададуть лікарям і дослідникам у галузі медичного машинного навчання нові практичні інсайти та методологічні рекомендації. Це дасть їм змогу краще розуміти можливості та обмеження застосування нейронних моделей залежно від конкретних характеристик медичних даних, що, своєю чергою, сприяє точнішій діагностиці та прогнозуванню захворювань.

По-друге, практична значимість роботи проявиться в розробці та адаптації методів машинного навчання для конкретних клінічних сценаріїв. Це може сприяти створенню індивідуалізованих і точних моделей, що відповідають особливостям кожного конкретного випадку, що важливо для персоналізованої медицини і точного прогнозування перебігу захворювань. Третій аспект практичної значимості пов'язаний із можливістю впровадження розроблених нейронних моделей у реальні медичні практики. Успішне використання таких моделей може підвищити ефективність діагностики, поліпшити передбачення ризиків захворювань і оптимізувати вибір методів лікування. Це, своєю чергою, сприяє підвищенню якості медичного обслуговування, зменшенню витрат і підвищенню пацієнтського задоволення. Таким чином, практична значимість цієї роботи полягає в тому, щоб надати конкретні інструменти та рекомендації для впровадження нейронних моделей у реальні клінічні сценарії, що сприятиме покращенню якості турботи про здоров'я пацієнтів та оптимізації процесів у медичній практиці.

Наукова новизна роботи виражається в кількох ключових аспектах, що роблять істотний внесок у галузь розроблення та застосування нейронних моделей для аналізу медичних даних.

По-перше, дослідження пропонує оригінальний підхід до адаптації методів машинного навчання до складних і різноманітних даних, характерних для медичних досліджень. Робота спрямована на виявлення оптимальних стратегій і методів навчання нейронних моделей з урахуванням специфіки медичних даних, таких як медичні зображення, генетичні послідовності та клінічні параметри.

По-друге, наукова новизна проявляється в глибокому аналізі успішних кейсів використання нейронних моделей у реальних клінічних сценаріях. Робота виокремлює конкретні приклади застосування цих моделей у контексті діагностики, прогнозування та терапії захворювань, що доповнює наукові знання в галузі ефективності таких технологій у клінічній практиці.

Третій аспект наукової новизни пов'язаний із детальним розглядом технологічних, етичних і регуляторних аспектів впровадження нейронних моделей у медичне середовище. Робота пропонує інтегрований підхід до вирішення етичних питань, а також розглядає проблеми в галузі безпеки та конфіденційності даних у контексті застосування сучасних методів машинного навчання. Нарешті, наукова новизна роботи проявляється в наданні практичних рекомендацій та інсайтів, орієнтованих на лікарське співтовариство і дослідників, що створює фундамент для подальших розробок і поліпшень у галузі медичного машинного навчання. Таким чином, ця робота представляє собою унікальний внесок у розширення знань і підвищення ефективності застосування нейронних моделей у медицині.

Подальші напрямки дослідження, ґрунтуючись на результатах цієї роботи, можуть бути спрямовані на кілька перспективних аспектів у сфері розроблення та застосування нейронних моделей для аналізу медичних даних. Перш за все, варто розглянути можливості подальшого вдосконалення методів навчання нейронних моделей у контексті медичних даних. Це включає в себе дослідження нових алгоритмів навчання, оптимізацію

параметрів моделей і врахування особливостей різних типів медичних даних для досягнення більш високої точності та інтерпретованості результатів. Важливим напрямком подальших досліджень є розширення сфери застосування нейронних моделей у медичній практиці. Це може включати в себе поглиблений аналіз можливостей моделей для спеціалізованих галузей медицини, таких як онкологія, неврологія або кардіологія, з урахуванням унікальних характеристик кожної галузі. Додатково, дослідження можуть бути спрямовані на поліпшення прийняття рішень лікарів під час використання нейронних моделей. Це включає в себе аналіз того, як результати моделей можуть інтегруватися в клінічну практику, а також яким чином надавати лікарям інформацію, необхідну для прийняття обґрунтованих рішень. Одним із важливих напрямків дослідження може стати розширення розуміння етичних і юридичних питань, пов'язаних із використанням нейронних моделей у медичних додатках. Це охоплює розробку стандартів безпеки даних, забезпечення прозорості та розуміння алгоритмів, а також ефективні методи регулювання цієї галузі. Нарешті, подальші дослідження можуть бути спрямовані на розробку більш комплексних моделей, здатних інтегрувати різні типи медичних даних для створення більш повного образу здоров'я пацієнта. Це може включати в себе мультимодальні моделі, здатні обробляти й аналізувати інформацію з різних джерел. Таким чином, подальші дослідження в галузі розроблення нейронних моделей для медичних застосувань є перспективним полем для розширення знань і застосування передових технологій у медицині.

Структура магістерської роботи: робота складається з трьох розділів, об'єм роботи – 89 сторінок, робота містить 9 рисунків, перелік використаних джерел має 17 посилань.

## РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

### 1.1 Аналіз предметної області

У сучасному світі медичні дані стають дедалі об'ємнішими і складнішими. У цьому контексті використання нейронних моделей в обробці медичних даних є важливим і перспективним підходом, здатним перетворити і поліпшити багато аспектів охорони здоров'я [1].

Одним із ключових аспектів використання нейронних моделей у медичних даних є можливість автоматизації процесу діагностики. Нейронні мережі можуть навчатися розпізнавати патології на зображеннях, таких як рентгенівські знімки або знімки магнітно-резонансної томографії (МРТ), що прискорює і покращує точність діагностики [2].

Нейронні моделі дозволяють створювати персоналізовані моделі здоров'я пацієнта на основі його медичних даних. Це включає в себе передбачення ризику розвитку різних захворювань, адаптацію лікування під індивідуальні особливості організму та оптимізацію рекомендацій щодо способу життя.

У галузі медичного зображення нейронні моделі можуть бути використані для сегментації органів, виявлення пухлин і аномалій. Це допомагає лікарям проводити більш точні та швидкі аналізи, а також підвищує чутливість методів діагностики. Нейронні моделі здатні аналізувати різні медичні показники та передбачати розвиток захворювань. Це може бути особливо корисним для раннього виявлення ризиків і вжиття заходів щодо їх запобігання.

Медичні дані часто представлені в текстовій формі, такій як медичні записи, звіти та історії хвороби. Нейронні моделі в галузі обробки природної

мови (NLP) дають змогу витягувати смислову інформацію з таких текстів, автоматизуючи процеси аналізу даних.

Найчастіше в медичній сфері дані надходять з різних джерел – лабораторних аналізів, зображень, датчиків здоров'я і багатьох інших. Нейронні моделі можуть ефективно інтегрувати ці дані, що створює комплексний огляд стану пацієнта. Аналіз даних з використанням нейронних моделей може допомогти у визначенні найкращих методів лікування для конкретних пацієнтів. Також, моделі можуть використовуватися для прогнозування ефективності терапії, що важливо для оптимізації лікувальних стратегій.

Нейронні моделі мають здатність обробляти великі обсяги даних і виявляти закономірності, що допомагає скорочувати помилки в діагностиці, особливо в порівнянні з традиційними методами. Автоматизація багатьох процесів з використанням нейронних моделей може значно скоротити час, що витрачається на аналіз медичних даних, а також оптимізувати використання медичних ресурсів, роблячи процеси більш ефективними.

Використання нейронних моделей у медичних даних відкриває нові можливості для наукових досліджень. Моделі можуть допомогти виявити нові залежності та взаємозв'язки, що сприяє розвитку більш точних методів діагностики та лікування. Медичні дані часто характеризуються високим ступенем складності та варіабельності. Нейронні моделі, завдяки своїй гнучкості та здатності обробляти складні структури, можуть краще справлятися з такою складністю, ніж традиційні методи.

Розвиток мобільних технологій дозволяє створювати додатки та пристрої, що використовують нейронні моделі для моніторингу здоров'я, діагностики та надання рекомендацій. Це відкриває шлях до ширшого доступу до медичних послуг. Ефективне використання нейронних моделей може допомогти знизити витрати на охорону здоров'я, тому що дозволить



точніше і швидше проводити діагностику і лікування, запобігаючи непотрібним процедурам і зменшуючи ризики. Нейронні моделі можуть бути навчені на нових даних, що забезпечує їх постійне оновлення та поліпшення з плином часу. Це дає їм змогу адаптуватися до нових вимог і тенденцій у галузі охорони здоров'я.

Нейронні моделі знаходять широке застосування в наукових дослідженнях, включно з вивченням генетичних факторів, молекулярних механізмів хвороб і створенням інноваційних методів лікування. Однак, незважаючи на всі перераховані переваги, існують і виклики, пов'язані з використанням нейронних моделей у медичних даних. Насамперед, це питання конфіденційності та безпеки даних пацієнтів. Також необхідно звертати увагу на прозорість роботи моделей та їхню інтерпретованість для лікарів, щоб забезпечити довіру до прийнятих рішень. Загалом, використання нейронних моделей в обробці медичних даних надає унікальні можливості для поліпшення точності діагностики, персоналізації лікування і поліпшення загального рівня охорони здоров'я. Розвиток цієї галузі обіцяє нові перспективи для прогресу в медичній науці та практиці, створюючи більш ефективні та інтелектуальні підходи до догляду за пацієнтами.

## 1.2 Огляд загальних понять про штучні нейронні мережі

Штучні нейронні мережі (ШНМ) є сучасним класом алгоритмів машинного навчання, натхненних структурою та функціонуванням біологічних нейронних мереж. Розуміння основних концепцій нейронних моделей є важливим кроком для опанування цієї теми (рис. 1.1) [3-5].

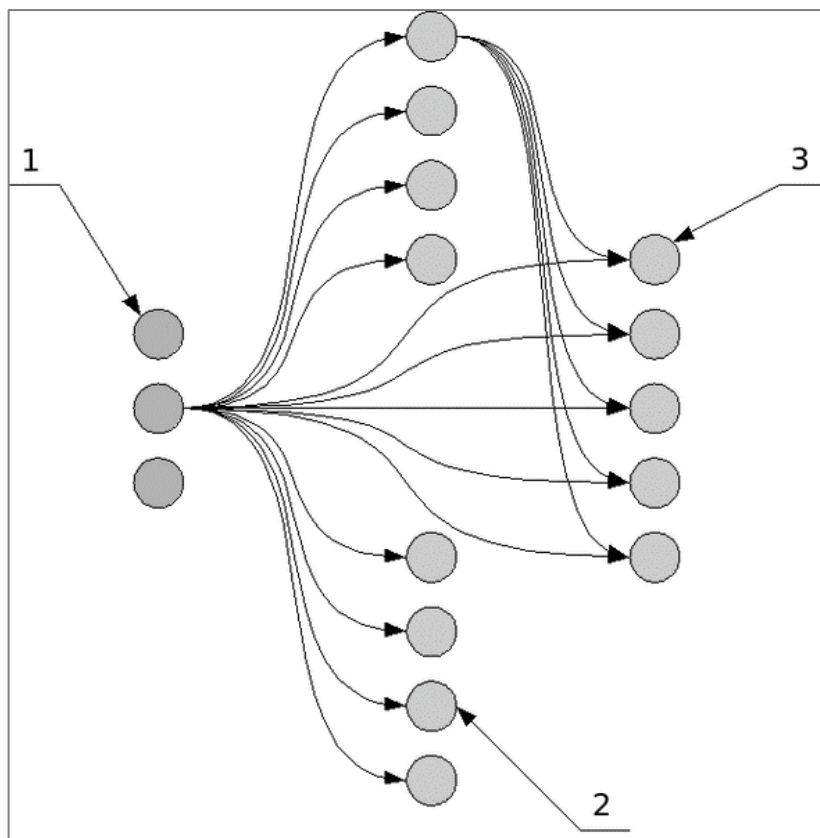


Рисунок 1.1 – Схематичне зображення штучної нейронної мережі

У центрі нейронних мереж знаходиться ідея нейрона, основного будівельного блоку. Нейрон приймає вхідні сигнали, зважує їх з певними вагами і передає результат через функцію активації. Це створює вихідний сигнал, який є входом для наступного шару або виходом мережі. Ваги нейронів представляють собою параметри, які налаштовуються в процесі навчання. Вони визначають важливість кожного входу для вихідного сигналу. Оновлення ваг відбувається під час навчання з метою мінімізації помилки.

Функція активації застосовується до виходу нейрона і визначає, чи повинен нейрон активуватися і передавати сигнал далі. Популярні функції активації включають сигмоїдальну, гіперболічну тангенс і ReLU (Rectified Linear Unit). Нейрони групуються в шари. Вхідний шар отримує дані,

приховані шари обробляють інформацію, а вихідний шар формує остаточний результат. Кожен шар має свої ваги та функції активації.

Функція втрат вимірює різницю між передбаченим і фактичним результатами. Завдання навчання полягає в мінімізації цієї функції, що досягається підбором оптимальних ваг. Навчання нейронної мережі передбачає передачу даних через мережу, обчислення помилки, оновлення ваг для зменшення помилки і повторення цього процесу багаторазово. Розрізняють різні методи навчання, включаючи градієнтний спуск і його варіації.

Архітектура нейронної мережі визначає кількість шарів, кількість нейронів у кожному шарі та їхні зв'язки. Глибокі нейронні мережі (ГНМ) мають безліч шарів, що дає їм змогу виявляти складніші залежності в даних.

Глибоке навчання – це підхід до машинного навчання, заснований на використанні глибоких нейронних мереж. ГНМ здатні автоматично вивчати уявлення ознак на різних рівнях абстракції, що робить їх ефективними для складних завдань, таких як розпізнавання зображень і обробка природної мови.

ЗНМ (згорткові нейронні мережі) розроблені для обробки структурованих даних, таких як зображення. Вони використовують згорткові шари для виділення локальних ознак і пулінгові шари для зменшення розмірності. РНМ (рекурентні нейронні мережі) призначені для роботи з послідовними даними, такими як текст або тимчасові ряди. Вони мають пам'ять, що дає їм змогу враховувати попередні стани під час опрацювання нових вхідних даних.

Автоенкодери – це нейронні мережі, навчені відтворювати свої вхідні дані. Вони складаються з кодувальника, який стискає дані, і декодувальника, який відновлює їх. Автоенкодери можуть використовуватися для вилучення важливих ознак. Перенесення навчання – це метод, за якого заздалегідь

навчені моделі застосовуються до нового завдання. Це дає змогу використовувати знання, отримані на одному завданні, для розв'язання іншого, навіть якщо є обмежені дані для нового завдання.

Регуляризація включає в себе методи, спрямовані на запобігання перенавчання моделі. Dropout – це один із популярних методів регуляризації, при якому випадкові нейрони відключаються в процесі навчання.

Гіперпараметри – це параметри моделі, які не навчаються в процесі навчання, але мають бути встановлені и до його початку. Приклади включають в себе швидкість навчання, кількість шарів і розмір батча. Трансформери представляють собою архітектуру нейронних мереж, розроблену для обробки послідовних даних, таких як текст. Вони використовують механізм уваги для ефективного аналізу контексту.

NLP (природня мова) – це галузь машинного навчання, присвячена обробці та розумінню природної мови. NLP моделі, такі як BERT і GPT, привертають увагу своєю здатністю до роботи з текстовою інформацією. Навчання без вчителя передбачає навчання моделі на нерозмічених даних. Кластеризація та асоціативні правила – це приклади методів навчання без учителя.

Розвиток нейронних мереж також порушує питання етики. Прозорість алгоритмів, захист приватності та питання справедливості стають важливими аспектами в галузі штучного інтелекту. Інтерпретованість нейронних мереж – це важливий напрям досліджень. Забезпечення можливості розуміння, як модель ухвалює рішення, є ключовим аспектом їхнього застосування в реальних задачах.

Нейронні мережі продовжують бути об'єктом інтенсивних досліджень. Можливості для поліпшення ефективності, навчання на невеликих об'ємах даних і підвищення інтерпретованості залишаються відкритими напрямками. Отже, нейронні мережі представляють собою потужний інструмент

машинного навчання з широким спектром застосувань. Розуміння основних концепцій і тенденцій у цій галузі є важливим кроком для ефективного використання та розвитку цієї технології в майбутньому.

### 1.3 Нейронні моделі

Нейронні моделі, включно з глибокими нейронними мережами (ГНМ), стали невід'ємною частиною машинного навчання, але, як і будь-яка технологія, у них є як переваги, так і недоліки [4-6].

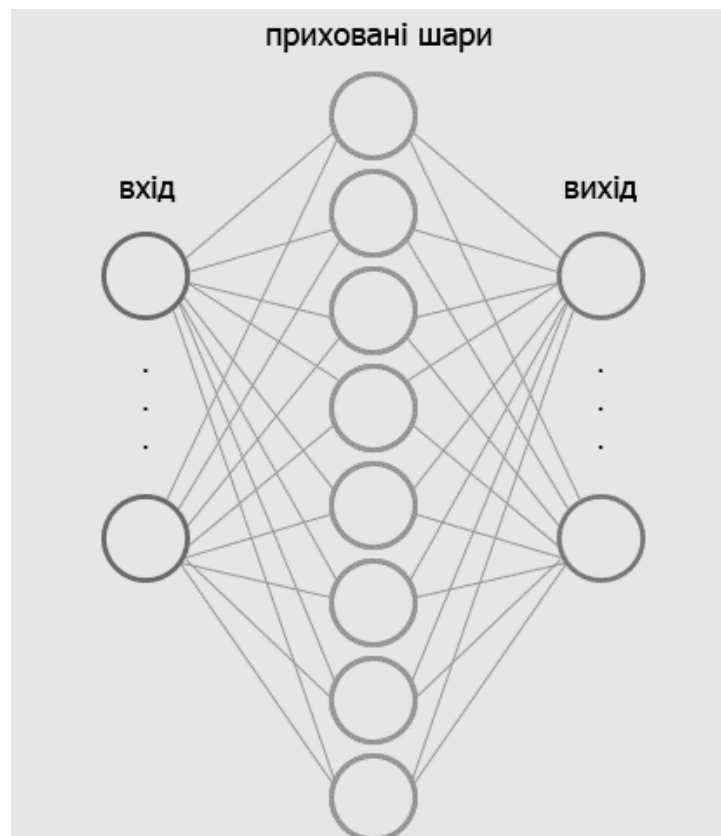


Рисунок 1.2 – Нейронна модель. Вхідні, приховані та вихідні шари

#### 1.3.1 Переваги нейронних моделей

Переваги нейронних моделей [1, 3, 7]:

1) обробка складних даних. Однією з ключових переваг нейронних моделей є їхня здатність обробляти складні та багатовимірні дані. ГНМ, наприклад, можуть автоматично вивчати ознаки на різних рівнях абстракції, що робить їх ефективними для задач розпізнавання образів, класифікації та регресії;

2) автоматичне вилучення ознак. Нейронні моделі здатні автоматично витягувати важливі ознаки з даних у процесі навчання. Це усуває необхідність ручного створення ознак і дозволяє моделі самостійно визначати значущі характеристики вхідних даних;

3) універсальність. Нейронні моделі можуть бути застосовані до різних типів даних, включно із зображеннями, звуком, текстом і часовими рядами. Ця універсальність робить їх важливим інструментом у різних галузях, як-от комп'ютерний зір, обробка природної мови (NLP) та багато інших;

4) глибоке навчання для складних завдань. Глибокі нейронні мережі стали ключовим компонентом у вирішенні складних завдань, таких як розпізнавання мови, машинний переклад та ігри. Їхня здатність моделювати складні залежності в даних дає їм змогу ефективно справлятися з такими завданнями;

5) перенос навчання. Нейронні моделі часто використовують перенесення навчання, що дозволяє використовувати знання, отримані на одному завданні, для поліпшення продуктивності на іншому. Це особливо корисно у випадках, коли доступна обмежена кількість розмічених даних;

6) здатність до навчання на великих обсягах даних. Нейронні моделі, особливо глибокі, можуть показувати хороші результати за наявності великих обсягів даних. Це пов'язано з їхньою здатністю навчатися на мільйонах або навіть мільярдах прикладів.

### 1.3.2 Недоліки нейронних моделей

Недоліки нейронних моделей [1, 3, 7]:

1) неясність ваг та інтерпретованість. Одним із серйозних недоліків нейронних моделей є їхня складність і непрозорість. Ваги, одержувані в результаті навчання, можуть бути важкими для інтерпретації, що ускладнює розуміння причинно-наслідкових зв'язків у даних;

2) вимога до великих обсягів даних. Глибокі нейронні мережі, особливо якщо вони містять багато параметрів, вимагають великих обсягів даних для ефективного навчання. В іншому випадку вони можуть зіткнутися з проблемою перенавчання;

3) час навчання та обчислювальна складність. Навчання глибоких моделей може займати багато часу, особливо на великих наборах даних, і вимагає значних обчислювальних ресурсів. Це може стати значним обмеженням при обмежених ресурсах;

4) ризик перенавчання. Глибокі нейронні мережі можуть бути схильні до перенавчання, особливо якщо кількість даних обмежена. Це може призвести до того, що модель буде добре працювати на тренувальних даних, але не зможе узагальнюватися на нові дані;

5) неефективність на малих даних. У ситуаціях, де доступна обмежена кількість розмічених даних, нейронні моделі можуть показувати низьку продуктивність. У таких випадках перенесення навчання може бути обмежене;

6) залежність від архітектури та гіперпараметрів. Вибір правильної архітектури та гіперпараметрів є важким завданням. Неправильний вибір може призвести до повільного навчання, нестабільності та поганих результатів;

7) етичні та соціальні питання. У розвитку нейронних мереж виникають етичні питання, пов'язані з прозорістю моделей, захистом приватності даних і можливістю появи упередженості в рішеннях алгоритмів;

8) відсутність загального розуміння. Нейронні мережі можуть розглядатися як «чорні скриньки», і їхнє функціонування не завжди піддається легкому поясненню. Це створює складнощі в їхньому прийнятті та розумінні неспеціалістами;

9) сприйнятливність до шуму. Нейронні моделі можуть бути сприйнятливі до шуму в даних, що може призвести до неправильних прогнозів або узагальнень;

10) складність налаштування гіперпараметрів. Вибір правильних гіперпараметрів для нейронних мереж може бути складним завданням, що вимагає численних експериментів і обчислювальних ресурсів.

Насамкінець, нейронні моделі надають сучасний інструмент для розв'язання різних завдань машинного навчання, але їхнє застосування потребує уважного розгляду та оцінки переваг і недоліків у контексті конкретного завдання та доступних ресурсів.

#### 1.4 Сучасний стан використання нейронних моделей в медичній галузі

Стаття [1] фокусується на розв'язанні проблеми нестачі онлайн-медичних ресурсів, спричиненої сплеском онлайн-консультацій і неефективним використанням великого обсягу медичних даних.

У статті пропонується комплексна програма онлайн-інтелектуальної консультації. Першим етапом є збір гетерогенних даних з різних джерел з використанням декількох каналів. Потім використовується технологія глибокого навчання для створення медичного графа знань, який зберігається у вигляді сховища знань. Далі створюється інтелектуальна модель



автоматичної консультації та модель запиту медичних знань, використовуючи технологію обробки природної мови для аналізу семантики запитань користувачів. Відповіді витягуються зі сховища знань з використанням логіки семантичного запиту.

У кінцевому підсумку реалізується інтелектуальна модель запитань, що охоплює всі хвороби на основі медичного графа знань. Запропоноване рішення дає змогу задовольнити потреби пацієнтів в онлайн-консультаціях, знизити робоче навантаження медичного персоналу, надати послуги підтримки клінічних рішень, підвищити точність інтелектуального діагностування та знизити вартість консультацій.

Дослідження [2] стосується проблеми використання медичних великих даних, які стають важливими у зв'язку з накопиченням величезного обсягу медичної інформації в лікарняній практиці.

Автори досліджують архітектуру федеративного навчання з метою створення моделі виявлення захворювань на основі даних крові, форматуваних відповідно до стандартів цитометрії потоку, для полегшення багатолокальних медичних досліджень. Характеристики великих даних у крові та проблеми конфіденційності під час обміну даними про пацієнтів ускладнюють збір та обмін даними в центральному місці для побудови узагальненої моделі виявлення.

У статті представлено дослідження в стадії розробки під назвою FedM-FCM, федеративне навчання аналізу цитометрії потоку з використанням конвеєра від джерел даних до зсуву домену у федеративній моделі навчання. У структуру FedM-FCM входять основні компоненти: представлення даних багатовимірної цитометрії потоку, застосування моделей нейронних мереж на основі представлення даних, а також агрегація і розподіл параметрів навчання між лікарнями, що беруть участь, без обміну даними.

Стаття [3] обговорює використання штучних нейронних мереж у різних галузях, таких як інтелектуальні дорожні мережі, Інтернет речей і розумні медичні системи. Нейронні мережі широко застосовуються через їхню здатність обробляти великі обсяги даних паралельно, зберігати інформацію розподіленим чином і самоорганізовуватися та самонавчатися.

Технологія хмарних обчислень додатково сприяє розвитку додатків нейронних мереж. Однак передача управління правами на управління даними в хмару часто стикається з серйозними питаннями безпеки та конфіденційності, оскільки призначені для користувача дані часто містять чутливу інформацію. У медичній галузі реалізація алгоритмів класифікації з дотриманням приватності критично важлива для забезпечення конфіденційності електронних медичних послуг діагностики. Поточні схеми захисту конфіденційності медичних попередніх діагнозів, засновані на гомоморфному шифруванні, накладають значні обчислювальні та комунікаційні навантаження на користувачів і сервери. У даній статті запропоновано ефективну схему захисту конфіденційності медичних попередніх діагнозів на основі нейронних мереж і шифрування функції внутрішнього добутку, яка забезпечує конфіденційність користувачів під час попередньої діагностики за мінімального обчислювального та комунікаційного навантаження.

Стаття [4] обговорює проблему витоку конфіденційної інформації в процесі обміну медичними даними при широкому використанні технології великих даних у медичній сфері.

Автори пропонують метод для розв'язання цієї проблеми, розділяючи поведінку користувачів під час доступу до даних на два етапи: вибір об'єктів роботи та вибір інформації, яку відвідує користувач. У першому етапі оцінюють ступінь плутанини при виборі користувачем об'єктів доступу з використанням інформаційної ентропії. Потім цю ентропію використовують

як вхідні дані для штучної нейронної мережі (BP neural network) для кількісного оцінювання ризику поведінки під час доступу.

Отриманий ризик зворотно подається в довірений центр, який розраховує значення довіри користувача відповідно до принципу, що значення довіри користувача зворотно корелює з ризиком. Якщо значення довіри користувача не відповідає вимогам призначеної ролі, роль не буде призначена. Також, якщо квота ризику вичерпана, доступ буде заблоковано. Таким чином, запропонований підхід реалізує контроль доступу до медичних даних, даючи змогу ефективно управляти ризиками конфіденційності та запобігати несанкціонованому доступу.

Стаття [5] фокусується на важливості правильної та безперебійної роботи медичних пристроїв для пацієнтів і медичного сектору загалом.

Однак, незважаючи на різні протоколи і процедури, що застосовуються виробниками медичних пристроїв, мікросхеми зазвичай виготовляються іншими сторонами. У процесі виробництва напівпровідників, що використовуються в інтегральних схемах, характеристики компонентів можуть відрізнитися від партії до партії, і навіть зразки всередині однієї партії можуть мати різні характеристики. Для застосувань у критично важливих сферах життя вибір найбільш підходящих компонентів має вирішальне значення. У цій статті пропонується новий метод для перевірки якості транзисторів у напівпровідниковій індустрії.

Графіки I-V (струм-напруга) компонентів оцінюються кількома згортковими нейронними мережами (CNN) з використанням візуальних даних, аналогічних оцінці експертів. Потім ці архітектури машинного навчання (ML) використовують техніку ансамблю безлічі моделей, де негативний результат однієї архітектури перевизначає голосування інших архітектур, щоб забезпечити дуже суворий контроль якості. Запропонований

метод тестується на транзисторах CMOS, і результати можна порівняти з результатами експертів із 10-річним досвідом у галузі.

Дослідження [6] вказує на широке використання різних систем медичної та охорони здоров'я інформації останніми роками, що призвело до накопичення великого обсягу даних, пов'язаних із медичною галуззю, у лікарнях. Сплеск мобільної медицини призвів до все більш цифрового характеру медичної інформації, і медична індустрія вступила в еру великих даних. Ці медичні дані є вкрай цінними для діагностики та лікування захворювань, а також для медичних досліджень.

На жаль, більшість лікарень наразі обмежуються тільки збором і зберіганням медичних даних пацієнтів, але не проводять глибокий аналіз і використання цих даних. Автори статті пропонують використання методів дата-майнінгу для виявлення закономірностей у масових медичних даних, надаючи новий метод отримання знань медичним персоналом.

Серед методів дата-майнінгу в медицині виокремлюються інтелектуальні методи, такі як асоціативні правила, штучні нейронні мережі та теорія наближених множин, які демонструють унікальні переваги. Серед них асоціативне правило-майнінг здатне аналізувати частотні відношення набору даних у наборі транзакцій із заданого набору даних і наборів транзакцій.

Стаття [7] розглядає пропозицію поліпшеної моделі інтелектуального медичного діагнозу, що інтегрує традиційну китайську медицину (ТКМ) і західну медицину (ЗМ) з використанням методів дата-майнінгу.

У статті пропонується підхід, який вводить дані діагностики у дві підсистеми для навчання і тестування.

Алгоритм Apriori був поліпшений для майнінгу даних з діагностики та лікування в ТКМ з метою отримання більш повної інформації для діагностики. Потім використано оптимізовану генетичним алгоритмом (GA)

нейронну мережу зворотного розповсюдження (BP) для опрацювання реальних зразків і підвищення швидкості медичної діагностики. Зрештою, створено прототип клінічної системи діагностики, що об'єднує обидва методи.

Ця система застосовується в конкретній підтримці прийняття рішень і показує хороший ефект майнінгу. Передбачені дані значною мірою узгоджуються з клінічною практикою. У підсумку запропонований метод доводить свою здійсненність і корисність для зниження клінічних медичних помилок.

Дослідження [8] описує роботу, пов'язану зі зростаючим обсягом електронних медичних даних у контексті просування медичної інформатизації. У статті запропоновано метод розв'язання проблеми надмірності, відсутності семантики і неоднозначності в текстовій частині електронних медичних даних з використанням комбінації методу векторизації слів і теорії глибокого навчання.

Автори використовують метод векторизації слів для представлення медичних текстів у вигляді матриці фіксованої довжини. Потім вони об'єднують модель довгострокової короткострокової пам'яті (LSTM) з моделлю Уоп. Зрештою, застосовується метод інтегративного навчання для створення всебічної моделі навчання, яка досягає хороших результатів у класифікації тексту електронних медичних даних. Таким чином, стаття фокусується на розробці методу, що використовує векторизацію слів і глибоке навчання для обробки текстової інформації в електронних медичних даних з метою поліпшення їхньої класифікації.

Стаття [9] говорить про необхідність дослідження та оптимізації процесу класифікації медичних зображень з використанням інноваційних методів машинного навчання. Традиційні підходи до класифікації медичних

зображень досягли своєї межі, і їхнє застосування потребує значного часу та зусиль для вилучення та вибору характеристик для класифікації.

Автори статті пропонують новий метод, заснований на глибоких нейронних мережах, зокрема, на згорткових нейронних мережах (CNN), які продемонстрували свій потенціал у вирішенні багатьох завдань класифікації. Вони зазначають, що точна класифікація медичних зображень значно допомагає в клінічній практиці та терапії. Наприклад, діагностика пневмонії на рентгенівських знімках є важливим методом, але вимагає кваліфікації радіологів, що може бути важко і дорого в деяких регіонах. Таким чином, основним мотивом авторів є оптимізація класифікації медичних зображень з використанням глибоких нейронних мереж, таких як DNN (глибокі нейронні мережі), ANN (штучні нейронні мережі) і CNN (згорткові нейронні мережі).

Стаття [10] обговорює застосування глибоких нейронних мереж (DNN) у сфері медичної діагностики з використанням різних методів зображень, таких як рентгенографія, ультразвукові дослідження, комп'ютерна томографія та магнітно-резонансна томографія. Глибокі нейронні мережі є потужним інструментом для підтримки лікарів у діагностиці захворювань пацієнтів, завдяки своїм видатним результатам у завданнях класифікації зображень.

Для подальшого розвитку цієї галузі автори пропонують двофазний алгоритм, названий DNNDeepereningPruning, здатний автоматично генерувати компактні архітектури DNN за заданою базою даних. У першій фазі, або фазі поглиблення, алгоритм нарощує DNN, додаючи блоки шарів залишкових блоків один за одним доти, доки модель не перенавчиться на наданих даних. У другій фазі, або фазі обрізки, алгоритм обрізає створену модель DNN з першої фази, щоб отримати DNN з невеликою кількістю операцій з плаваючою крапкою, керованою уподобаннями користувача. Запропонований алгоритм об'єднує дві окремі області пошуку та обрізки

архітектури DNN в єдиний фреймворк і тестується на двох наборах медичних зображень із задовільними результатами.

Стаття [11] фокусується на проблемі ефективного прогнозування навантаження на мережеве з'єднання в медичних великих даних з використанням традиційних алгоритмів. Проблеми, такі як високі втрати пакетів, енергетичні втрати, тривалий час обробки, повільна швидкість потоку даних і недостатня стійкість до атак, вимагають нового підходу.

У статті пропонується стрімінговий алгоритм для обробки незбалансованих медичних великих даних на основі згорткової нейронної мережі (CNN). Запропонований алгоритм включає два етапи: на першому етапі будується модель декомпозиції-прогнозування з використанням хвильового аналізу та аналізу нейронних мереж для прогнозування навантаження на мережеве з'єднання.

На другому етапі, виходячи із ситуації з навантаженням на мережеве з'єднання, проводиться аналіз структури кожного шару згорткової нейронної мережі, будується модель оптимізації потоку медичних великих даних, вводиться функція ReLu для обчислення згорткової нейронної мережі, розв'язується оптимізаційна модель і завершується обробка потоку незбалансованих медичних великих даних. Результати експериментів показують, що точність прогнозування навантаження на мережеве з'єднання запропонованого стрімінгового алгоритму становить до 93%, найнижчий відсоток втрати пакетів лише 2.0%, енергетичні втрати в процесі потоку низькі, швидкість швидка, і ефективність анти-атаки висока, що сприяє реалізації поточкових даних.

Дослідження [12] обговорює питання аналізу медичних великих даних з використанням методів глибокого навчання.

З поширенням комп'ютерних технологій обсяг медичних даних значно зріс, і у відповідь на це з'явилися методи аналізу медичних великих даних,

орієнтовані на дані. Ці методи покликані забезпечувати підтримку для інтелектуальної ідентифікації медичних станів. Однак через різноманітність форматів медичних даних, безліч неповних записів і наявність шуму аналіз медичних великих даних залишається складним завданням.

Традиційні методи машинного навчання не завжди можуть ефективно витягувати інформацію з медичних великих даних, тоді як глибоке навчання будує ієрархічну модель, імітуючи людський мозок. Це забезпечує автоматичне вилучення ознак, складну конструкцію моделі та ефективне вираження ознак. Стаття пропонує модель аналізу даних на основі глибокого навчання для медичних зображень і транскрипцій з метою інтелектуальної ідентифікації та діагностики захворювань. Модель використовує масові медичні дані для вибору та оптимізації параметрів, автоматичного вивчення процесу патологічного аналізу, і, в кінцевому підсумку, інтелектуального проведення діагностики та прийняття рішень на основі аналізу результатів медичних великих даних.

Експериментальні результати показують, що метод здатний аналізувати медичні великі дані та проводити ранню діагностику захворювань, а також передбачати ризик певного захворювання в майбутньому на основі даних про фізичний стан пацієнта. Це може істотно знизити навантаження на лікарів і медичних дослідників, покращуючи їхню ефективність роботи.

Наукова робота [13] обговорює вплив швидкого розвитку технології великих даних і хмарних платформ на медичну індустрію.

Розвиток електронної медицини призводить до збільшення обсягу медичних даних, які володіють різноманітними характеристиками, такими як багатий режим, неповні дані, тимчасові особливості та надмірність. Однак наукове використання цих даних має важливе значення для результатів медичної діяльності. У статті проводиться огляд характеристик медичних даних, актуальних технологій майнінгу даних у медичній галузі,



досліджуються теорії, базові моделі, алгоритми, етапи процесу та ключові аспекти. Автори роблять загальний огляд поточних напрямків досліджень у галузі майнінгу медичних даних. Ця стаття слугує основою для подальших досліджень у цій галузі та сприяє поширенню знань про поточні дослідження в цьому напрямку.

Стаття [14] описує результати розробки штучних нейронних мереж (ШНМ) для прогнозування продуктивності медичних пристроїв на основі даних про відповідність. Дані про відповідність медичних пристроїв було отримано з періодичних перевірок, проведених акредитованою лабораторією за ISO 17020 у період з 2015 по 2019 рік.

Для розробки ІНС використовувалися 1738 зразків перевірки відповідності насосів для вливання та перфузорів. Із загальної кількості зразків 1391 (80%) використовували під час розроблення системи, і 346 (20%) зразків використовували для подальшої валідації продуктивності системи. У процесі розроблення системи тестували вплив різної кількості нейронів у прихованому шарі та функцій активації на загальну точність системи.

Також було випробувано дві архітектури нейронних мереж: прямого і зворотного поширення. Результати показують, що архітектура нейронної мережі прямого поширення з 10 нейронами в одному прихованому шарі має кращу продуктивність. Загальна точність цієї нейронної мережі становить 98,06% для прогнозування продуктивності перфузорів і 98,83% для прогнозування продуктивності насосів для вливання. Рекурентна нейронна мережа дала точність 98,41% для обох типів насосів. Результати демонструють, що дані про відповідність, отримані через щорічні перевірки медичних пристроїв, успішно можуть використовуватися для прогнозування продуктивності окремого медичного пристрою. Це має важливе значення для підвищення безпеки і точності діагностики та лікування пацієнтів.

Стаття [15] фокусується на дослідженні технології генерації тексту в галузі обробки природної мови та штучного інтелекту, особливо в медичній галузі. Технологія генерації тексту відіграє важливу роль у розробці інтелектуальних систем запитань і відповідей у медичній сфері.

Раніше в системах запитань і відповідей відображення згенерованих діалогів зазвичай обмежувалося запитаннями і відповідями і було складно адаптувати його для вертикальних областей з вираженими характеристиками. У цій статті для розв'язання завдання спеціалізованих запитань і відповідей у медичній галузі та надання користувачам повних відповідей пропонується підхід до генерації медичного тексту на основі архітектури TensorFlow.

У цьому підході використовуються послідовні моделі в Keras для перетворення задачі на задачу класифікації, розглядаючи проблему генерації тексту як задачу передбачення. Після навчання та експериментів з моделлю на великомасштабному китайському наборі даних запитань і відповідей у сфері медицини результати показують, що модель добре підходить для застосування в даній конкретній галузі. Модель може агрегувати знання в галузі медицини, витягувати корисну інформацію про лікування і генерувати тексти медичних знань.

Також було виявлено, що розбиття запитань на фрази та їхнє подальше введення в модель ефективно розв'язує проблему речень, що повторюються, у відповідях, що генеруються. Через те, що запропонована модель добре справляється із запитаннями пацієнтів і дає точні відповіді в більшості випадків, дане дослідження надає напрямок для поліпшення методів медичного інтелектуального запитання-відповіді та генерації тексту.

Дослідження [16] зачіпає питання прогнозування міграції медичного обслуговування, що стало одним із цікавих питань у галузі охорони здоров'я інформатики. Це зумовлено тим, що поведінка щодо міграції медичного обслуговування тісно пов'язана з оцінкою рівня медичного обслуговування в

регіоні, раціональним використанням медичних ресурсів і розподілом медичної страховки.

У статті представлено модель прогнозування міграції медичного обслуговування на основі даних медичної страховки. Спочатку будується граф медичного обслуговування на основі даних медичної страховки. Граф медичного обслуговування представляє собою гетерогенний граф, що містить сутності, такі як пацієнти, захворювання, лікарні, ліки, госпіталізації та зв'язки між ними. Однак наявні графові нейронні мережі не здатні вловлювати часові зв'язки між сутностями типу подій.

У зв'язку з цим у статті пропонується модель прогнозування на основі графових згорткових мереж (GCN), названа Event-involved GCN (EGCN). Запропонована модель агрегує звичайні сутності на основі механізму уваги та агрегує сутності типу подій на основі механізму вентиляції, аналогічного LSTM. Крім того, використовується механізм стрибкового з'єднання для отримання остаточного подання вузла. Для отримання вбудованих уявлень ліків на основі зовнішньої інформації (описів ліків) у запропонованій моделі розгортається автокодер, здатний вбудовувати описи ліків. Нарешті, проведено великі експерименти на реальному наборі даних медичної страховки. Результати експериментів показують, що прогностична здатність запропонованої моделі перевершує найкращі наявні моделі.

Дослідження [17] стосується дослідження багатомодальних медичних ознак, що представляють медичні дані, та аналізу різних методів навчання для оцінювання ризику захворювань з метою збору інформації з медичних даних і генерації інтелектуальних запитів, що відповідають застосункам. Представлення медичних даних здійснюється методом глибокого навчання. Для досягнення адаптивності моделі, дослідження і вилучення різних ознак хвороби використовують одну і ту ж процедуру.

Класифікація медичних даних є важливим завданням видобутку даних, яке привертає увагу дослідників по всьому світу протягом десятиліть. Для сортування здоров'я використовують різні алгоритми видобутку даних, такі як класифікація, кластеризація, регресія, м'яке обчислення та інші. Алгоритми відбору ознак відіграють ключову роль у кожному завданні машинного навчання. Вибір найкращого методу призводить до оптимальної підмножини властивостей, що підвищує точність і скорочує час навчання. Це також корисно для видалення непотрібних характеристик у разі високорозмірних наборів даних.

Стаття надає короткий огляд моделей класифікації медичних даних, щоб дати коротку оцінку загальних моделей добору ознак і класифікації, спеціально використовуваних для класифікації медичних даних. Різні аспекти, як-от доступні набори медичних даних, вибір ознак, вибір класифікації, проблеми в ідентифікації та аналіз ключових методів добору ознак і детальні механізми, представлені в цій оглядовій статті.

### 1.5 Висновки до першого розділу

У першому розділі проведено аналіз предметної області, спрямований на вивчення ключових аспектів, що визначають контекст дослідження. В результаті аналізу визначено, що штучні нейронні мережі є важливою складовою області, розширюючи можливості обробки інформації та вирішення завдань.

Окрім цього, вивчено основні концепції нейронних моделей, розкриваючи їх структуру та принципи функціонування. На основі здобутих знань була визначена актуальність використання нейронних моделей в медичній галузі. Зазначено, що сучасні технології глибокого навчання та нейронні мережі можуть ефективно використовуватися для аналізу та

обробки медичних даних, сприяючи автоматизації та підвищенню точності діагностики.

У підсумку, аналіз представлених в розділі концепцій нейронних моделей в медицині підтверджує їхню значимість та потенціал для вдосконалення медичної практики. Поручені аспекти дозволяють розглядати застосування штучних нейронних мереж у контексті вирішення проблем сучасної медицини та покращення якості надання медичних послуг.

## РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ НЕЙРОННИХ МОДЕЛЕЙ ДЛЯ АНАЛІЗУ МЕДИЧНИХ ДАНИХ

### 2.1 Збір даних та їх попереднє опрацювання

Збір і попереднє опрацювання даних є наріжними каменями в розробці систем машинного навчання, особливо в контексті аналізу медичних даних.

Ці етапи надають фундаментальні основи для подальшої побудови моделей, навчання і, в кінцевому підсумку, ухвалення поінформованих рішень у галузі охорони здоров'я. Процес збору даних починається з визначення цілей дослідження або завдання, яке необхідно вирішити. У разі медичних даних, це може містити різноманітні джерела інформації, такі як медичні записи пацієнтів, результати обстежень, зображення (наприклад, знімки КТ або МРТ), лабораторні дані, текстові звіти та багато іншого. Одним із ключових аспектів під час збору даних у медичній галузі є не тільки їхня кількість, а й якість. Повні й точні дані критично важливі для успішного навчання моделей. Важливо приділяти увагу виключенню помилок і артефактів у даних, оскільки неправильні або спотворені записи можуть внести плутанину і знизити точність моделі [14, 16].

Коли дані зібрано, наступним кроком є їхнє попереднє опрацювання. Цей етап охоплює низку підзадач, починаючи від очищення даних від викидів і аномалій до перетворення формату і структури даних для зручності подальшого аналізу. Одним із перших завдань попередньої обробки даних є усунення пропусків. У медичних даних пропуски можуть виникнути з різних причин, таких як помилки введення, технічні збої або навіть відсутність деяких вимірювань у пацієнта. Вирішення цієї проблеми може включати в себе заповнення пропусків, використовуючи статистичні методи або середні значення, або, в разі недостатнього обсягу даних, видалення відповідних

записів. Іншим важливим аспектом є нормалізація даних. Медичні дані можуть мати різні шкали та одиниці виміру, що може ускладнити порівняння та аналіз.

Нормалізація дозволяє привести всі дані до одного стандарту, що спрощує зіставлення і використання даних для навчання моделей. Очищення даних від викидів також є критичним етапом. Аномалії в даних можуть спотворити результати і впливати на навчання моделей.

Методи, такі як медична експертиза та статистичні техніки, можуть бути використані для виявлення та видалення викидів. Наступним етапом попередньої обробки даних є інтеграція. У медичних дослідженнях дані можуть надходити з різних джерел, і об'єднання їх у єдиний набір даних може бути складним завданням. Це може вимагати приведення даних до загального формату, створення унікальних ідентифікаторів для пацієнтів, об'єднання різнорідних даних тощо. Важливим кроком є також перетворення даних у зручний для навчання формат. Наприклад, зображення можуть бути масштабовані або перетворені у формат, придатний для входу в згорткові нейронні мережі. Текстова інформація може потребувати токенізації та векторизації для підготовки до використання в рекурентних мережах.

Крім того, важливо приділяти увагу балансуванню класів у даних, особливо під час навчання моделей для завдань класифікації. Нерівномірний розподіл класів може призвести до перекоосу і знизити ефективність навчання моделі. Після завершення етапу попереднього опрацювання, дані готові до використання в навчанні моделей машинного навчання.

Однак важливо розуміти, що попереднє опрацювання даних – це ітеративний процес, і під час роботи над моделлю може знадобитися внесення додаткових коригувань і поліпшень у дані. Загальний успіх проекту машинного навчання в медицині значною мірою залежить від ретельного і грамотного виконання збору та попереднього оброблення даних. Точність і

достовірність результатів, отриманих на етапі навчання моделі, безпосередньо залежать від якості підготовлених даних. Ефективне управління цими етапами не тільки забезпечує успішне навчання моделей, але також сприяє підвищенню рівня довіри лікарів і пацієнтів до застосування машинного навчання в медичній практиці.

## 2.2 Згорткові нейронні мережі

Згорткові нейронні мережі (CNN) є класом глибоких нейронних мереж, які широко використовуються в обробці зображень і аналізі даних, де просторова структура має ключове значення (рис. 2.1) [17].

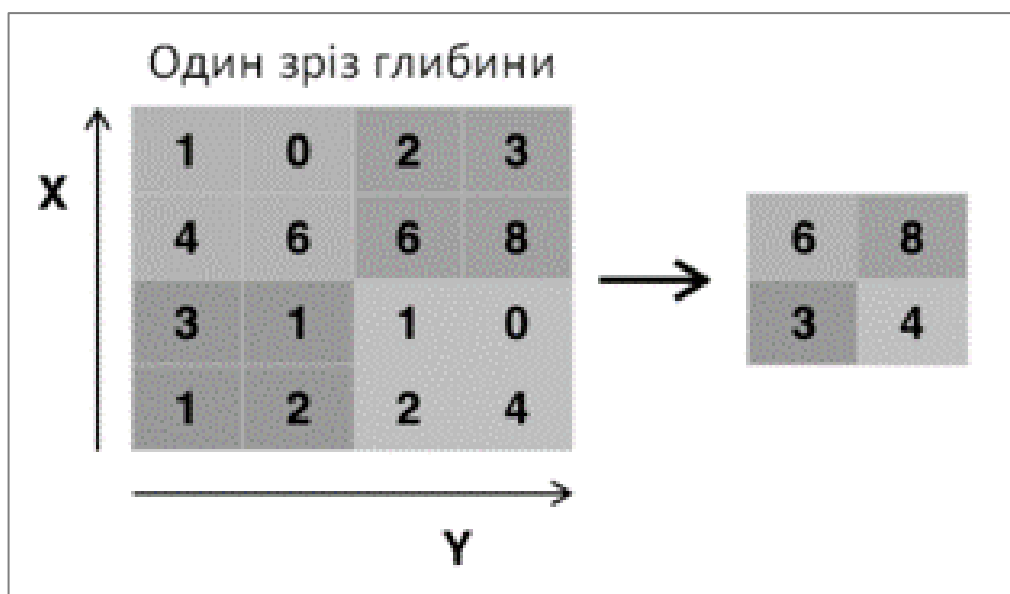


Рисунок 2.1 – Згорткові нейронні мережі

Ці мережі розроблені з урахуванням особливостей вхідних даних, таких як зображення, і дають змогу автоматично витягувати ієрархії ознак зі складних візуальних даних. Основою згорткових нейронних мереж є згорткові шари, які дозволяють моделі виявляти локальні ієрархії ознак у зображеннях. Згортка – це математична операція, за якої ядро (або фільтр)



просканує вхідне зображення, виділяючи особливості та створюючи карти ознак. Це дозволяє нейронній мережі «навчатися» розрізняти різні аспекти зображень, такі як краї, форми та текстури.

Однією з ключових переваг згорткових шарів є їхня здатність до інваріантності до масштабу і перенесення. Це означає, що нейронна мережа, навчена на одних даних, здатна розпізнавати ті ж самі об'єкти на зображеннях, навіть якщо вони розташовані в різних частинах зображення або мають різний розмір.

Крім згорткових шарів, у згорткових нейронних мережах часто використовують пулінг-шари, які виконують субдискретизацію, зменшуючи розмір карт ознак і зберігаючи найважливішу інформацію. Це дозволяє зменшити обчислювальне навантаження та підвищити інваріантність до невеликих трансформацій зображень. Архітектура згорткових нейронних мереж часто містить у собі кілька послідовних згорткових і пулінгових шарів, що створює ієрархію ознак. Також можуть використовуватися повнозв'язні шари в кінці мережі для виконання класифікації або інших завдань. Процес навчання згорткових нейронних мереж включає в себе передачу ваг (вагові коефіцієнти) згорткових і повнозв'язних шарів таким чином, щоб мінімізувати функцію втрат.

Цей процес здійснюється з використанням алгоритмів оптимізації, таких як стохастичний градієнтний спуск (SGD) або його модифікації, які налаштовують ваги, з огляду на помилку між прогнозами моделі та фактичними мітками класів. Згорткові нейронні мережі успішно застосовують у різних галузях, таких як комп'ютерний зір, медична діагностика, автоматичне розпізнавання мови та інші.

У медицині, наприклад, згорткові мережі використовують для автоматизованого аналізу медичних зображень, таких як знімки КТ або МРТ, що дозволяє підвищити точність діагностики та прискорити процес обробки

даних. Однак, незважаючи на численні переваги, згорткові нейронні мережі не позбавлені обмежень. Наприклад, потрібна велика кількість розмічених даних для ефективного навчання, і висока обчислювальна потужність може бути необхідна для навчання глибоких архітектур.

Крім того, інтерпретація рішень згорткових нейронних мереж може становити складність, що робить їхнє застосування в чутливих галузях, як-от медицина, підданим додатковим вимогам до зрозумілості моделей. Підсумовуючи, згорткові нейронні мережі представляють собою потужний інструмент в обробці зображень і аналізі даних з просторовою структурою. Їхня ефективність у медичній діагностиці та інших галузях підтверджує їхню значимість і перспективи для майбутнього розвитку в галузі штучного інтелекту та машинного навчання.

### 2.3 Рекурентні нейронні мережі

Рекурентні нейронні мережі (RNN) представляють собою клас глибоких нейронних мереж, розроблених для ефективного опрацювання послідовних даних і врахування контексту в часі. На відміну від згорткових нейронних мереж, які спеціалізуються на аналізі просторових структур, рекурентні мережі орієнтовані на опрацювання часових послідовностей, таких як тимчасові ряди, тексти або аудіозаписи. Основною ідеєю рекурентних нейронних мереж є використання зворотного зв'язку та спільного вагового простору для передавання інформації з попередніх кроків часу в поточний. Це дозволяє моделі враховувати залежності в послідовних даних і вловлювати довгострокові залежності між елементами послідовності. Однією з ключових характеристик RNN є наявність прихованого стану (hidden state), який представляє собою внутрішнє уявлення моделі про попередній контекст. Прихований стан оновлюється на кожному кроці часу

відповідно до поточного входу і попереднього прихованого стану, що дає змогу моделі зберігати інформацію про минулі події [11-13].

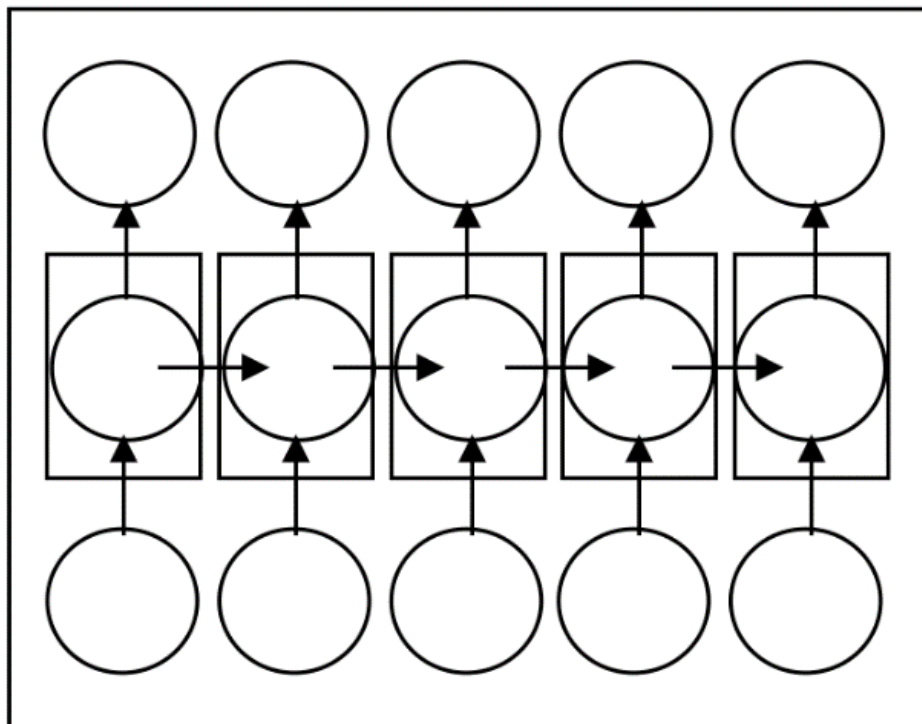


Рисунок 2.2 – Рекурентна нейронна мережа

Однак, у стандартних рекурентних мережах існує проблема затухаючих (vanishing) і вибухаючих (exploding) градієнтів. Це пов'язано з тим, що градієнти можуть зменшуватися або збільшуватися експоненціально зі збільшенням глибини мережі, що ускладнює навчання на довгострокових залежностях. Для подолання цієї проблеми було розроблено більш просунуті архітектури, такі як довга короткострокова пам'ять (LSTM) і gated recurrent unit (GRU). LSTM представляє собою модифікацію RNN, що містить у собі спеціальні осередки пам'яті, які можуть зберігати й оновлювати інформацію протягом тривалого часу.

Ця архітектура містить у собі три вентиля – вхідний, забування і вихідний – які регулюють потік інформації в комірку пам'яті, що дає змогу моделі довго зберігати і переносити інформацію в послідовних даних.

Альтернативою LSTM є GRU, який також володіє спеціальною структурою вентилів, але має менше параметрів і обходиться без окремої комірки пам'яті. GRU надає аналогічну здатність моделі до вловлювання довгострокових залежностей, при цьому вимагаючи менше обчислювальних ресурсів. Сфера застосування рекурентних нейронних мереж величезна і охоплює різноманітні завдання, такі як машинний переклад, розпізнавання мови, аналіз тексту, тимчасові ряди та багато іншого.

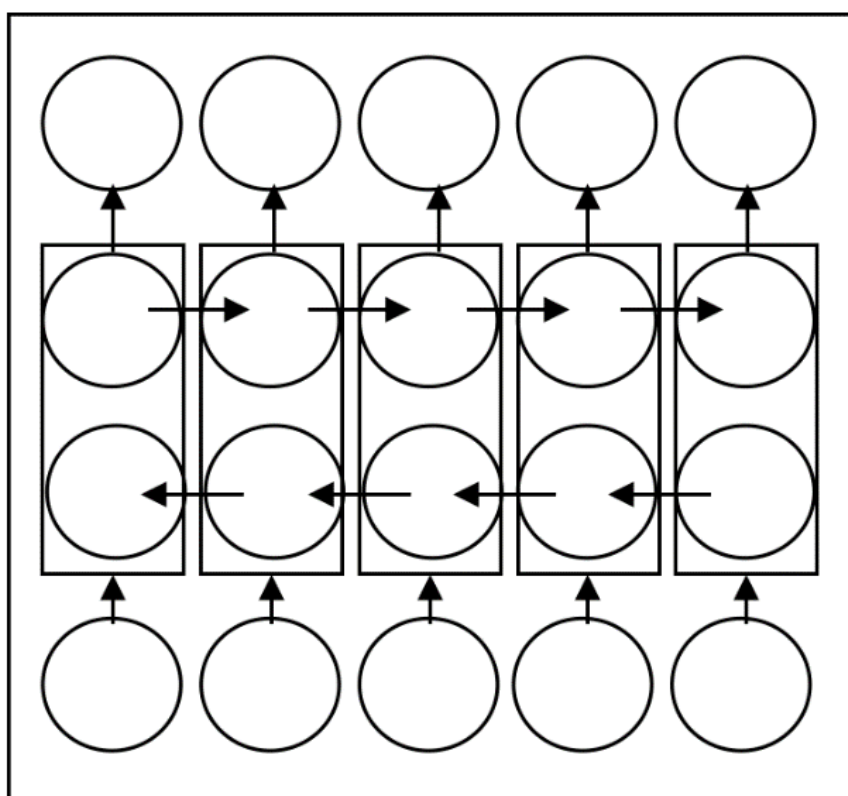


Рисунок 2.3 – Рекурентна нейронна мережа

У медицині, наприклад, RNN використовують для аналізу послідовних даних пацієнтів, передбачення перебігу захворювань і діагностики на основі медичних часових рядів.

Проте, рекурентні нейронні мережі також мають свої обмеження. Однією з них є обчислювальна складність при навчанні глибоких моделей,

що може потребувати значних обчислювальних ресурсів. Крім того, RNN не завжди добре справляються з обробкою довгих послідовностей, і в деяких випадках можуть бути менш ефективними, ніж більш сучасні архітектури, такі як трансформери. Але рекурентні нейронні мережі є потужним інструментом для обробки послідовних даних і врахування часових залежностей. Їхні різноманітні архітектури, включно з LSTM і GRU, дозволяють долати обмеження стандартних RNN і успішно застосовуватися в різних галузях, сприяючи розвитку досліджень у сфері штучного інтелекту та машинного навчання.

## 2.4 Короткострокова пам'ять

Короткострокова пам'ять (Short-Term Memory, LSTM) є особливим типом рекурентної нейронної мережі (RNN), розробленою для опрацювання й аналізу послідовних даних, зберігаючи інформацію про довгострокові залежності в даних. Спочатку запропонована Юргеном Шмідхубером і Себастьяном Хочеїтером 1997 року, архітектуру LSTM було створено для розв'язання проблеми загасання градієнтів, що виникає під час навчання стандартних рекурентних мереж (RNN) [12-14].

Основна відмінність LSTM від традиційних RNN полягає в наявності спеціалізованих комірок пам'яті, які забезпечують довгострокове збереження інформації. Осередок пам'яті складається з трьох вентилів: вхідного, забування і вихідного. Ці вентилялі контролюють потік інформації в комірку і забезпечують можливість як запам'ятовування, так і забування інформації протягом часу.

Вентиль забування (Forget Gate) дає змогу моделі визначати, яку частину інформації в комірці пам'яті слід забути. Це розв'язує проблему загасання градієнтів, оскільки модель може обирати, які значення залишити в

пам'яті на основі поточного контексту і завдання. Це важливо для ефективного навчання на довгострокових залежностях у даних.

Вентиль введення (Input Gate) дає змогу визначити, яку нову інформацію слід додати в комірку пам'яті. Він керує важливістю нових даних і вирішує, які значення слід оновити. Це забезпечує гнучкість моделі в адаптації до різних обставин і навчання на різних шаблонах даних.

Вентиль виведення (Output Gate) визначає, яку інформацію з комірки пам'яті слід використовувати для поточного виходу моделі. Цей вентиль регулює, яку інформацію з пам'яті комірки буде передано на вихід і використано для передбачень або класифікації.

Однією з ключових особливостей LSTM є здатність зберігати інформацію в пам'яті протягом багатьох часових кроків. Це досягається завдяки тому, що вентиля контролюють потік інформації, запобігаючи загасанню градієнтів і забезпечуючи ефективне навчання на довгострокових залежностях. Це робить LSTM ідеальним вибором для розв'язання завдань, де важлива здатність моделі враховувати контекст упродовж тривалого часу, наприклад, у задачах машинного перекладу, аналізі текстів і часових рядів. Також слід зазначити, що LSTM адаптували і розширили для вирішення складніших завдань. Наприклад, стеки LSTM (LSTM Stacks) містять у собі кілька шарів LSTM, що дозволяє моделі витягувати складніші ієрархії ознак. Завдяки цим розширенням, LSTM успішно застосовують у різних галузях, таких як обробка природної мови, аналіз часових рядів, медична діагностика та інші.

Навчання глибоких LSTM-мереж може бути обчислювально витратним, і вони вимагають великого обсягу розмічених даних для ефективного навчання. Крім того, LSTM можуть стикатися з проблемою перенавчання на невеликих наборах даних, що вимагає застосування методів регуляризації та інших технік для запобігання втрати узагальнювальної

здатності моделі. Насамкінець, короткострокова пам'ять (LSTM) є важливим класом нейронних мереж, спроектованим для ефективного опрацювання часових залежностей у послідовних даних. Її унікальна архітектура, включно з вентилями забування, введення і виведення, дає змогу моделі ефективно працювати з довгостроковими залежностями та успішно застосовуватися в різних галузях, сприяючи розвитку досліджень у сфері штучного інтелекту і машинного навчання.

## 2.5 Глибокі нейронні мережі прямого розповсюдження

Глибокі нейронні мережі прямого розповсюдження (feedforward neural networks, FNN) є потужним класом моделей у сфері машинного навчання та штучного інтелекту, які використовують для розв'язання різноманітних завдань, таких як класифікація, регресія, кластеризація та інші [15].

Ці мережі є окремим випадком нейронних мереж і мають здатність передавати інформацію тільки в одному напрямку: від вхідних вузлів, через приховані шари, до вихідних вузлів без зворотних зв'язків. У цьому контексті необхідно розглянути основні концепції, архітектуру та принципи навчання глибоких нейронних мереж прямого поширення [3].

Основна ідея глибоких нейронних мереж прямого поширення полягає в тому, щоб використовувати безліч шарів нейронів, організованих послідовно, для вираження складних нелінійних залежностей у даних. Ці моделі складаються з трьох основних типів шарів: вхідного шару, прихованих шарів і вихідного шару. Вхідний шар приймає на вхід дані, приховані шари виконують перетворення, а вихідний шар генерує фінальні передбачення або класифікації. Кожен вузол (або нейрон) у шарах, крім вхідного, пов'язаний із вузлами попереднього і наступного шарів. Кожен зв'язок між вузлами має вагу, яка регулює внесок вузла в активацію вузла наступного шару. Крім ваг,

кожен вузол може мати зміщення (bias), яке додається до зваженої суми входів перед застосуванням функції активації.

Функції активації відіграють ключову роль у глибоких нейронних мережах прямого розповсюдження, оскільки вони надають моделі нелінійність і дають їй змогу виражати складніші залежності. Різні функції активації можуть використовуватися залежно від завдання і вимог. Наприклад, сигмоїдальну функцію активації часто застосовують в останньому шарі для завдань бінарної класифікації, а функції, такі як ReLU (Rectified Linear Unit), застосовують у прихованих шарах для подолання проблеми загасання градієнтів.

Архітектура глибоких нейронних мереж прямого поширення визначається кількістю шарів і кількістю вузлів у кожному шарі. Мережі з великим числом шарів вважаються глибокими, і ця глибина може призвести до підвищення здатності моделі виявляти абстрактні ієрархії ознак у даних. Однак, зі збільшенням числа шарів може виникнути проблема загасання градієнтів, що ускладнює навчання глибоких моделей.

Різні техніки, такі як нормалізація пакета (Batch Normalization) та ініціалізація ваг, розроблені для подолання цих проблем. Процес навчання глибоких нейронних мереж прямого поширення містить у собі два основні етапи:

- 1) пряме поширення (forward pass);
- 2) зворотне поширення помилки (backpropagation).

Під час прямого поширення вхідні дані подають на вхідний шар, а потім активації по черзі передають через приховані шари до вихідного шару, де генерують передбачення моделі. Помилка між передбаченнями і фактичними значеннями вимірюється з використанням функції втрат. На етапі зворотного поширення помилки градієнт помилки за вагами обчислюється з використанням методу градієнтного спуску.



Градiєнт поширюється від вихідного шару до вхідного, оновлюючи ваги і зміщення таким чином, щоб зменшити помилку. Цей процес повторюється на безлічі ітерацій (epoch) доти, доки модель не досягне задовільного рівня продуктивності. Глибокі нейронні мережі прямого поширення широко застосовуються в різних галузях. У комп'ютерному зорі вони успішно застосовуються для розпізнавання об'єктів і обробки зображень. В обробці природної мови, глибокі мережі використовуються для машинного перекладу, аналізу тексту і створення чат-ботів.

У медичній діагностиці, вони застосовуються для аналізу зображень з зображень знімків і передбачення захворювань. Незважаючи на свою ефективність, глибокі нейронні мережі прямого поширення мають деякі обмеження. Одним з основних обмежень є необхідність великого обсягу розмічених даних для навчання успішних моделей, що може бути проблемою в задачах, де дані дорогі в зборі або розмітці. Крім того, навчання глибоких моделей може вимагати значних обчислювальних ресурсів, особливо в разі використання великої кількості шарів і параметрів. Насамкінець, глибокі нейронні мережі прямого розповсюдження є сучасним інструментом у галузі машинного навчання та штучного інтелекту. Їхня здатність моделювати складні залежності в даних зробила їх популярними в різних галузях, незважаючи на виклики в навчанні та вимоги до даних. З постійним розвитком технологій і методів, глибокі нейронні мережі прямого поширення продовжують залишатися ключовим напрямком у сфері досліджень і застосувань штучного інтелекту.

## 2.6 Глибокі згорткові нейронні мережі

Глибокі згорткові нейронні мережі (CNN) є класом моделей у галузі комп'ютерного зору та аналізу зображень [11].

Ці нейронні мережі особливо ефективні в розпізнаванні патернів у візуальних даних і знайшли широке застосування в різних галузях, включно з розпізнаванням об'єктів, сегментацією зображень, класифікацією і навіть генерацією зображень. Основою глибоких згорткових нейронних мереж є згорткові шари, які призначені для автоматичного вилучення ієрархії ознак із вхідних даних. Ці шари використовують операцію згортки для сканування вхідного зображення з використанням ядра (фільтра), виділяючи локальні патерни і структури.

Потім, результат застосування згортки утворює карти ознак, які передаються наступним шарам для подальшого аналізу. Ключовою перевагою згорткових шарів є їхня здатність до автоматичного виявлення локальних залежностей у зображеннях, що робить їх ефективнішими для аналізу візуальних даних, ніж традиційні повнозв'язні шари. Згорткові операції дають змогу мережі навчатися розпізнавати різні рівні деталізації, починаючи від країв і текстур, і закінчуючи складнішими об'єктами і структурами.

Однак глибокі згорткові нейронні мережі часто включають не тільки згорткові, а й пулінг-шари. Пулінг (pooling) представляє собою операцію субдискретизації, яка зменшує розмір карти ознак, зберігаючи найважливішу інформацію. Це допомагає зменшити кількість параметрів моделі та покращити інваріантність до масштабу і невеликих трансформацій об'єктів на зображенні. Архітектура глибоких згорткових нейронних мереж зазвичай містить у собі кілька послідовних згорткових і пулінгових шарів, що створює ієрархію ознак. Після цих шарів зазвичай додаються повнозв'язні шари, які виконують класифікацію або інші завдання, засновані на виділених ознаках. Процес навчання глибоких згорткових нейронних мереж передбачає передачу ваг згорткових і повнозв'язних шарів таким чином, щоб мінімізувати функцію втрат. Цей процес реалізується з використанням

алгоритмів оптимізації, таких як стохастичний градієнтний спуск (SGD) або його модифікації.

Навчання таких мереж часто вимагає великого обсягу розмічених даних і обчислювальних ресурсів. Одним із важливих застосувань глибоких згорткових нейронних мереж є комп'ютерний зір. Вони успішно застосовуються в задачах розпізнавання об'єктів, де моделі можуть автоматично виділяти і класифікувати об'єкти на зображенні. Це знаходить застосування в автомобільній промисловості (автономні автомобілі), медичній діагностиці (аналіз медичних зображень), обробці відео та інших галузях. Важливим кроком у розвитку згорткових мереж стало використання попередньо навчених моделей на великих наборах даних, таких як ImageNet. Це дозволяє моделям витягувати універсальні ознаки зображень і успішно застосовувати їх до різних завдань без необхідності навчання моделі з нуля. Однак, незважаючи на видатні досягнення, глибокі згорткові нейронні мережі також мають свої обмеження.

Наприклад, вони можуть бути чутливі до вхідних даних і схильні до атак, таких як додавання шуму або зміна зображення. Також їхнє застосування в завданнях, що вимагають зрозумілості (наприклад, у медичній сфері), може бути ускладнене, оскільки процес ухвалення рішень усередині мережі може бути складно інтерпретований.

Отже, глибокі згорткові нейронні мережі є класом моделей, спроектованих для ефективного аналізу та обробки візуальних даних. Їхнє успішне застосування в комп'ютерному зорі та інших галузях стало важливим внеском у розвиток штучного інтелекту. Незважаючи на свої обмеження, вони залишаються ключовим напрямком досліджень і розробок у галузі машинного навчання.

## 2.7 Проблема оверфітінгу

Проблема оверфітінгу є одним з основних викликів у галузі машинного навчання та статистики. Вона виникає в процесі навчання моделі, коли вона надто точно підлаштовується під тренувальні дані, захоплюючи шум і випадкові коливання в них, що зрештою призводить до погіршення її узагальнювальної здатності на нових даних, які раніше не зустрічалися. Оверфітінг є формою перенавчання, за якої модель стає надто складною і налаштовується на шум у даних, замість того щоб виявляти загальні закономірності та структури [12].

Це може призвести до того, що модель буде погано узагальнюватися на нові дані, оскільки вона запам'ятовує, а не виявляє загальні патерни. Основні причини оверфітінгу включають незбалансований обсяг даних, наявність великої кількості ознак і недостатнє використання методів регуляризації. Додатково, недостатня кількість даних для навчання також може бути фактором, що створює сприятливі умови для появи оверфітінгу. Одним із ключових аспектів розуміння проблеми оверфітінгу є баланс між зміщенням (bias) і розкидом (variance).

Зміщення відображає помилку, пов'язану зі спрощенням моделі та ігноруванням складних патернів у даних, тоді як розкид стосується чутливості моделі до змін у даних. Ідеальний баланс між зміщенням і розкидом призводить до оптимальної узагальнювальної здатності моделі, мінімізуючи помилки як на тренувальних, так і на тестових даних. Однак, при прагненні до поліпшення точності на тренувальних даних, модель може почати «підлаштовуватися» під них надто ретельно, що призводить до оверфітінгу. Це зумовлено тим, що модель починає зачувати шум і випадкові варіації в даних, замість того щоб витягувати загальні закономірності. Один із методів боротьби з оверфітінгом – це використання

додаткових даних для тренування. Збільшення обсягу даних часто допомагає моделі краще узагальнюватися, запобігаючи перенавчанню. Іншим методом є правильний вибір моделі та її складності.

Використання надто складних моделей з великим числом параметрів може призвести до оверфіттингу, тому важливо ретельно підбирати складність моделі під обсяг і характер даних. Іншим підходом до управління оверфіттингом є використання методів регуляризації. Регуляризація внесена в модель для контролю її складності та запобігання перенавчанню. Вона додає штраф за складність моделі до функції втрат, що допомагає запобігти надмірному налаштуванню на тренувальні дані. Техніки регуляризації включають в себе L1- і L2-регуляризацію, які додають штрафи на абсолютні значення ваг або їхні квадрати відповідно. Ці методи допомагають керувати величиною ваг, запобігаючи їхньому надмірному збільшенню.

Ще однією поширеною технікою є відсів (dropout), за якого випадково вибрані вузли в мережі відключаються в процесі навчання. Це допомагає зменшити взаємодію між вузлами та запобігає заучуванню даних. Важливим моментом є правильний поділ даних на тренувальні, валідаційні та тестові підмножини. Валідаційну підмножину використовують для оцінювання продуктивності моделі в процесі навчання, і налаштування гіперпараметрів здійснюють на основі її результатів. Тестова підмножина не бере участі в процесі навчання і використовується для остаточного оцінювання узагальнювальної здатності моделі.

Оверфіттинг також може бути керований шляхом використання ранньої зупинки (early stopping). Цей метод передбачає припинення навчання, коли продуктивність моделі на валідаційному наборі даних починає погіршуватися після деякої кількості епох. Це запобігає перенавчанню і зберігає модель на точці, де її узагальнювальна здатність максимальна. У контексті глибокого навчання, де моделі можуть мати мільйони параметрів,

також широко використовуються згорткові та пулінг-шари, які допомагають витягувати важливі просторові та часові залежності в даних. Ці шари також сприяють зниженню кількості параметрів і запобігають надлишковому навчанню. Необхідно також зазначити, що проблема оверфітінга актуальна не тільки для задач класифікації, а й для задач регресії та інших видів машинного навчання. Вона впливає на широкий спектр моделей, від лінійних регресій до глибоких нейронних мереж. Насамкінець, проблема оверфітінга представляє собою серйозний виклик у сфері машинного навчання, що вимагає уважного контролю та управління. Правильний вибір моделі, використання регуляризації, збільшення обсягу даних та інші техніки допомагають запобігти надлишковому навчанню і забезпечити створення моделей, що мають високу узагальнюючу здатність на нових даних.

## 2.8 Методи регуляризації

Методи регуляризації є важливим інструментом у галузі машинного навчання, призначеним для контролю перенавчання моделей. Вони вводять додаткові обмеження на параметри моделей, щоб запобігти надлишковій складності, зробити їх стійкішими до шуму в даних і поліпшити їхню узагальнювальну здатність на нових даних, які раніше не зустрічалися.

Один із найбільш широко використовуваних методів регуляризації – це L1-регуляризація, також відома як регуляризація Лассо. Цей метод додає до функції втрат моделі штраф, пропорційний абсолютним значенням її параметрів. Математично, L1-регуляризація представлена додаванням суми абсолютних значень ваг до функції втрат [10].

Цей штраф сприяє зменшенню деяких ваг до нуля, що, своєю чергою, призводить до розрідженості моделі. Такий ефект дає змогу відбирати найважливіші ознаки та зменшувати розмірність простору параметрів моделі.

Іншим поширеним методом регуляризації є L2-регуляризація, або регуляризація Гребнера. У цьому випадку, до функції втрат додається штраф, пропорційний квадратам ваг моделі. У математичному поданні це виражається додаванням суми квадратів ваг до функції втрат. L2-регуляризація не призводить до розрідженості ваг, але ефективно запобігає надлишковій складності моделі, пригнічуючи вплив великих ваг на загальну помилку. Крім того, існує комбінований підхід, званий Elastic Net, який включає в себе і L1, і L2-регуляризацію.

Цей метод вводить два параметри, які дають змогу балансувати вплив обох регуляризацій на модель. Elastic Net має переваги обох регуляризацій, зберігаючи властивість розрідженості та контролюючи загальну складність моделі. Важливим аспектом методів регуляризації є їхнє застосування не тільки в лінійних моделях, а й у складніших структурах, таких як гребенева регресія (ridge regression) і ласо-регресія (lasso regression) для лінійної регресії, або в поєднанні з глибокими нейронними мережами в глибокому навчанні. У глибокому навчанні, наприклад, застосування регуляризації стає критично важливим у зв'язку з високою складністю моделей і великим числом параметрів. Однією з поширених технік регуляризації в глибокому навчанні є Dropout.

Dropout представляє собою випадкове відключення (обнулення) деяких вузлів у мережі під час навчання. Це допомагає запобігти перенавчанню, оскільки мережа не може покладатися на певні вузли під час кожної ітерації, що змушує її навчати більш стійкі та узагальнювальні ознаки. У контексті глибокого навчання також використовуються L1- і L2-регуляризації для ваг у нейронних мережах. Це допомагає контролювати розмір ваг і їхній вплив на загальну функцію втрат. Під час використання регуляризації в глибоких нейронних мережах важливо ретельно налаштовувати параметри регуляризації, щоб досягти оптимального балансу між запобіганням

перенавчання і збереженням високої продуктивності моделі. Однією з актуальних галузей досліджень у методах регуляризації є розробка адаптивних методів, які здатні автоматично регулювати параметри регуляризації залежно від характеристик даних і структури моделі. Це може підвищити зручність використання цих методів у практичних задачах машинного навчання. Необхідно також зазначити, що методи регуляризації є важливим компонентом навчання моделей на невеликих обсягах даних.

У таких випадках, коли обсяг даних обмежений, існує ризик перенавчання, і застосування регуляризації стає критично важливим для створення стабільних і узагальнюючих моделей. На закінчення, методи регуляризації представляють собою потужний інструмент у боротьбі з проблемою перенавчання в машинному навчанні. Вони забезпечують баланс між складністю моделі та її здатністю до узагальнення, що робить їх важливими компонентами при побудові стабільних і ефективних моделей.

## 2.9 Відбір ознак

Відбір ознак – це важливий етап у процесі розроблення моделей машинного навчання, спрямований на виявлення найінформативніших і найрелевантніших ознак для передбачення цільової змінної.

Він покликаний зменшити розмірність даних, підвищити продуктивність моделі, уникнути перенавчання, а також поліпшити розуміння процесу прийняття рішень [9].

Ключовий момент у доборі ознак – це розуміння того, як кожна ознака впливає на цільову змінну і які з них найважливіші для конкретного завдання. Існує кілька підходів до відбору ознак, і вони можуть бути як статистичними, так і евристичними методами. Одним із підходів є фільтрація ознак.



Цей метод ґрунтується на статистичних метриках, таких як кореляція або взаємна інформація, для оцінки важливості ознак. Ознаки ранжуються за заданою метрикою, і потім можна вибирати топ-N ознак для використання в моделі. Однак, варто зазначити, що цей метод не завжди враховує взаємодію між ознаками і може упустити складні залежності. Інший метод – вбудований відбір ознак. Цей підхід включає в себе методи навчання моделі з автоматичним вибором найважливіших ознак у процесі навчання. Наприклад, дерева рішень можуть самі вирішувати, які ознаки найважливіші для поділу даних на класи.

Також існують регресійні моделі, здатні автоматично визначати важливість ознак для передбачення цільової змінної. Однак, навіть із використанням вбудованих методів, іноді виникає необхідність у додатковому доборі ознак, особливо за наявності великої кількості ознак або за наявності корельованих змінних.

Методи обгортання – ще один клас методів відбору ознак. Вони засновані на евристичних алгоритмах, які оцінюють продуктивність моделі з різними підмножинами ознак. Прикладом є рекурсивне виключення ознак (RFE), за якого модель навчається ітеративно зі зменшенням числа ознак до досягнення оптимального набору. Однак ці методи можуть бути обчислювально витратними і не завжди застосовні для великих даних. Наступний метод – вкладений відбір ознак, який інтегрує етап добору в процес навчання моделі.

Наприклад, в алгоритмах машинного навчання, таких як метод опорних векторів (SVM), є вбудована можливість вибору ознак залежно від їхньої важливості для визначення роздільної гіперплощини. Важливою концепцією у відборі ознак є «мішок слів» (bag of words) для текстових даних. Цей метод передбачає, що сенс речення залежить від зустрічальності слів у ньому,

незалежно від їхнього порядку. При цьому вибір певних слів може суттєво впливати на якість аналізу тексту.

У рамках мішка слів може бути застосовано відбір ознак, щоб вибрати найбільш релевантні слова для завдання аналізу тексту. Просунуті методи добору ознак включають в себе використання алгоритмів машинного навчання, таких як градієнтний бустинг, для оцінки важливості ознак. Ці методи враховують взаємодію ознак і здатні виявляти нелінійні залежності в даних. Однак, незважаючи на всі переваги відбору ознак, варто відзначити і деякі його недоліки. Зокрема, за неправильного вибору методу добору або недостатньої експертизи, можна втратити важливі ознаки, що впливають на цільову змінну. Крім того, в умовах мінливості даних, методи відбору ознак можуть втратити актуальність, і необхідно періодично переоцінювати їхню ефективність.

В цілому відбір ознак представляє собою важливий етап у побудові моделей машинного навчання. Існує безліч методів, що варіюються від статистичних підходів до використання просунутих алгоритмів машинного навчання. Ключовими аспектами є розуміння завдання, вибір відповідного методу відбору, і баланс між зменшенням розмірності даних і збереженням важливих для моделі ознак. Відбір ознак сприяє створенню більш ефективних, інтерпретованих і узагальнюючих моделей.

## 2.10 Передача навчання

Передача навчання (transfer learning) представляє собою методологію в машинному навчанні, яка ґрунтується на використанні знань, отриманих під час розв'язання однієї задачі, для поліпшення продуктивності в іншій задачі. Ця концепція натхненна ідеєю того, що моделі, навчені на одному завданні,

можуть витягти загальні ознаки і структури, які можна застосувати до інших завдань [16].

Передача навчання дозволяє використовувати досвід, отриманий на одному наборі даних, для поліпшення узагальнювальної здатності моделі на нових даних. Однією з основних причин використання передачі навчання є те, що в багатьох сценаріях навчання моделі з нуля з невеликим обсягом даних може бути непрактичним або неможливим. Замість цього, модель може бути попередньо навчена на великому наборі даних, після чого її знання можуть бути використані для розв'язання конкретного завдання з меншим обсягом даних. Це особливо корисно в галузях, де збір і розмітка великих обсягів даних трудомісткі. Одним із типів передачі навчання є перенесення навчання на основі ознак (feature-based transfer learning).

У цьому разі модель попередньо навчається на завданні класифікації з використанням великого обсягу даних, а потім витягнуті ознаки або ваги моделі переносяться на нову модель, яка навчається на завданні з меншим обсягом даних. Це дозволяє використовувати загальні ознаки, виявлені на первісному завданні, для ефективнішого розв'язання нового завдання.

Інший тип передачі навчання – це перенесення навчання на основі екземплярів (instance-based transfer learning). У цьому випадку використовуються конкретні приклади з навчального набору для нового завдання, разом з їхніми мітками класів. Ці екземпляри використовуються для навчання моделі на новому завданні, що допомагає моделі навчатися на основі подібних випадків, замість абстрактних ознак.

Процес передачі навчання часто включає в себе кілька ключових етапів.

Перший етап – це вибір попередньо навченої моделі або моделі, яку можна адаптувати під нове завдання. Це може бути модель, попередньо навчена на великому наборі даних, наприклад, у галузі комп'ютерного зору,

де моделі, навчені на зображеннях ImageNet, часто використовують як основу для інших завдань.

Після вибору попередньо навченої моделі наступним етапом є адаптація цієї моделі під нове завдання. Це може включати в себе зміну архітектури моделі, навчання додаткових шарів, або застосування додаткових методів, таких як навчання з учителем або без вчителя для поліпшення продуктивності на конкретному завданні.

Третій етап – це оцінка і тонке налаштування моделі на новому завданні. На цьому етапі модель навчається на новому наборі даних, і її продуктивність оцінюється з використанням метрик, специфічних для конкретного завдання. За результатами оцінки можна провести додаткове налаштування параметрів моделі для досягнення найкращих результатів. Передача навчання успішно застосовується в багатьох галузях машинного навчання. У галузі комп'ютерного зору, наприклад, моделі, попередньо навчені на великих наборах даних, можуть бути адаптовані для вирішення завдання розпізнавання об'єктів на конкретних зображеннях. В обробці природної мови моделі, навчені на великих корпусах тексту, можуть бути використані для завдання класифікації тексту або пошуку схожих текстів.

Однак передача навчання також стикається з низкою викликів. Один із них – це проблема доменної різниці (domain gap), коли структура даних нової задачі суттєво відрізняється від даних, на яких була навчена попередньо навчена модель. Це може призвести до втрати узагальнювальної здатності моделі на нових даних. Крім того, вибір оптимальної моделі для передачі навчання може бути нетривіальним завданням, яке потребує досвіду та експертних знань. Некоректний вибір моделі може призвести до недонавчання або перенавчання.

Проте, передача навчання продовжує привертати увагу дослідників і практиків у машинному навчанні, і нові методи та підходи до її поліпшення

з'являються в літературі. Ця концепція залишається одним із важливих інструментів для ефективного використання знань і досвіду, отриманих в одній царині, для поліпшення продуктивності моделей в інших концепціях.

2.11 Огляд програмних засобів для машинного навчання нейронних моделей

### 2.11.1 Бібліотека Scikit-learn

Scikit-learn (або sklearn) – це бібліотека для машинного навчання мовою програмування Python, що надає ефективні інструменти для аналізу даних і побудови моделей [14].

Вона створена на базі інших популярних бібліотек, таких як NumPy, SciPy і Matplotlib, і представляє собою один із найпопулярніших засобів для розв'язання задач класифікації, регресії, кластеризації тощо.

Основною метою scikit-learn є надання простого та однакового інтерфейсу для реалізації різних методів машинного навчання. Вона містить безліч функцій для попереднього опрацювання даних, вибору моделей, підбору гіперпараметрів і оцінки продуктивності моделей. Дизайн бібліотеки орієнтований на зручність використання і полегшення робочого процесу дослідника даних і розробника.

Одним із ключових понять у scikit-learn є концепція «Estimator». Estimator – це об'єкт, який може бути навчений на даних і використаний для робити передбачення на нових даних. Кожен Estimator у scikit-learn реалізує методи `fit` для навчання моделі та `predict` для передбачення значень на нових даних. Прикладами Estimators є моделі класифікації, регресії, кластеризації та багато інших. Для того щоб ефективно працювати з даними, scikit-learn надає безліч інструментів для попередньої обробки. Процес попереднього

опрацювання містить кроки, такі як масштабування даних, заповнення пропущених значень, кодування категоріальних ознак і багато інших. Бібліотека надає зручні функції та класи для виконання цих кроків, що дає змогу зосередитися на аналізі та побудові моделей, мінімізуючи рутинні завдання попередньої обробки. Scikit-learn також пропонує інструменти для роботи з текстовими даними, зокрема методи для вилучення ознак із тексту, векторизації тексту та тематичного моделювання. Це робить бібліотеку корисною для розв'язання завдань аналізу тексту, таких як класифікація документів або кластеризація текстових колекцій. Важливим компонентом scikit-learn є підтримка методів вибору ознак. Методи відбору ознак можуть бути корисними для зменшення розмірності даних, поліпшення узагальнювальної здатності моделі та прискорення навчання. Scikit-learn надає класи для реалізації як універсальних методів, так і специфічних методів відбору ознак для конкретних видів даних. Бібліотека також містить інструменти для оцінювання продуктивності моделей.

Метрики, такі як точність, F1-міра, коефіцієнт кореляції Метьюса і багато інших, доступні для різних типів завдань. Крім того, scikit-learn надає функції для крос-валідації, оптимізації гіперпараметрів і побудови кривих навчання, що допомагає аналізувати і покращувати моделі. Scikit-learn має широку підтримку алгоритмів машинного навчання. Ці алгоритми включають в себе методи лінійної регресії, метод опорних векторів, дерева рішень, випадкові ліси, метод k-найближчих сусідів, метод головних компонент і багато інших.

Кожен із цих алгоритмів має свої сильні та слабкі сторони, що дозволяє обирати метод, який найбільше підходить, залежно від конкретного завдання і характеристик даних. Крім класичних алгоритмів, scikit-learn також містить модуль для використання методів глибокого навчання через бібліотеку

TensorFlow. Це дає змогу використовувати переваги глибокого навчання в рамках знайомого інтерфейсу scikit-learn.

Одним із ключових принципів scikit-learn є підтримка навчання з учителем і без учителя. Навчання з учителем охоплює завдання класифікації та регресії, де модель навчається на маркованих даних з метою передбачення цільової змінної. Навчання без вчителя охоплює завдання кластеризації, зменшення розмірності та оцінювання щільності, де модель навчається на даних без явно зазначених міток. Scikit-learn активно підтримується спільнотою і постійно оновлюється. Нові методи, алгоритми та інструменти регулярно додаються для відображення останніх тенденцій у галузі машинного навчання та даних.

Крім того, документація бібліотеки надає великі приклади використання, що полегшує розуміння принципів роботи і застосування методів scikit-learn у практичних завданнях. Загалом бібліотека scikit-learn є сучасним і універсальним інструментом для машинного навчання мовою програмування Python. Його простота використання, широкі можливості для попередньої обробки даних, підтримка різноманітних алгоритмів і широкі засоби оцінювання продуктивності роблять його одним із найпопулярніших виборів для дослідників даних і розробників у галузі машинного навчання.

### 2.11.2 Бібліотека SciPy

SciPy – це бібліотека для наукових обчислень мовою програмування Python. Ця бібліотека надає безліч функцій для розв'язання різноманітних завдань у галузі науки, інженерії та статистики [11].

SciPy базується на бібліотеці NumPy, надаючи додаткові можливості для взаємодії з даними, виконання статистичних аналізів, чисельного інтегрування, оптимізації та інших наукових обчислень. Одним із ключових

компонентів SciPy є модуль `scipy.optimize`, який надає функції для розв'язання задач оптимізації. Оптимізація включає в себе пошук мінімуму або максимуму функції, що є частим завданням у наукових дослідженнях та інженерних додатках. Модуль `scipy.optimize` включає різні методи оптимізації, такі як методи на основі градієнта, методи без градієнта, глобальні методи оптимізації та методи оптимізації з обмеженнями.

Ще одним важливим компонентом SciPy є `scipy.integrate`, який надає функції для чисельного інтегрування. Чисельне інтегрування широко використовується для обчислення наближених значень інтегралів, які можуть бути важкими або неможливими для обчислення аналітично. Модуль включає методи для одновимірного та багатовимірного чисельного інтегрування, а також можливості роботи зі звичайними диференціальними рівняннями. Іншою важливою частиною SciPy є `scipy.stats`, який надає статистичні функції для аналізу даних. Цей модуль включає в себе широкий спектр статистичних методів, таких як тести гіпотез, оцінка параметрів розподілів, генерація випадкових чисел та інші. З використанням `scipy.stats` дослідники можуть проводити статистичний аналіз даних, перевіряти гіпотези та робити висновки про розподіл даних.

Модуль `scipy.linalg` надає функції лінійної алгебри, включно з операціями з матрицями, розв'язанням лінійних систем рівнянь, обчисленням власних значень і векторів, а також розкладанням матриць. Це важливий інструмент для розв'язання різних завдань, пов'язаних із лінійною алгеброю, яка широко використовується в різних галузях, зокрема у фізиці, інженерії та машинному навчанні. Для роботи з сигналами і зображеннями в SciPy передбачено модуль `scipy.signal`. Він містить методи для фільтрації сигналів, виділення ознак, кореляції та інших операцій, пов'язаних з обробкою сигналів. Це корисний інструмент у царині сигнального оброблення та аналізу зображень.



Модуль `scipy.spatial` надає функції для роботи з просторовими даними та алгоритмами, такими як пошук найближчих сусідів, побудова дерев просторового пошуку і відстані між точками. Це важливий інструмент для завдань, пов'язаних з аналізом просторових даних, таких як геоінформаційні системи, комп'ютерний зір і обробка зображень.

Модуль `scipy.special` надає функції для роботи з математичними спеціальними функціями, такими як функції Бесселя, функції Ейрі, гамма-функції тощо. Ці функції часто зустрічаються у фізиці, інженерії та математиці, і їхня наявність у SciPy полегшує розв'язання різноманітних завдань. SciPy також включає модуль `scipy.interpolate`, що надає функції для інтерполяції даних. Інтерполяція використовується для наближення значень функції між відомими точками, що може бути корисно в задачах аналізу даних, візуалізації та побудови моделей. Ще однією важливою частиною SciPy є `scipy.cluster`, що надає методи для кластеризації даних. Кластеризація дозволяє групувати дані на основі їхньої схожості, що може бути корисним для виявлення патернів у даних і створення структурованих кластерів.

Нарешті, SciPy також включає модуль `scipy.io` для введення/виведення даних. Цей модуль надає функції для роботи з різними форматами файлів, такими як MATLAB, NetCDF, WAV та інші. Це полегшує взаємодію з даними, представленими в різних форматах. Особливо варто відзначити, що SciPy тісно інтегрована з NumPy, що забезпечує однаковість у роботі з масивами даних. NumPy надає основу для багатьох операцій, що виконуються в SciPy, і забезпечує ефективну роботу з багатовимірними масивами. Важливо підкреслити, що SciPy є відкритим програмним забезпеченням з відкритим вихідним кодом, що дозволяє вільно використовувати, розповсюджувати та змінювати бібліотеку відповідно до вимог своїх проєктів.

Отже, SciPy представляє собою потужну і велику бібліотеку для наукових обчислень у мові програмування Python. Її функціональність охоплює різні галузі математики, статистики, фізики, аналізу даних і багато інших. Широкий спектр інструментів і методів, що надаються SciPy, робить її невід'ємною частиною інструментарію для дослідників та інженерів, які працюють у галузі наукових обчислень.

### 2.11.3 Бібліотека Joblib

Joblib представляє собою бібліотеку для паралельного виконання завдань у Python. Ця бібліотека була створена з метою поліпшення ефективності виконання обчислень, особливо в контексті наукових обчислень, де обробка великих обсягів даних або виконання складних обчислень може бути трудомістким процесом.

Joblib надає зручні засоби для розпаралелювання обчислень, кешування результатів та ефективного використання ресурсів комп'ютера. Основним компонентом Joblib є підмодуль `joblib.parallel`, який надає можливості для паралельного виконання коду. Цей модуль дозволяє легко розпаралелювати виконання циклів, функцій та інших операцій, що робить його корисним інструментом для прискорення виконання обчислень у багатозадачних сценаріях. Один із ключових елементів Joblib – це функція `Parallel`, яка дає змогу паралельно виконувати ітерації циклів у Python [8].

Шляхом зазначення параметрів, таких як кількість ядер процесора та інші параметри, користувач може налаштувати виконання коду в паралельному режимі. Це особливо корисно під час обробки даних, де цикли можуть бути досить тривалими. Ще однією важливою частиною бібліотеки є модуль `Memory`, який надає засоби для кешування результатів виконання функцій. Це може бути корисним у випадках, коли виконання функції

потребує багато часу, але результати залишаються незмінними між викликами. Кешування дозволяє уникнути повторного обчислення результатів і заощадити час. Joblib також підтримує розподілені обчислення через `joblib.parallel.Distributed` та інтеграцію з Apache Spark для масштабованих завдань. Це робить бібліотеку потужним інструментом для обробки даних у розподілених і паралельних середовищах. Ще одним важливим компонентом Joblib є модуль `joblib.Memory`, який надає інструменти для кешування результатів функцій у файловій системі.

Це особливо корисно в ситуаціях, коли обчислення функції вимагає багато часу, і результати залишаються постійними між запусками програми. Кешування в пам'яті допомагає заощадити час і ресурси, уникаючи повторного обчислення вже відомих результатів. Joblib також надає засоби для контролю над паралельним виконанням через параметри, такі як `n_jobs` і `backend`.

Параметр `n_jobs` дає змогу вказати кількість паралельних завдань, які можуть виконуватися одночасно.

Параметр `backend` визначає використовувану бібліотеку для реалізації паралельного виконання, таку як `threading` або `multiprocessing`. Ще одним важливим аспектом Joblib є його інтеграція з іншими бібліотеками Python, такими як NumPy і SciPy. Joblib може ефективно розпаралелювати виконання операцій, що надаються цими бібліотеками, що прискорює обробку даних і виконання обчислень.

Одним із цікавих застосувань Joblib є його використання в машинному навчанні. Наприклад, під час навчання моделей на великих об'ємах даних, де кожна ітерація потребує опрацювання великих обсягів інформації, розпаралелювання з використанням Joblib може значно прискорити процес навчання.

Joblib також надає можливості для створення користувацьких завдань, які можуть бути ефективно розпаралелені. Це дає користувачеві гнучкість у визначенні та управлінні завданнями, які вимагають паралельного виконання. Важливо зазначити, що Joblib активно підтримується та оновлюється. Нові версії бібліотеки включають поліпшення продуктивності, виправлення помилок і нові можливості. Це підкреслює активність і значущість бібліотеки в галузі наукових обчислень і обробки даних у Python.

Отже, Joblib є потужним інструментом для паралельного виконання завдань у Python. Його функціональність охоплює безліч аспектів паралельних обчислень, від розпаралелювання циклів до кешування результатів та інтеграції з іншими бібліотеками. Joblib забезпечує ефективні засоби для прискорення виконання складних обчислень і підвищення продуктивності в галузі наукових та інженерних обчислень.

## 2.12 Висновки до другого розділу

У другому розділі проведено детальне дослідження методів машинного навчання нейронних моделей для аналізу медичних даних. Розділ розпочався з аналізу процесу збору даних та їх попередньої обробки, визначаючи важливість цього етапу в підготовці даних для подальшого використання в моделях машинного навчання.

Далі в розділі детально розглядалися згорткові нейронні мережі, рекурентні нейронні мережі, короткострокова пам'ять, глибокі нейронні мережі прямого розповсюдження та глибокі згорткові нейронні мережі. Кожен метод був розглянутий у контексті його застосування до аналізу медичних даних, надаючи читачеві розуміння їхньої структури та особливостей використання в даному контексті.

Окрема увага була приділена проблемі оверфітінгу та методам регуляризації, оскільки ці аспекти важливі для підтримання стабільної та точної роботи нейронних моделей на основі медичних даних.

Розглянуто різноманітні техніки, спрямовані на запобігання перенавчанню та забезпечення загальної здатності моделей.

Додатково, в розділі був проведений аналіз важливого етапу машинного навчання – відбору ознак. Цей процес допомагає визначити найбільш важливі та впливові ознаки в медичних даних, що є ключовим для побудови ефективних моделей.

Розділ завершується оглядом програмних засобів для машинного навчання нейронних моделей, зокрема бібліотек Scikit-learn, SciPy та Joblib. Scikit-learn вирізняється своєю універсальністю та зручним інтерфейсом, в SciPy реалізовано багато функцій для наукових обчислень, а Joblib допомагає у реалізації паралельних обчислень для ефективної роботи з обсягами даних.

Отже, розділ надає розуміння в сфері використання методів машинного навчання для аналізу медичних даних та визначає важливість відповідних програмних інструментів для успішної реалізації та оптимізації моделей. Це дослідження виступає важливим етапом у розвитку сучасних підходів до обробки та аналізу медичних даних за допомогою інструментів машинного навчання.

## РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ

### 3.1 Постановка задачі

Мета завдання: класифікація медичних даних з використанням моделі випадкового лісу.

Для досягнення поставленої мети завдання програмним шляхом необхідно:

- завантажити медичний датасет «Breast Cancer Wisconsin»;
- розділити дані на навчальний і тестовий набори;
- нормалізувати дані для підготовки їх до моделювання;
- ініціалізувати модель випадкового лісу;
- здійснити пошук оптимальних параметрів для моделі за допомогою Grid Search і крос-валідації;
- навчити модель на навчальних даних та визначити її найкращі параметри;
- зберегти найкращу модель для подальшого використання;
- провести оцінку точності моделі на тестових даних;
- побудувати та вивести звіт про класифікацію;
- побудувати матрицю помилок та візуалізувати її у вигляді теплової карти;
- вивести важливість ознак у моделі;
- побудувати граф структури першого дерева у випадковому лісі;
- візуалізувати криву навчання для оцінки моделі на навчальних та тестових даних;
- використовуючи StratifiedKFold, обчислити крос-валідаційні оцінки точності, precision, recall та F1 для найкращої моделі;

– вивести середні значення та стандартні відхилення для кожної метрики.

### 3.2 Процес виконання програмного коду

Програмний код вирішує задачу класифікації на основі медичного датасету «Breast Cancer Wisconsin» за допомогою моделі випадкового лісу.

Покроковий опис процесу виконання розробленого програмного коду:

1) завантаження даних та підготовка. Завантажуються дані медичного датасету «Breast Cancer Wisconsin» за допомогою `load_breast_cancer()` з бібліотеки `Scikit-learn`. Дані розділяються на ознаки `X_medical` та цільову змінну `y_medical`. Дані поділяються на навчальний і тестовий набори (80% для навчання, 20% для тестування) за допомогою `train_test_split`;

2) нормалізація даних. Дані нормалізуються за допомогою `StandardScaler` для стандартизації ознак;

3) ініціалізація моделі випадкового лісу. Створюється модель випадкового лісу за допомогою `RandomForestClassifier` з встановленим випадковим `seed (random_state=42)`;

4) пошук оптимальних параметрів моделі. Задається сітка параметрів для пошуку оптимальних значень за допомогою `Grid Search (param_grid_medical)`. Використовується `GridSearchCV` для визначення оптимальних параметрів моделі з використанням крос-валідації;

5) вивід оптимальних параметрів. Виводяться найкращі параметри для моделі після завершення `Grid Search`;

6) збереження та завантаження моделі Найкраща модель зберігається у файл «`best_model_medical.pkl`» за допомогою `joblib.dump`. Та потім завантажується з файлу за допомогою `joblib.load`;

7) навчання та оцінка моделі. Найкраща модель навчається на навчальних даних. Здійснюється передбачення класів для тестових даних. Оцінюється точність моделі, виводиться звіт про класифікацію та будується матриця помилок;

8) виводиться графік важливості ознак, що визначає внесок кожної ознаки у визначення класів;

9) побудова та виведення графу структури першого дерева у випадковому лісі. Побудова теплової карти для візуалізації матриці помилок;

10) використовується функція `plot_learning_curve` для побудови кривої навчання та оцінки точності на навчальних та тестових даних;

11) використовується `StratifiedKfold` для крос-валідації. Обчислюються крос-валідаційні оцінки для точності, `precision`, `recall` та `F1`;

12) виводяться середні значення та стандартні відхилення для кожної метрики крос-валідації.

### 3.3 Використані технології

Використані під час розроблення програмного коду бібліотеки можна представити наступним чином:

1) `scikit-learn (sklearn)`. `Scikit-learn` використовується для завантаження медичного датасету, розділення даних, стандартизації ознак, ініціалізації та налаштування моделі випадкового лісу, а також для оцінки та візуалізації результатів. Вона включає в себе різноманітні інструменти для машинного навчання, крос-валідації та оптимізації моделей;

2) `Matplotlib` використовується для побудови графіків, таких як графіки важливості ознак, криві навчання та інші візуалізації;

3) `Seaborn` використовується для покращення візуального вигляду деяких графіків, зокрема для побудови теплової карти матриці помилок;



4) Joblib використовується для збереження та завантаження моделі. В цьому випадку, вона використовується для збереження та завантаження найкращої моделі випадкового лісу;

5) NumPy та SciPy використовуються для обробки та маніпулювання числовими даними, зокрема для роботи з масивами та матрицями. Ці бібліотеки є стандартними інструментами для розробки та аналізу даних у середовищі Python. Вони забезпечують широкий набір функцій для вирішення завдань машинного навчання, візуалізації даних та інших завдань аналізу даних.

### 3.4 Аналіз отриманих результатів

Після завершення процесу навчання нейронної моделі, була отримана порівняльна характеристика важливості ознак у моделі (рис. 3.1).

Аналіз важливості ознак (рис. 3.1) вказує на те, які параметри мають визначальний вплив на класифікацію.

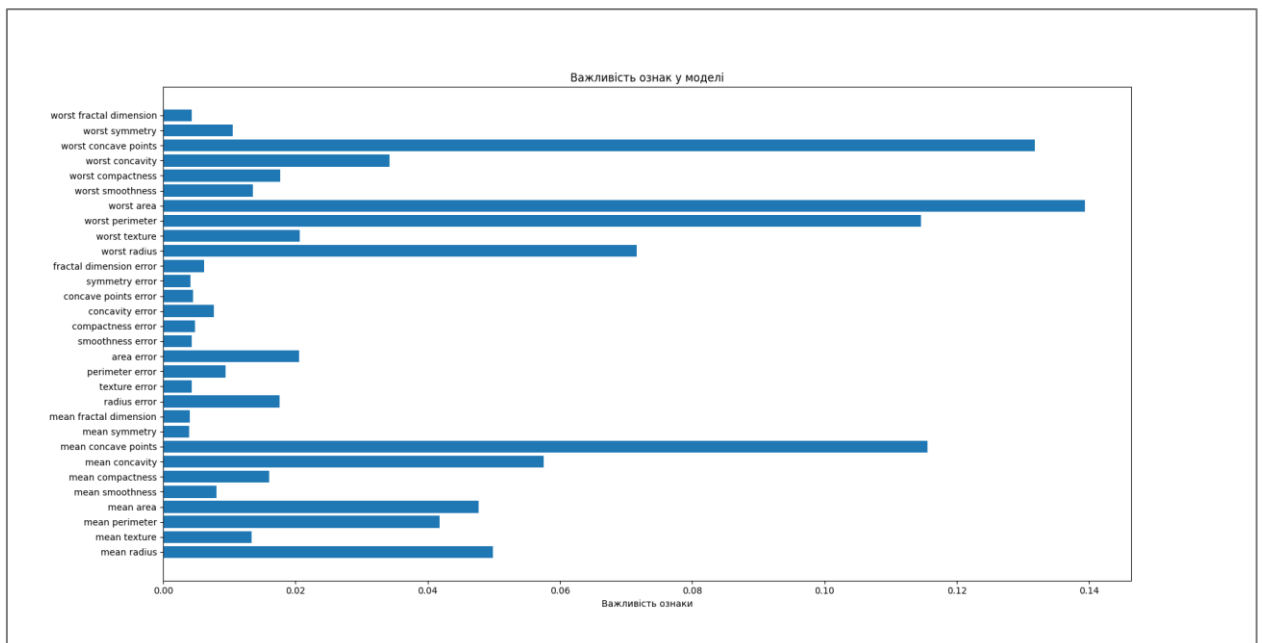


Рисунок 3.1 – Важливість ознак у моделі

На рисунку 3.2 зображена структура першого дерева у випадковому лісі.

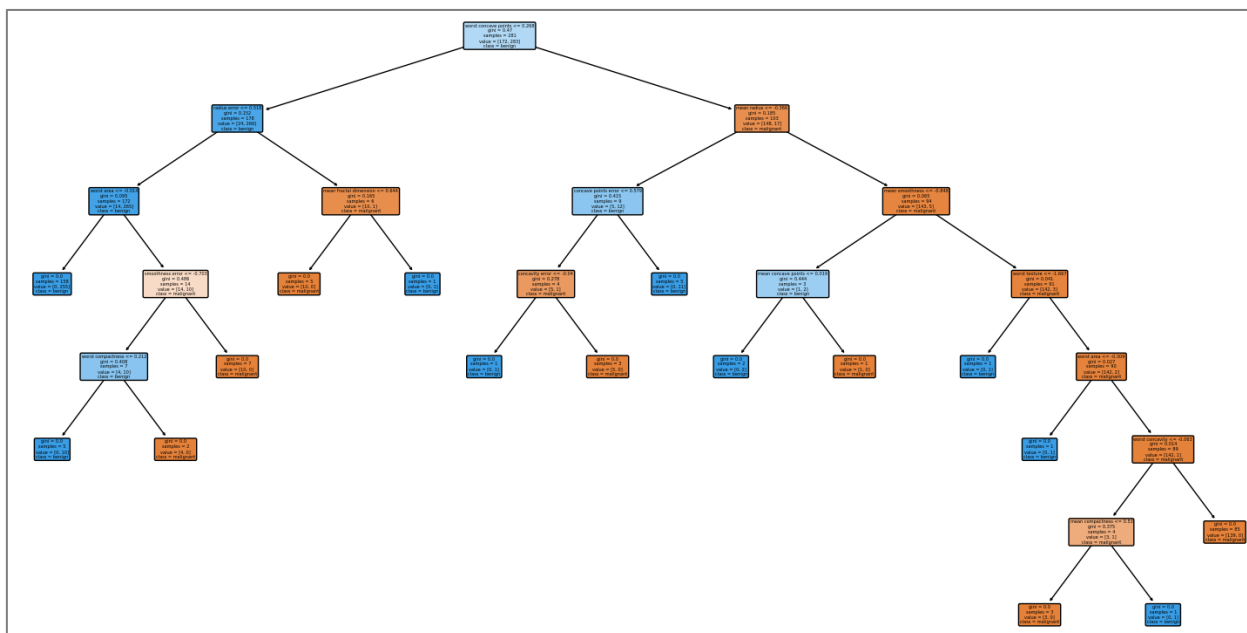


Рисунок 3.2 – Структура першого дерева у випадковому лісі

Матриця кореляції представлена на рисунку 3.3.

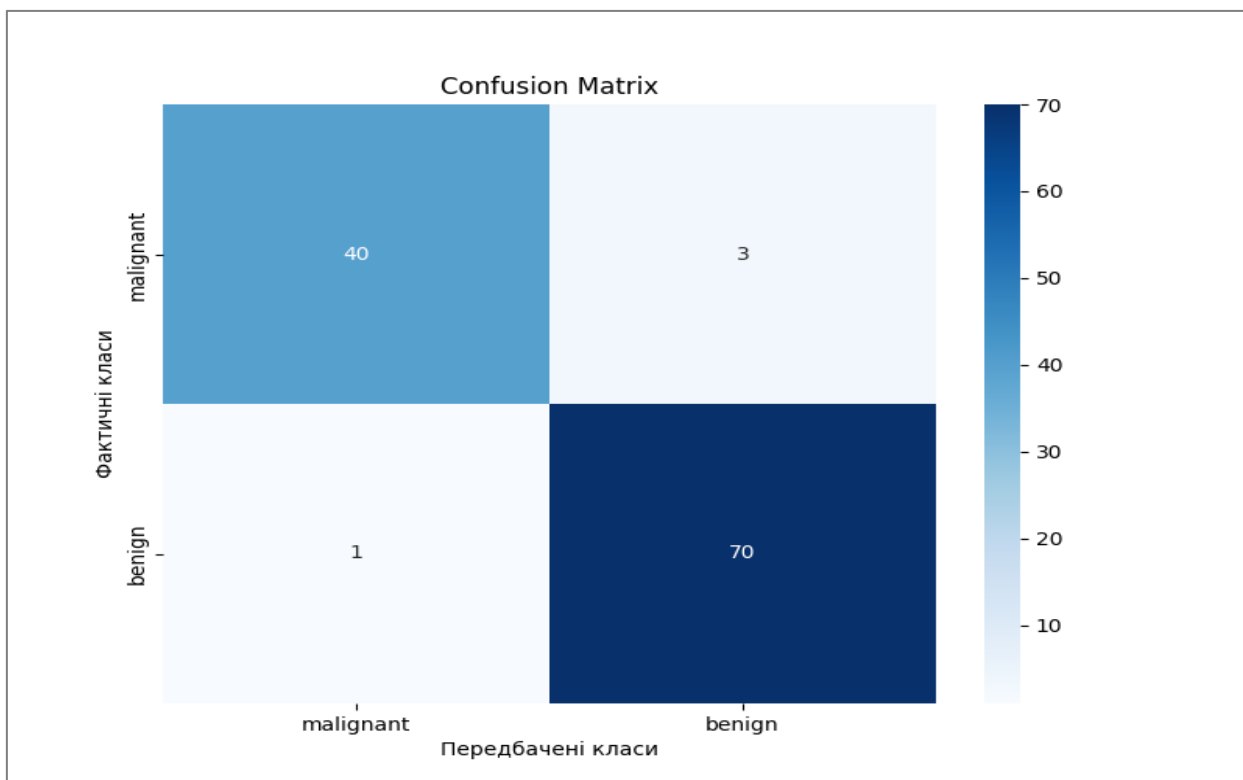


Рисунок 3.3 – Матриця кореляції

Також після завершення процесу навчання була побудована крива навчання, яка представлена на рисунку 3.4. З рисунку видно, що при зростанні розміру навчального набору, точність виборки зростає.

В цілому розроблена нейронна модель за допомогою сучасних інструментів підходить для аналізу медичних даних.

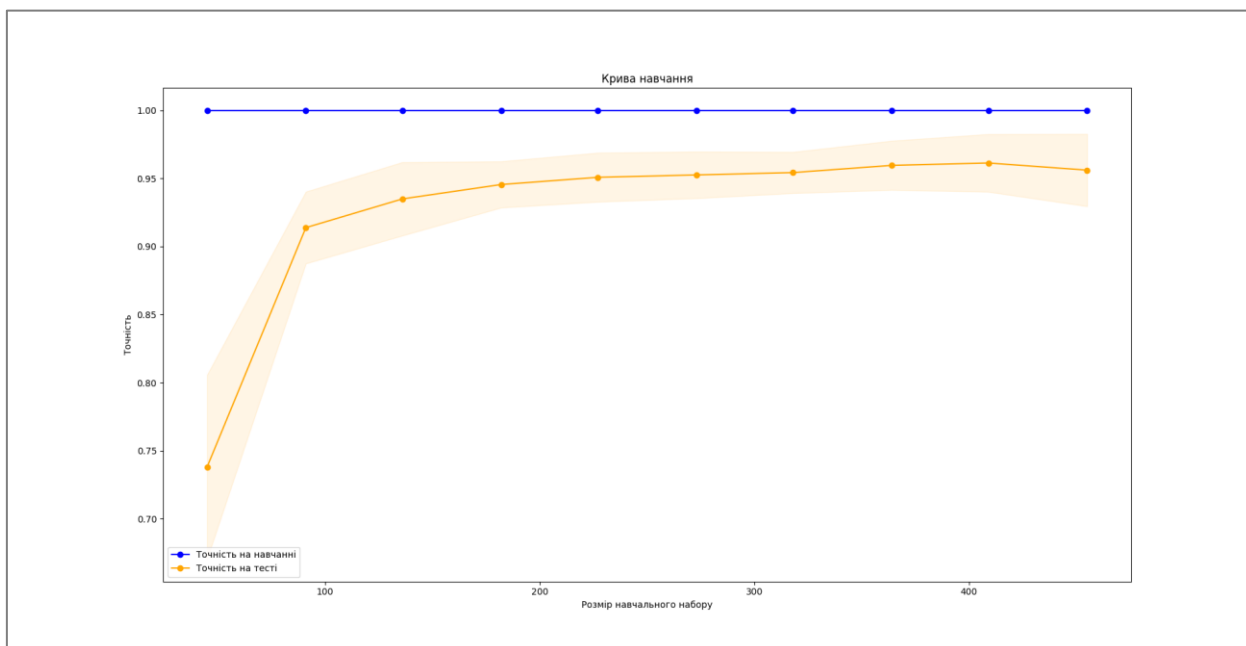


Рисунок 3.4 – Крива навчання

### 3.5 Висновки до третього розділу

У даному розділі проведено детальний аналіз та програмну реалізацію класифікації медичних даних на основі датасету «Breast Cancer Wisconsin». Задача полягала у використанні моделі випадкового лісу для ефективної класифікації медичних зразків на доброякісні та злоякісні.

Процес виконання програмного коду включав у себе кілька ключових етапів. Спочатку, медичні дані були завантажені та підготовлені, включаючи розділення на тренувальний та тестовий набори, а також нормалізацію ознак для підготовки до моделювання.

Далі, ініціалізувалася та оптимізувалася модель випадкового лісу, використовуючи Grid Search та крос-валідацію для пошуку оптимальних параметрів. Найкраща модель була збережена для подальшого використання. Після навчання моделі вона використовувалася для передбачення класів на тестових даних, а отримані результати були оцінені за допомогою різноманітних метрик, таких як точність, precision, recall та F1.

Також проведено візуалізацію результатів, включаючи графіки важливості ознак, структури першого дерева у випадковому лісі та теплову карту матриці помилок.

У процесі вирішення задачі використовувалися різні технології та бібліотеки. Бібліотека Scikit-learn використовувалася для завантаження даних, розділення наборів та ініціалізації моделі випадкового лісу. Matplotlib та Seaborn використовувалися для візуалізації результатів та створення графіків. Joblib використовувався для збереження та завантаження моделі. NumPy та SciPy були використані для операцій з масивами та матрицями.

Отримані результати свідчать про успішне вирішення задачі класифікації медичних даних. Модель випадкового лісу показала високу точність та ефективність у передбаченні класів «доброякісний» та «злроякісний». Аналіз важливості ознак вказує на те, які параметри мають визначальний вплив на класифікацію.

Загалом, використані технології та підходи до моделювання та візуалізації надали зрозумілі та достовірні результати, що можуть бути використані для подальшого вдосконалення та дослідження в галузі медичної класифікації.

## ВИСНОВКИ

Вступаючи в дослідження, було важливо провести аналіз предметної області, щоб визначити ключові аспекти та виклики, які існують у сфері застосування нейронних моделей для аналізу медичних даних.

Перший розділ підкреслив значення нейронних мереж, їхніх моделей та сучасний стан їх використання в медичній галузі.

Другий розділ був присвячений детальному дослідженню методів машинного навчання для нейронних моделей в контексті аналізу медичних даних. Вивчені різні аспекти, такі як збір та попередня обробка даних, використання згорткових та рекурентних нейронних мереж, глибоких нейронних мереж прямого розповсюдження та глибоких згорткових нейронних мереж.

Проблема оверфітінгу та методи регуляризації були акцентовані, оскільки вони є ключовими у забезпеченні стійкості та точності моделей.

Також вивчено важливий етап в машинному навчанні – відбір ознак, який допомагає визначити найбільш впливові аспекти в медичних даних.

Огляд програмних засобів для машинного навчання, таких як Scikit-learn, SciPy та Joblib, підкреслив їх значущість у розвитку та оптимізації нейронних моделей. Ці бібліотеки надають зручний та ефективний інструментарій для роботи з даними та виконання обчислень у медичному контексті.

У третьому розділі була виконана програмна реалізація, в якій були визначені постановка задачі, процес виконання програмного коду та використані технології. Аналіз результатів дослідження вказав на ефективність використаних методів та їхню придатність для розв'язання конкретних завдань у медичному аналізі. Загальні висновки з усіх розділів

підкреслюють важливість розуміння та використання сучасних методів машинного навчання для нейронних моделей у медичному дослідженні.

Дослідження визначає ключові етапи та техніки, які можуть сприяти успішній реалізації та оптимізації моделей для аналізу медичних даних, а також визначає переваги використання відповідних програмних інструментів у цьому контексті.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. X. Liu та Y. Zhong, “Research on Intelligent Diagnosis Model Based on the Medical Knowledge Graph of Multi-source Data Fusion”, у 2022 Int. Conf. Intell. Transp., Big Data Smart City (ICITBS), Hengyang, China, 26–27 берез. 2022. IEEE, 2022.

DOI: <https://doi.org/10.1109/icitbs55627.2022.00056>

2. E. Park, H. S. Nam та J.-W. Song, “Federated Learning Models using Flow Cytometry Data of Blood Test in Medical Decision Support”, у 2022 IEEE Int. Conf. Big Data (Big Data), Osaka, Japan, 17–20 груд. 2022. IEEE, 2022.

DOI: <https://doi.org/10.1109/bigdata55660.2022.10020379>

3. X. Cui, H. Liu, M. Tang та Y. Ma, “A Medical Pre-diagnosis Scheme Based on Neural Network and Inner Product Function Encryption”, у 2023 4th Int. Conf. Electron. Communication Artif. Intell. (ICECAI), Guangzhou, China, 12–14 трав. 2023. IEEE, 2023.

DOI: <https://doi.org/10.1109/icecai58670.2023.10176838>

4. N. Lu та R. Jiang, “An Adaptive Access Control Model Based on Trust and Risk for Medical Big Data”, у 2018 IEEE 3rd Int. Conf. Communication Inf. Syst. (ICCIS), Singapore, Singapore, 28–30 груд. 2018. IEEE, 2018.

DOI: <https://doi.org/10.1109/icomis.2018.8644815>

5. H. M. Kocak, A. T. Naskali, o. Pinarer та J. Mitard, “Detecting Transistor Defects in Medical Systems Using a Multi Model Ensemble of Convolutional Neural Networks”, у 2021 IEEE Int. Conf. Big Data (Big Data), Orlando, FL, USA, 15–18 груд. 2021. IEEE, 2021.

DOI: <https://doi.org/10.1109/bigdata52589.2021.9671667>

6. F. Lv, “Data Preprocessing and Apriori Algorithm Improvement in Medical Data Mining”, у 2021 6th Int. Conf. Communication Electron. Syst. (ICCES), Coimbatre, India, 8–10 лип. 2021. IEEE, 2021.

DOI: <https://doi.org/10.1109/icces51350.2021.9489242>

7. Z. Wang та ін., “A Prediction Model of Clinical Diagnosis by The Combination of Traditional Chinese and Western Medicine Based on Data Mining”, у 2021 Int. Conf. Intell. Transp., Big Data Smart City (ICITBS), Xi'an, China, 27–28 берез. 2021. IEEE, 2021.

DOI: <https://doi.org/10.1109/icitbs53129.2021.00188>

8. L.-J. Li, C.-Q. Niu, D.-X. Pu та X.-Y. Jin, “Electronic Medical Data Analysis Based on Word Vector and Deep Learning Model”, у 2018 9th Int. Conf. Inf. Technol. Medicine Educ. (ITME), Hangzhou, 19–21 жовт. 2018. IEEE, 2018.

DOI: <https://doi.org/10.1109/itme.2018.00114>

9. M. F. Khan, R. N. S. Kumar, T. Patil, A. Reddy, V. Mane та S. Santhoshkumar, “Neural Network Optimized Medical Image Classification with a Deep Comparison”, у 2022 Int. Conf. Augmented Intell. Sustain. Syst. (ICAISS), Trichy, India, 24–26 листоп. 2022. IEEE, 2022.

DOI: <https://doi.org/10.1109/icaiss55157.2022.10011109>

10. F. E. Fernandes та G. G. Yen, “Automatic Searching and Pruning of Deep Neural Networks for Medical Imaging Diagnostic”, IEEE Trans. Neural Netw. Learn. Syst., с. 1–11, 2020.

DOI: <https://doi.org/10.1109/tnnls.2020.3027308>

11. W. Gao, L. Chen та T. Shang, “Stream of Unbalanced Medical Big Data Using Convolutional Neural Network”, IEEE Access, т. 8, с. 81310–81319, 2020.

DOI: <https://doi.org/10.1109/access.2020.2991202>

12. H. Sun, Z. Liu, G. Wang, W. Lian та J. Ma, “Intelligent Analysis of Medical Big Data Based on Deep Learning”, IEEE Access, т. 7, с. 142022–142037, 2019.

DOI: <https://doi.org/10.1109/access.2019.2942937>



13. Y. Liu та B. Gao, “Research on medical data mining technology”, у 2017 IEEE 3rd Inf. Technol. Mechatronics Eng. Conf. (ITOEC), Chongqing, 3–5 жовт. 2017. IEEE, 2017.

DOI: <https://doi.org/10.1109/itoec.2017.8122473>

14. F. Hrvat, L. Spahic, L. G. Pokvic та A. Badnjevic, “Artificial Neural Networks for Prediction of Medical Device Performance based on Conformity Assessment Data: Infusion and perfusor pumps case study”, у 2020 9th Mediterranean Conf. Embedded Comput. (MECO), Budva, Montenegro, 8–11 черв. 2020. IEEE, 2020.

DOI: <https://doi.org/10.1109/meco49872.2020.9134359>

15. K. Huang, F. Ji, W. Lu та Y. Xiao, “Research on Text Generation of Medical Intelligent Question and Answer Based on Bi-LSTM and Neural Network Technology”, у 2022 IEEE/ACIS 22nd Int. Conf. Comput. Inf. Sci. (ICIS), Zhuhai, China, 26–28 черв. 2022. IEEE, 2022.

DOI: <https://doi.org/10.1109/icis54925.2022.9882349>

16. Y. Ren, Y. Shi, K. Zhang, Z. Chen та Z. Yan, “Medical Treatment Migration Prediction Based on GCN via Medical Insurance Data”, IEEE J. Biomed. Health Inform., т. 24, № 9, с. 2516–2522, верес. 2020.

DOI: <https://doi.org/10.1109/jbhi.2020.3008493>

17. M. Z. Ahmed та C. Mahesh, “A Comprehensive Review on Machine Learning Models for Medical Data Classification”, у 2021 2nd Int. Conf. Smart Electron. Communication (ICOSEC), Trichy, India, 7–9 жовт. 2021. IEEE, 2021.

DOI: <https://doi.org/10.1109/icosec51865.2021.9591907>

## ДОДАТКИ

## Додаток А

## Лістинг програмного коду

```
# Імпорт бібліотек
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split, GridSearchCV,
learning_curve, StratifiedKFold, cross_val_score
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
from sklearn.tree import plot_tree
import matplotlib.pyplot as plt
import joblib
import numpy as np
import seaborn as sns

# Завантажимо медичний датасет Breast Cancer Wisconsin
medical_data = load_breast_cancer()
X_medical = medical_data.data
y_medical = medical_data.target

# Розділимо дані на навчальний і тестовий набори
X_train_medical, X_test_medical, y_train_medical, y_test_medical =
train_test_split(
    X_medical,
```

```
y_medical,  
test_size=0.2,  
random_state=42  
)  
  
# Нормалізуємо дані  
scaler_medical = StandardScaler()  
X_train_medical = scaler_medical.fit_transform(X_train_medical)  
X_test_medical = scaler_medical.transform(X_test_medical)  
  
# Ініціалізуємо модель випадкового лісу  
model_medical = RandomForestClassifier(random_state=42)  
  
# Задаємо сітку параметрів для Grid Search  
param_grid_medical = {  
    'n_estimators': [50, 100, 150],  
    'max_depth': [None, 10, 20, 30],  
    'min_samples_split': [2, 5, 10],  
    'min_samples_leaf': [1, 2, 4]  
}  
  
# Створюємо об'єкт Grid Search із крос-валідацією  
grid_search_medical = GridSearchCV(model_medical, param_grid_medical, cv=5,  
scoring='accuracy')  
  
# Навчаємо Grid Search на навчальних даних  
grid_search_medical.fit(X_train_medical, y_train_medical)
```

```
# Виводимо найкращі параметри
print("Найкращі параметри:", grid_search_medical.best_params_)

# Отримуємо найкращу модель із Grid Search
best_model_medical = grid_search_medical.best_estimator_

# Зберігаємо найкращу модель
joblib.dump(best_model_medical, 'best_model_medical.pkl')

# Завантажуємо модель із файлу
loaded_model_medical = joblib.load('best_model_medical.pkl')

# Навчаємо найкращу модель на навчальних даних
best_model_medical.fit(X_train_medical, y_train_medical)

# Передбачаємо класи для тестових даних
y_pred_medical = best_model_medical.predict(X_test_medical)

# Оцінюємо точність моделі
accuracy_medical = accuracy_score(y_test_medical, y_pred_medical)
print(f"Точність моделі: {accuracy_medical}")

# Виводимо звіт про класифікацію
classification_rep_medical = classification_report(
    y_test_medical,
    y_pred_medical,
    target_names=medical_data.target_names
)
```

```
print("Звіт про класифікацію:\n", classification_rep_medical)

# Побудуємо матрицю помилок
conf_matrix_medical = confusion_matrix(y_test_medical, y_pred_medical)
print("Матриця помилок:\n", conf_matrix_medical)

# Візуалізуємо важливість ознак
feature_importances_medical = best_model_medical.feature_importances_
feature_names_medical = medical_data.feature_names

plt.barh(range(len(feature_importances_medical)), feature_importances_medical,
align='center')
plt.yticks(range(len(feature_importances_medical)), feature_names_medical)
plt.xlabel('Важливість ознаки')
plt.title('Важливість ознак у моделі')
plt.show()

# Візуалізуємо перше дерево в найкращій моделі
plt.figure(figsize=(12, 8))
plot_tree(
    best_model_medical.estimators_[0],
    feature_names=feature_names_medical,
    class_names=medical_data.target_names,
    filled=True,
    rounded=True
)
plt.title('Структура першого дерева у випадковому лісі')
plt.show()
```

```
# Візуалізуємо confusion matrix у вигляді теплової карти
```

```
plt.figure(figsize=(8, 6))
sns.heatmap(
    conf_matrix_medical,
    annot=True,
    fmt="d",
    cmap="Blues",
    xticklabels=medical_data.target_names,
    yticklabels=medical_data.target_names
)
plt.xlabel('Передбачені класи')
plt.ylabel('Фактичні класи')
plt.title('Confusion Matrix')
plt.show()
```

```
# Функція для побудови кривої навчання
```

```
def plot_learning_curve(model, X, y):
    train_sizes, train_scores, test_scores = learning_curve(
        model,
        X,
        y,
        cv=5,
        scoring='accuracy',
        train_sizes=np.linspace(0.1, 1.0, 10)
    )

    train_scores_mean = np.mean(train_scores, axis=1)
```

```

train_scores_std = np.std(train_scores, axis=1)
test_scores_mean = np.mean(test_scores, axis=1)
test_scores_std = np.std(test_scores, axis=1)

plt.figure(figsize=(10, 6))
plt.fill_between(train_sizes, train_scores_mean - train_scores_std,
                 train_scores_mean + train_scores_std, alpha=0.1, color='blue')
plt.fill_between(train_sizes, test_scores_mean - test_scores_std,
                 test_scores_mean + test_scores_std, alpha=0.1, color='orange')
plt.plot(train_sizes, train_scores_mean, 'o-', color='blue', label='Точність на
навчанні')
plt.plot(train_sizes, test_scores_mean, 'o-', color='orange', label='Точність на
тесті')
plt.xlabel('Розмір навчального набору')
plt.ylabel('Точність')
plt.title('Крива навчання')
plt.legend(loc='best')
plt.show()

# Візуалізуємо криву навчання для найкращої моделі
plot_learning_curve(best_model_medical, X_medical, y_medical)

# Використовуємо StratifiedKFold для крос-валідації
stratified_kfold_medical = StratifiedKFold(n_splits=5, shuffle=True,
random_state=42)

# Обчислимо крос-валідаційні оцінки точності для кращої моделі
cross_val_acc_medical = cross_val_score(

```

```
best_model_medical,  
X_medical,  
y_medical,  
cv=stratified_kfold_medical,  
scoring='accuracy'  
)  
cross_val_precision_medical = cross_val_score(  
    best_model_medical,  
    X_medical,  
    y_medical,  
    cv=stratified_kfold_medical,  
    scoring='precision_macro'  
)  
cross_val_recall_medical = cross_val_score(  
    best_model_medical,  
    X_medical,  
    y_medical,  
    cv=stratified_kfold_medical,  
    scoring='recall_macro'  
)  
cross_val_f1_medical = cross_val_score(  
    best_model_medical,  
    X_medical,  
    y_medical,  
    cv=stratified_kfold_medical,  
    scoring='f1_macro'  
)
```



*# Виведемо середні значення і стандартні відхилення для кожної метрики*

```
print(f'Середня точність: {np.mean(cross_val_acc_medical):.2f} ±  
{np.std(cross_val_acc_medical):.2f}')
```

```
print(f'Середня precision: {np.mean(cross_val_precision_medical):.2f} ±  
{np.std(cross_val_precision_medical):.2f}')
```

```
print(f'Середній recall: {np.mean(cross_val_recall_medical):.2f} ±  
{np.std(cross_val_recall_medical):.2f}')
```

```
print(f'Середня F1: {np.mean(cross_val_f1_medical):.2f} ±  
{np.std(cross_val_f1_medical):.2f}')
```