

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ перший (бакалаврський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

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

Освітня програма _____ Комп'ютерна інженерія _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Толок Степан Володимирович _____
(прізвище, ім'я, по батькові)

1. Тема роботи Модель і засоби машинного навчання у медичній інформаційній системі

затверджена наказом по університету від “ 26 ” травня 2025 р. № 425 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 14 липня 2025 р.

3. Вхідні дані до роботи 1. Модель повинна підтримувати бінарну або багатокласову класифікацію.

2. Результат моделі має інтерпретуватися у вигляді вірогідності або ризикової категорії.

3. Модель повинна працювати в умовах неповних або частково відсутніх даних (з підтримкою обробки пропущених значень).

4. Забезпечити зручний інтерфейс взаємодії моделі з МІС (REST API або локальна інтеграція).

4. Перелік питань, що потрібно опрацювати у роботі _____

1. Аналіз сучасного стану використання машинного навчання у МІС _____

2. Вибір методів машинного навчання _____

3. Проектування моделі машинного навчання та вибір засобів для її реалізації _____

4. Розробка програмного забезпечення _____

5. Тестування моделі _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

Слайд-презентація – 14 слайдів _____

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

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

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Строк / терміни виконання етапів роботи | Примітка |
|---|---|---|----------|
| 1 | Аналіз сучасного стану використання машинного навчання у МІС | 10.06.25-13.06.25 | |
| 2 | Вибір методів машинного навчання | 14.06.25-17.06.25 | |
| 3 | Проектування моделі машинного навчання та вибір засобів для її реалізації | 18.06.25-21.06.25 | |
| 4 | Розробка програмного забезпечення | 23.06.25-28.06.25 | |
| 5 | Тестування моделі | 30.06.25-02.07.25 | |
| 6 | Оформлення матеріалів кваліфікаційної роботи | 03.07.25-05.07.25 | |
| 7 | Подання кваліфікаційної роботи керівникові та її попередній захист | 07.07.25-09.07.25 | |
| 8 | Подання кваліфікаційної роботи на рецензування | 10.07.25-11.07.25 | |

Дата видачі завдання “ 09 ” червня 2025 р.

Здобувач _____

(підпис)

Керівник роботи _____

(підпис)

ст. викл. Дмитро РОСІНСЬКИЙ _____

(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 66 с., 10 рис., 5 табл., 1 дод., 32 джерела.

КЛАСИФІКАЦІЯ СТАНУ ПАЦІЄНТА, МАШИННЕ НАВЧАННЯ, МЕДИЧНА ІНФОРМАЦІЙНА СИСТЕМА, МЕДИЧНІ ДАНІ, ПРОГНОЗУВАННЯ РИЗИКІВ, RANDOM FOREST, REST API.

Метою роботи є дослідження та практична реалізація моделі машинного навчання разом із добором програмних засобів для її ефективної інтеграції в медичну інформаційну систему.

Досліджено можливості застосування методів машинного навчання у медичних інформаційних системах для підвищення ефективності прийняття клінічних рішень. Розроблено концептуальну модель МІС із вбудованим інтелектуальним модулем, що здійснює автоматизоване прогнозування ризику захворювання на основі медичних даних пацієнта. Обґрунтовано вибір алгоритму Random Forest як основи для побудови прогнозної моделі. Окрему увагу приділено аналізу та добору програмних засобів, таких як Python, scikit-learn, FastAPI, PostgreSQL, SHAP.

Виконано повний цикл реалізації системи: збирання та обробка даних, навчання моделі, створення REST-сервісу, інтеграція з інтерфейсом користувача. Проведено тестування розробленого рішення, що засвідчило високі значення ключових метрик. Отримані результати можуть бути впроваджені в клінічну практику для підтримки лікарських рішень.

ABSTRACT

Bachelor's thesis: 66 pages, 10 figures, 5 tables, 1 appendix, 32 sources.

CLINICAL DATA, INTELLIGENT DECISION SUPPORT SYSTEM, MACHINE LEARNING, MEDICAL INFORMATION SYSTEM, PATIENT CONDITION CLASSIFICATION, RANDOM FOREST, REST API, RISK PREDICTION.

The aim of this work is to investigate and implement a machine learning model along with selecting appropriate software tools for its effective integration into a medical information system.

This bachelor's thesis explores the application of machine learning techniques in medical information systems to enhance clinical decision-making. A conceptual model of a medical information system (MIS) was developed, incorporating an intelligent module capable of automatically predicting the risk of disease based on structured medical input. The Random Forest algorithm was chosen as the basis for the predictive model. Particular attention was paid to the analysis and selection of software tools, including Python, scikit-learn, FastAPI, PostgreSQL, and SHAP.

The complete implementation cycle was carried out, including data collection and preprocessing, model training, REST service development, and integration with a user interface. Testing of the implemented solution demonstrated high values of key metrics. The results were compared to existing solutions, confirming the competitiveness of the proposed approach. The obtained results can be implemented in clinical practice to support medical decision-making.

ЗМІСТ

| | |
|---|----|
| СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ | 9 |
| ВСТУП | 10 |
| 1 АНАЛІЗ СУЧАСНОГО СТАНУ ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ У МЕДИЧНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ | 12 |
| 1.1 Медичні інформаційні системи: призначення, структура та особливості функціонування | 12 |
| 1.2 Огляд технологій машинного навчання та їх потенціал у медицині | 14 |
| 1.3 Аналіз існуючих програмних рішень із застосуванням машинного навчання у медичній сфері | 15 |
| 1.3.1 Рішення в області аналізу медичних зображень | 16 |
| 1.3.2 Прогностичні системи та клінічна аналітика | 17 |
| 1.3.3 Платформи підтримки прийняття клінічних рішень | 17 |
| 1.3.4 Моніторинг та виявлення аномалій | 18 |
| 1.3.5 Технічні особливості та проблеми, що потребують розв'язання | 18 |
| 1.4 Постановка задачі дослідження та визначення вимог до моделі | 19 |
| 2 ВИБІР МЕТОДІВ МАШИННОГО НАВЧАННЯ | 21 |
| 2.1 Базові поняття та принципи машинного навчання | 21 |
| 2.1.1 Основні компоненти систем машинного навчання | 21 |
| 2.1.2 Типи навчання | 22 |
| 2.1.3 Етапи побудови ML-моделі | 22 |
| 2.1.4 Специфіка застосування ML у медичних задачах | 23 |
| 2.2 Огляд методів та алгоритмів машинного навчання для медичних задач | 25 |
| 2.2.1 Навчання з учителем | 25 |
| 2.2.2 Навчання без учителя | 26 |
| 2.2.3 Глибоке навчання | 27 |

| | |
|--|-----------|
| 2.2.4 Підкріплювальне навчання | 27 |
| 2.3 Порівняльний аналіз алгоритмів машинного навчання | 28 |
| 2.4 Метрики оцінювання якості моделей | 29 |
| 2.5 Обґрунтування вибору конкретних методів для реалізації у роботі | 33 |
| 3 ПРОЄКТУВАННЯ МОДЕЛІ МАШИННОГО НАВЧАННЯ ТА ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ | 35 |
| 3.1 Розробка концептуальної моделі медичної інформаційної системи із застосуванням машинного навчання | 35 |
| 3.2 Архітектура програмного рішення | 37 |
| 3.2.1 Характеристика основних компонентів | 38 |
| 3.2.2 Проєктування бази даних | 38 |
| 3.3 Вибір програмних засобів та бібліотек для реалізації моделей машинного навчання | 39 |
| 3.4 План реалізації моделі та її інтеграція в МІС | 41 |
| 3.4.1 Підготовка даних | 42 |
| 3.4.2 Побудова та навчання моделі | 42 |
| 3.4.3 Оцінювання та інтерпретація результатів | 43 |
| 3.4.4 Розгортання у вигляді мікросервісу | 43 |
| 3.4.5 Інтеграція з базою даних та МІС | 44 |
| 4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ | 45 |
| 4.1 Розробка програмної реалізації моделі машинного навчання | 45 |
| 4.2 Підготовка та обробка набору даних для навчання та тестування моделі | 47 |
| 4.2.1 Джерела даних | 47 |
| 4.2.2 Попередня обробка та нормалізація даних | 48 |
| 4.3 Тестування програмного рішення | 49 |
| 4.3.1 Методика тестування моделі | 50 |
| 4.3.2 Проведення експериментів та аналіз отриманих результатів | 51 |

| | |
|--|----|
| 4.4 Порівняльний аналіз отриманих результатів із існуючими аналогами | 52 |
| ВИСНОВКИ..... | 54 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ | 56 |
| ДОДАТОК А Графічний матеріал кваліфікаційної роботи..... | 59 |

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ЕМК – електронна медична картка

МІС – медична інформаційна система

API – програмний інтерфейс прикладного програмування (англ., Application Programming Interface)

AUC-ROC – площа під кривою робастного приймача (англ., Area Under the Receiver Operating Characteristic Curve)

CSV – формат значень, розділених комами (англ., Comma-Separated Values)

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

DNN – глибока нейронна мережа (англ., Deep Neural Network)

ML – машинне навчання (англ., Machine Learning)

PCA – метод головних компонент (англ., Principal Component Analysis)

REST – архітектурний стиль взаємодії систем (англ., Representational State Transfer)

RF – алгоритм випадкового лісу (англ., Random Forest)

SHAP – метод пояснення передбачень моделей (англ., SHapley Additive exPlanations)

SVM – метод опорних векторів (англ., Support Vector Machine)

UI – користувацький інтерфейс (англ., User Interface)

ВСТУП

Стрімкий розвиток інформаційних технологій, зокрема методів машинного навчання (ML), суттєво впливає на галузь охорони здоров'я. Медичні інформаційні системи (МІС) стають невід'ємною складовою процесів діагностики, моніторингу стану пацієнтів, обробки медичних даних та підтримки прийняття рішень. У цьому контексті інтеграція інтелектуальних алгоритмів у структуру МІС є не лише актуальним, але й необхідним кроком для підвищення якості медичних послуг, персоналізації лікування та оптимізації клінічних процесів. Застосування моделей машинного навчання дозволяє виявляти закономірності в медичних даних, які можуть бути недоступними для традиційного аналітичного підходу, що особливо важливо при роботі з великими обсягами неструктурованої або частково структурованої інформації [1].

Метою кваліфікаційної роботи є розробка концептуальної моделі та практичного інструментарію для використання алгоритмів машинного навчання в межах медичної інформаційної системи для підтримки клінічного прийняття рішень та автоматизації аналізу медичних даних.

Для досягнення поставленої мети були сформульовані такі основні завдання:

- здійснити аналіз сучасного стану застосування методів машинного навчання в МІС;
- розробити модель інтеграції алгоритмів машинного навчання в архітектуру медичної системи;
- реалізувати прототип системи з можливістю обробки реальних медичних даних;
- провести оцінювання ефективності побудованого рішення на основі визначених метрик точності та стабільності прогнозування.

Об'єктом дослідження є медичні інформаційні системи, що

підтримують збирання, зберігання та обробку даних пацієнтів, а також виконання допоміжних функцій у сфері охорони здоров'я.

Предметом дослідження є методи та моделі машинного навчання, які можуть бути інтегровані у структуру МІС з метою підвищення її інтелектуального потенціалу та автоматизації процесів медичної аналітики.

Результати дослідження мають практичне значення для розробників медичних програмних рішень та установ охорони здоров'я, які впроваджують цифрові інструменти для підвищення ефективності своєї роботи. Реалізований програмний прототип може бути адаптований для вирішення низки типових завдань у галузі охорони здоров'я, зокрема – виявлення груп ризику, прогнозування стану пацієнтів, автоматизації процесів обробки медичних записів. Крім того, побудована модель може бути використана як основа для подальших досліджень у сфері інтелектуальної медицини [2].

1 АНАЛІЗ СУЧАСНОГО СТАНУ ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ У МЕДИЧНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ

1.1 Медичні інформаційні системи: призначення, структура та особливості функціонування

Медичні інформаційні системи (МІС) є критично важливою складовою цифрової трансформації сучасної охорони здоров'я. Їхнє основне призначення полягає в забезпеченні автоматизованого збору, зберігання, обробки та аналізу медичної інформації з метою підвищення ефективності клінічних і адміністративних процесів, зменшення ризику помилок та покращення якості медичних послуг. У найбільш загальному вигляді МІС можна охарактеризувати як складні програмно-апаратні комплекси, що функціонують у тісному зв'язку з організаційною структурою медичного закладу та потребами його персоналу.

Функціональність МІС охоплює широкий спектр завдань: ведення електронної медичної документації, підтримку клінічних рішень, управління лабораторно-діагностичними процесами, адміністрування ресурсів закладу, формування звітності та інтеграцію з іншими національними або міжнародними інформаційними платформами охорони здоров'я. Важливу роль у структурі таких систем відіграють електронні медичні картки (ЕМК), що забезпечують централізований доступ до історії хвороби пацієнта, результатів обстежень, призначень та спостережень лікаря. У впровадженні МІС особливого значення набуває дотримання вимог щодо безпеки, конфіденційності та захисту персональних даних відповідно до чинного законодавства та міжнародних стандартів [3].

Типова архітектура медичної інформаційної системи має багаторівневу організацію, до якої входять (рисунки 1.1):

- рівень збору даних (модулі реєстрації, сенсорні пристрої, системи

введення вручну);

- рівень зберігання (бази даних, електронні сховища, репозиторії медичних записів);

- рівень обробки та логіки (аналітичні модулі, системи підтримки прийняття рішень, алгоритми контролю);

- рівень інтерфейсу (веб-додатки, мобільні інтерфейси, робочі станції медперсоналу).



Рисунок 1.1 – Багаторівнева архітектура типової МІС

Особливістю функціонування МІС є необхідність інтеграції різноманітних джерел даних – від структурованих реєстрів до неструктурованих медичних зображень та текстів. Ця багатоманітність ускладнює побудову уніфікованої моделі обробки інформації та вимагає високого рівня гнучкості від програмної архітектури. Крім того, медичні системи повинні працювати в умовах підвищених вимог до надійності,

доступності та безперервності обслуговування, оскільки будь-який збій може мати критичні наслідки для пацієнта.

Сучасні тренди розвитку МІС включають активну інтеграцію інструментів штучного інтелекту, зокрема машинного навчання, для підвищення аналітичних можливостей систем. Це відкриває нові перспективи для створення адаптивних, самонавчальних рішень, здатних підтримувати лікаря в складних діагностичних випадках або передбачати динаміку захворювання [4].

1.2 Огляд технологій машинного навчання та їх потенціал у медицині

Машинне навчання (ML) є підрозділом штучного інтелекту, що дозволяє створювати системи, здатні автоматично покращувати свою продуктивність на основі досвіду без явного програмування. У контексті медицини ML відкриває значні можливості для автоматизації обробки складних даних, побудови діагностичних та прогностичних моделей, оптимізації лікувальних стратегій, виявлення закономірностей, неочевидних для людини.

Сучасні алгоритми машинного навчання умовно поділяються на кілька груп залежно від типу навчання. Навчання з учителем (supervised learning) широко використовується в задачах класифікації (наприклад, визначення наявності захворювання за симптомами або медичними зображеннями) та регресії (прогнозування рівня ризику ускладнень). Типовими алгоритмами є логістична регресія, дерева рішень, випадкові ліси (Random Forest), метод опорних векторів (SVM) та нейронні мережі.

Навчання без учителя (unsupervised learning) дозволяє виявляти структуру в даних без наявності міток або очікуваних результатів. Цей підхід застосовується для кластеризації пацієнтів, виявлення аномалій у фізіологічних даних, а також для побудови профілів захворювань. До поширених алгоритмів належать метод k-середніх, ієрархічна кластеризація

та метод головних компонент (PCA).

Підкріплювальне навчання (reinforcement learning), хоча менш поширене в реальній клінічній практиці, демонструє перспективність у динамічному управлінні лікуванням, особливо при розробці адаптивних систем підтримки рішень у інтенсивній терапії або хронічному лікуванні.

Особливу роль у медичних дослідженнях відіграє глибоке навчання (deep learning), що базується на глибоких штучних нейронних мережах. Його потенціал найбільш відчутний у задачах комп'ютерного зору – розпізнавання патологій на знімках КТ, МРТ, рентгену або гістопатологічних препаратах. Конволюційні нейронні мережі (CNN) здатні досягати рівня точності, порівнянного з досвідченими лікарями-радіологами [5].

Крім того, застосування рекурентних нейронних мереж (RNN, LSTM) у прогнозуванні часових рядів дозволяє будувати моделі розвитку захворювань у часі, аналізувати динаміку життєвих показників або відповідь на лікування. Завдяки цьому ML-технології набувають дедалі більшого значення в системах моніторингу пацієнтів, особливо при застосуванні носимих пристроїв та телемедичних сервісів [6].

Разом із зростанням доступності електронних медичних записів та біомедичних баз даних, інструменти машинного навчання відіграють дедалі вагомішу роль у медичній науці, відкриваючи шлях до персоналізованої медицини, автоматизованої діагностики, а також нових підходів до виявлення факторів ризику та ефективного планування лікування.

1.3 Аналіз існуючих програмних рішень із застосуванням машинного навчання у медичній сфері

Протягом останнього десятиліття ринок медичних програмних продуктів із підтримкою машинного навчання демонструє стабільне зростання, що зумовлено як появою великих обсягів медичних даних, так і потребою у підвищенні ефективності клінічної діагностики, прогнозування

та персоналізованої терапії. Існуючі програмні рішення у цій сфері можна умовно класифікувати за основними функціональними напрямками (рисунок 1.2): діагностика на основі зображень, прогнозування результатів лікування, підтримка клінічних рішень, виявлення аномалій та автоматизований моніторинг. На рисунку 1.2 середній рівень ілюструє три основні напрямки застосування, а нижній рівень – перехресна функція моніторингу та виявлення аномалій, яка підтримує, доповнює або реалізує частину функціоналу в кожному з напрямків.



Рисунок 1.2 – Програмні рішення із застосуванням ML у медичній сфері

1.3.1 Рішення в області аналізу медичних зображень

Одним із найбільш технологічно зрілих напрямів є застосування глибокого навчання в обробці медичних зображень. Зокрема, платформа Aidoc є прикладом комерційного рішення, яке використовує конволюційні нейронні мережі для автоматичного виявлення патологій на КТ-знімках. Система аналізує зображення в режимі реального часу та повідомляє лікарів про потенційно критичні аномалії, такі як внутрішньочерепні крововиливи або легеневі емболії [7].

Іншим прикладом є Google DeepMind Health, яка у співпраці з Moorfields Eye Hospital розробила модель діагностики захворювань сітківки ока з використанням глибоких нейронних мереж. Результати дослідження показали, що точність діагностики моделі була порівнянною з провідними офтальмологами [8]. Подібні підходи застосовуються й у проектах Lunit INSIGHT та Zebra Medical Vision, які спеціалізуються на розпізнаванні аномалій у рентгенографії та мамографії.

1.3.2 Прогностичні системи та клінічна аналітика

Інша категорія систем орієнтована на прогнозування розвитку захворювань та ризиків ускладнень. Наприклад, система IBM Watson for Oncology, використовуючи методи глибокого навчання та обробки природної мови, аналізує медичні записи, результати аналізів та наукові публікації для формування персоналізованих рекомендацій щодо лікування онкологічних пацієнтів. Незважаючи на певну критику щодо узагальненості рекомендацій, Watson залишається одним із наймасштабніших прикладів застосування AI у медицині [9].

Система Epic Sepsis Model, вбудована в інформаційне середовище клінік, здійснює моніторинг стану пацієнтів у режимі реального часу та за допомогою статистичних моделей прогнозує ризик сепсису, що дає змогу вжити заходів до появи клінічних проявів. Однак ефективність моделі залежить від контексту її використання і вимагає налаштування з урахуванням локальних даних [10].

1.3.3 Платформи підтримки прийняття клінічних рішень

Розвиток CDSS (Clinical Decision Support Systems) також активно інтегрує машинне навчання. Наприклад, система Pillar розроблена компанією Tempus пропонує лікарям онкологічні рішення на основі молекулярного

профілю пацієнта та бази клінічних випробувань. Подібні системи дозволяють не лише рекомендувати терапію, а й формувати обґрунтування вибору, ґрунтуючись на доказовій базі.

Ще одним прикладом є Infermedica, яка поєднує симптом-чекер із ML-алгоритмами для попереднього медичного сортування. Система використовується в телемедичних сервісах для автоматичної триаж-оцінки пацієнтів та формування первинного висновку перед консультацією з лікарем.

1.3.4 Моніторинг та виявлення аномалій

Рішення на основі ML активно застосовуються також у сфері безперервного моніторингу стану пацієнтів. Наприклад, Current Health використовує носимі пристрої для збору життєвих показників і за допомогою ML-моделей оцінює ймовірність клінічного погіршення стану. Такі системи особливо корисні при амбулаторному лікуванні пацієнтів похилого віку або з хронічними захворюваннями.

Також варто згадати про Early Warning Systems, які використовуються в палатах інтенсивної терапії для автоматизованого попередження про небезпечні зміни у фізіологічних параметрах. У деяких випадках моделі працюють краще за стандартні шкали оцінки ризику, оскільки здатні виявляти складні нелінійні взаємозв'язки [10].

1.3.5 Технічні особливості та проблеми, що потребують розв'язання

Незважаючи на помітний прогрес, широкомасштабне впровадження ML-рішень у медицині стикається з низкою викликів. Найпоширеніші з них – обмежена доступність якісних розмічених даних, складність інтерпретації рішень моделей (problem of explainability), потреба у валідації рішень згідно з клінічними протоколами, а також питання етичного характеру та захисту

персональних даних. Відповідно, більшість ефективних систем є або вузькоспеціалізованими, або використовуються в дослідницькому режимі.

Крім того, результативність ML-рішень суттєво залежить від контексту їх використання, демографічних особливостей пацієнтів, рівня підготовки персоналу та наявної IT-інфраструктури. Таким чином, адаптація моделей до локального середовища залишається критично важливою умовою успішного впровадження.

1.4 Постановка задачі дослідження та визначення вимог до моделі

На основі аналізу сучасного стану розвитку медичних інформаційних систем та технологій машинного навчання сформульовано прикладну задачу, що відповідає актуальним потребам клінічної практики та дозволяє продемонструвати інтеграцію інтелектуальних алгоритмів у контексті підтримки медичних рішень.

Метою дослідження є створення моделі машинного навчання, яка здатна виконувати класифікацію ризику розвитку критичних ускладнень у пацієнтів на основі структурованих медичних даних, отриманих з електронної медичної картки. Зокрема, увага зосереджується на задачі раннього виявлення ймовірності розвитку гострої дихальної недостатності або сепсису у стаціонарних пацієнтів.

Вхідними даними для моделі є:

- демографічна інформація (вік, стать);
- ключові клінічні показники (температура, пульс, сатурація, артеріальний тиск);
- історія хронічних захворювань;
- результати лабораторних досліджень;
- поточні симптоми та медикаментозне лікування.

Вихідним значенням є ймовірність належності пацієнта до однієї з ризикових груп (наприклад, висока, середня або низька ймовірність

ускладнень).

Функціональні вимоги:

- модель повинна підтримувати бінарну або багатокласову класифікацію;
- результат моделі має інтерпретуватися у вигляді вірогідності або ризикової категорії;
- модель повинна працювати в умовах неповних або частково відсутніх даних (з підтримкою обробки пропущених значень);
- необхідно забезпечити зручний інтерфейс взаємодії моделі з медичною інформаційною системою (REST API або локальна інтеграція).

Нефункціональні вимоги:

- інтерпретованість моделі – необхідність обґрунтування результатів (наприклад, через SHAP, LIME або правила класифікації);
- висока чутливість (recall) – важлива для виявлення пацієнтів з потенційно небезпечним станом;
- швидкодія – модель має давати прогноз у межах <1 с після надходження запиту;
- масштабованість – можливість адаптації моделі до роботи з великими масивами даних у лікарнях або національних платформах;
- безпечність – відповідність вимогам захисту персональних даних у медичному середовищі.

Обмеження:

- дані повинні бути анонімізованими та знеособленими;
- застосування моделі обмежене тільки тим контингентом пацієнтів, за яких доступні повні життєві показники при госпіталізації;
- для забезпечення узагальненості моделі необхідно використовувати дані з різних джерел або лікарень.

2 ВИБІР МЕТОДІВ МАШИННОГО НАВЧАННЯ

2.1 Базові поняття та принципи машинного навчання

Машинне навчання (англ. Machine Learning, ML) – це підрозділ штучного інтелекту, який вивчає методи створення алгоритмів, здатних автоматично виявляти закономірності у даних та приймати рішення без необхідності явного програмування логіки поведінки. У контексті медицини машинне навчання набуло особливої важливості як інструмент для підвищення ефективності клінічних рішень, персоналізації лікування, автоматизованого виявлення патологій та підтримки діагностики на основі великих обсягів даних, таких як електронні медичні картки, результати обстежень, лабораторні показники та зображення.

2.1.1 Основні компоненти систем машинного навчання

У найзагальнішому вигляді модель машинного навчання реалізує функцію прогнозування або класифікації, яка будується на основі емпіричних даних. Формально, ця модель виконує апроксимацію деякої цільової функції $f: X \rightarrow Y$, де X – простір вхідних ознак (наприклад, медичних параметрів), а Y – простір вихідних значень (наприклад, клас діагнозу, рівень ризику, прогнозований стан).

Навчальна вибірка представлена у вигляді множини прикладів $\{(x_i, y_i)\}_{i=1}^n$, які слугують базою для побудови моделі. Основною метою є отримання узагальнюючої функції, яка демонструватиме високу точність на нових, ще не бачених системою прикладах, що є ключовим критерієм ефективності.

2.1.2 Типи навчання

Машинне навчання поділяється на кілька основних типів, кожен з яких характеризується специфічним підходом до роботи з даними та застосовується залежно від клінічного завдання:

- навчання з учителем (supervised learning) – передбачає наявність навчального набору, де кожному об'єкту x_i відповідає відома мітка y_i . Цей тип навчання застосовується у задачах класифікації (наприклад, виявлення ризику серцево-судинних захворювань) та регресії (наприклад, прогноз рівня глюкози в крові);

- навчання без учителя (unsupervised learning) – використовується, коли цільова змінна відсутня. Системи цього типу намагаються знайти структуру у даних (кластеризація, зменшення розмірності), що дозволяє, наприклад, групувати пацієнтів за схожими патернами перебігу захворювання;

- підкріплювальне навчання (reinforcement learning) – базується на взаємодії агента з середовищем, у якому він отримує зворотній зв'язок у вигляді винагороди. У медицині цей підхід актуальний для задач адаптивного контролю терапії та управління дозуванням медикаментів у режимі реального часу;

- глибоке навчання (deep learning) – це спеціалізований напрямок навчання з учителем, в якому використовуються глибокі нейронні мережі (DNN) з великою кількістю прихованих шарів. Особливу ефективність такі моделі демонструють у задачах розпізнавання зображень (КТ, МРТ), обробки електрокардіограм та прогнозування на основі часових рядів [12].

2.1.3 Етапи побудови ML-моделі

Незалежно від типу обраного навчання, процес машинного навчання включає послідовні етапи:

Етап 1. Збирання та підготовка даних: включає агрегацію джерел (ЕМК, біомедичні прилади), очищення, стандартизацію, імпутацію пропущених значень, нормалізацію числових змінних.

Етап 2. Формування ознакового простору: здійснюється перетворення сирих даних на набір репрезентативних числових ознак, що максимально корелюють із цільовим прогнозом. Може включати також відбір найінформативніших ознак (feature selection).

Етап 3. Вибір та навчання моделі: на цьому етапі обирається відповідний алгоритм – дерево рішень, логістична регресія, Random Forest, нейронна мережа тощо – і здійснюється його навчання з використанням навчальної вибірки.

Етап 4. Оцінювання ефективності: модель тестується на відкладеній (тестовій) частині даних із використанням якісних метрик (точність, повнота, F1, AUC, специфічність тощо) – детально це розглядатиметься далі.

Етап 5. Інтеграція та застосування: після валідації модель впроваджується у медичне середовище, наприклад, як модуль підтримки прийняття рішень у МІС або в системі моніторингу пацієнтів.

Крім технічної ефективності, важливою вимогою до медичних моделей є інтерпретованість – тобто можливість пояснити, чому саме модель прийняла певне рішення щодо пацієнта. Це критично для клінічного застосування, оскільки лікар повинен розуміти логіку прогнозу. У зв'язку з цим активно використовуються методи пояснюваного машинного навчання (Explainable AI, XAI), такі як SHAP, LIME, attention-механізми [13].

2.1.4 Специфіка застосування ML у медичних задачах

Медичні дані часто мають такі особливості, як неоднорідність, наявність пропущених значень, дисбаланс класів, високу кореляцію між ознаками, чутливість до артефактів. Тому застосування стандартних ML-алгоритмів вимагає їх адаптації: використання регуляризації, розширення

вибірки (data augmentation), балансування класів (SMOTE), а також додаткових етапів клінічної валідації.

У медичній практиці важливо також враховувати ризик помилок першого роду (false positive) та другого роду (false negative). Вибір компромісу між ними залежить від клінічного сценарію. Наприклад, у задачах виявлення сепсису критичним є мінімізувати хибнонегативні прогнози (щоб не пропустити пацієнта з реальним ризиком), тоді як у задачах планування дороговартісних обстежень перевага надається зменшенню хибнопозитивних результатів.

Таблиця 2.1 – Типи машинного навчання

| Тип навчання | Основні алгоритми | Типові медичні задачі |
|--------------------------|--|---|
| Навчання з учителем | Логістична регресія Random Forest SVM Нейронні мережі | Класифікація діагнозів Прогноз ризику ускладнень Оцінка виживаності |
| Навчання без учителя | K-середніх DBSCAN PCA Autoencoders | Кластеризація пацієнтів Виявлення аномалій Аналіз геноміки |
| Глибоке навчання | CNN RNN/LSTM GNN | Аналіз зображень КТ/МРТ Прогнозування на часових рядах Аналіз EHR |
| Підкріплювальне навчання | Q-learning Policy Gradients | Адаптивне дозування Управління станом у реальному часі |

2.2 Огляд методів та алгоритмів машинного навчання для медичних задач

Методи машинного навчання охоплюють широкий спектр алгоритмів, які можуть бути адаптовані до різних типів медичних задач – від класифікації пацієнтів до прогнозування ускладнень або обробки зображень. У межах даного дослідження розглядаються найбільш поширені та практично застосовні алгоритми з трьох основних категорій: навчання з учителем, навчання без учителя та глибоке навчання.

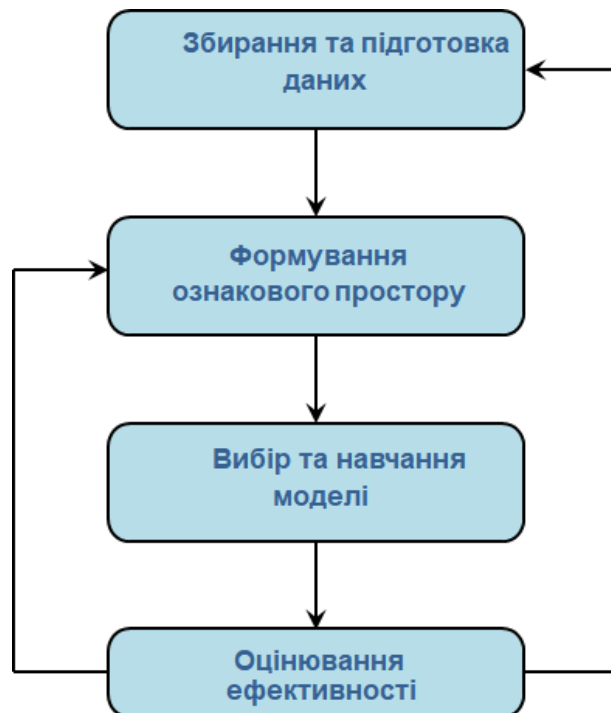


Рисунок 2.1 – Схема процесу машинного навчання

2.2.1 Навчання з учителем

Алгоритми цієї групи є найбільш придатними для реалізації у практичних медичних рішеннях, оскільки дозволяють будувати моделі на основі вже відомих результатів діагностики чи клінічного перебігу. Серед них:

- логістична регресія – базовий інтерпретований метод, придатний для

задач бінарної класифікації. Його перевагою є можливість оцінити вагу кожної змінної, що робить його придатним для клінічної інтерпретації (наприклад, при оцінці факторів ризику) [14];

- дерева рішень та ансамблеві методи (Random Forest, XGBoost) – забезпечують високу точність за рахунок комбінування декількох моделей. Особливо ефективні для задач з великою кількістю неоднорідних вхідних ознак [15];

- метод опорних векторів (SVM) – добре працює в умовах обмежених наборів даних та високої розмірності, часто застосовується для класифікації складних патернів у лабораторних показниках чи електрофізіологічних сигналах [16];

- багатошарові перцептрони (MLP) – клас нейронних мереж, які ефективно використовуються для регресійних задач та задач з кількома класами, за умови ретельної підготовки даних [17].

2.2.2 Навчання без учителя

Цей підхід використовується в умовах, коли дані не мають позначених результатів. У медицині це часто стосується задач виявлення нових підтипів хвороб або прихованих закономірностей:

- кластеризація (K-середніх, DBSCAN) дозволяє об'єднувати пацієнтів у групи за подібністю клінічних показників чи реакцією на терапію [15];

- методи зменшення розмірності (PCA, t-SNE) використовуються для візуалізації та виявлення структур у багатовимірних медичних даних (наприклад, у геноміці) [16];

- autoencoders – тип нейронних мереж, які дозволяють створювати стислий представницький простір для подальшого виявлення аномалій або ознак ускладнень на ранніх стадіях [17].

2.2.3 Глибоке навчання

Глибокі нейронні мережі, особливо згорткові та рекурентні, показують найкращі результати в задачах з великою кількістю вхідних параметрів або складною структурою даних:

- CNN (Convolutional Neural Networks) – найефективніші при роботі з медичними зображеннями (МРТ, КТ, гістологія). Вони навчаються автоматично виділяти ознаки, релевантні для діагностики, без потреби в ручному проектуванні ознак [6, 15];
- RNN / LSTM (Recurrent Neural Networks) – дозволяють аналізувати часові ряди, наприклад, для прогнозування стану пацієнта на основі динаміки життєвих показників [17];
- Transformer-архітектури – останнім часом набувають популярності для роботи з послідовними клінічними записами, зокрема в контексті обробки електронних медичних карток [14, 16].

2.2.4 Підкріплювальне навчання

Незважаючи на обмежену поширеність у клінічній практиці, підкріплювальне навчання демонструє перспективи в адаптивному керуванні лікувальними стратегіями:

- Q-learning, Deep Q-Networks – використовуються для створення агентів, які навчаються оптимальної поведінки у середовищі (наприклад, управління інсулінотерапією) [5];
- Policy gradient methods – дозволяють будувати стратегії прийняття рішень на основі безперервної оцінки винагород, що особливо важливо для персоналізованих протоколів [6].

2.3 Порівняльний аналіз алгоритмів машинного навчання

Вибір оптимального алгоритму машинного навчання для конкретної медичної задачі є критично важливим етапом розробки. Універсального рішення не існує – ефективність моделі залежить від характеристик вхідних даних, типу задачі, обмежень інтерпретованості, вимог до обчислювальних ресурсів та очікуваної точності. У цьому підрозділі здійснюється порівняльний аналіз найбільш поширених алгоритмів, орієнтованих на задачі класифікації у клінічних умовах.

Однією з ключових переваг логістичної регресії є її проста інтерпретованість: модель генерує вагові коефіцієнти для кожної ознаки, що дозволяє лікарю безпосередньо оцінити вплив кожного фактору на ймовірність настання події. Однак її продуктивність різко знижується у випадках складних, нелінійно роздільних даних, що часто характерно для багатофакторних клінічних ситуацій [14].

Методи на основі дерев рішень, зокрема Random Forest та XGBoost, демонструють високу точність при роботі з неоднорідними, частково неповними або змішаними даними. Вони добре масштабуються та допускають обробку великої кількості ознак. Крім того, вони є менш чутливими до вибору гіперпараметрів, порівняно з нейронними мережами. Їхнім недоліком є відносно обмежена інтерпретованість, хоча останні підходи (наприклад, SHAP) дозволяють частково вирішити цю проблему [15].

Метод опорних векторів (SVM) здатен ефективно вирішувати задачі класифікації у випадках, коли кількість ознак значно перевищує кількість спостережень, що характерно для медичних задач з високою розмірністю (наприклад, аналіз генетичних маркерів або спектральних даних) [16]. Його слабким місцем є погане масштабування на великі вибірки, що обмежує його використання в лікарнях з потоками даних у режимі реального часу.

Глибокі нейронні мережі (DNN, CNN, LSTM) демонструють найвищу

ефективність при роботі зі складними джерелами інформації, зокрема зображеннями, часовими рядами або електронними медичними записами. Такі архітектури дозволяють автоматично витягувати високорівневі ознаки з «сирих» даних, що є особливо корисним у випадках, коли ручне конструювання ознак є складним або неможливим. Наприклад, CNN-моделі продемонстрували результативність, порівнювану з лікарями-дерматологами у задачах діагностики меланоми [15, 6]. Проте ці моделі є ресурсомісткими, потребують великих навчальних вибірок і мають слабку інтерпретованість, що обмежує їх застосування в клінічному середовищі без додаткових пояснювальних механізмів [13].

Підкріплювальне навчання, попри перспективність, все ще рідко використовується в клінічних продуктах через складність формалізації винагороди та довгий цикл навчання. Однак дослідження в галузі інтенсивної терапії та управління інсулінотерапією демонструють його потенціал [16].

Загалом, при виборі моделі для задачі класифікації ризику розвитку ускладнень у пацієнтів доцільно орієнтуватися на методи, які забезпечують компроміс між точністю, інтерпретованістю та обчислювальною ефективністю. Для задач із структурованими клінічними даними найбільш придатними є ансамблеві методи або нейромережеві підходи з поясненням результатів, у той час як для зображень або сигналів – безумовно, глибокі згорткові моделі.

2.4 Метрики оцінювання якості моделей

Оцінювання якості моделей машинного навчання є критично важливим етапом їх розробки та впровадження, особливо у чутливих галузях, таких як охорона здоров'я. Недостатньо просто побудувати модель, яка забезпечує певний рівень точності; необхідно також переконатися, що вона збалансовано працює з усіма класами, не допускає критичних помилок, а також надає передбачувані та стабільні результати при зміні вхідних даних.

Для цього застосовується низка кількісних метрик, які дають змогу оцінити різні аспекти поведінки моделі.

Найбільш базовою метрикою є загальна точність або «правильність» (accuracy), яка визначається як відношення правильно класифікованих прикладів до загальної кількості:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} ,$$

де: TP – істинно позитивні випадки (true positives); TN – істинно негативні; FP – хибно позитивні (false positives); FN – хибно негативні (false negatives).

Попри свою інтуїтивну зрозумілість, точність не є достатньою метрикою для медичних задач з дисбалансом класів, коли позитивні випадки (наприклад, реальні ускладнення) трапляються рідко. У таких випадках модель може показувати високу точність, ігноруючи важливі рідкісні події [15].

Для задач раннього виявлення ускладнень надзвичайно важливими є чутливість (recall) та специфічність (specificity).

Чутливість – ймовірність правильно виявити всі позитивні випадки (тобто, пацієнтів із реальним ризиком):

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} .$$

Специфічність – ймовірність правильно ідентифікувати негативні випадки (пацієнтів без ускладнень):

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} .$$

Вибір між цими метриками залежить від клінічного контексту. У випадках, коли критично не пропустити жодного пацієнта з ускладненням, перевага надається високій чутливості [16].

Точність (precision) відображає точність позитивних передбачень моделі:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

Вона є важливою у випадках, коли неправильне виявлення ускладнень призводить до зайвих витрат або втрати довіри до системи. Для балансування між precision та recall застосовується F1-міра, яка є гармонійним середнім:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Ця метрика особливо корисна у задачах, де важливо знайти компроміс між виявленням і точністю – наприклад, в автоматичному сортуванні пацієнтів для подальшої діагностики [17].

Ще однією важливою характеристикою моделі є її дискримінаційна здатність, тобто здатність відрізнити позитивні випадки від негативних. Для цього застосовується ROC-крива (Receiver Operating Characteristic), що графічно відображає залежність між чутливістю моделі та ймовірністю хибнопозитивних передбачень за різних значень порогу. Площа під цією кривою (AUC-ROC) є загальною мірою якості класифікації:

- AUC \approx 0,5 – модель не краще випадкового вгадування;
- AUC \approx 1,0 – ідеальна класифікація.

Моделі з AUC $>$ 0,85 вважаються ефективними для клінічного використання [6, 15].

У випадках, коли класифікація має кілька можливих результатів

(наприклад, "низький", "середній", "високий" ризик), кожна з наведених метрик розширюється за допомогою макро-, мікро- та зважених усереднень. Це дозволяє адекватно враховувати як баланс класів, так і їхню важливість для задачі.

Окрім математичних метрик, у медичних інформаційних системах важливо також враховувати практичні характеристики моделі:

- швидкодія (latency) – час до отримання прогнозу після введення даних;
- інтерпретованість (explainability) – можливість пояснення причини прогнозу;
- надійність – стабільність при невеликих змінах у даних;
- можливість інтеграції в існуючу МІС (через API або модулі підтримки рішень).

Ці критерії детально враховуються при обґрунтуванні вибору моделі у наступному підрозділі.

На рисунку 2.2 наведено графік, що ілюструє порівняння поширених моделей машинного навчання за метриками F1-міри та AUC-ROC.

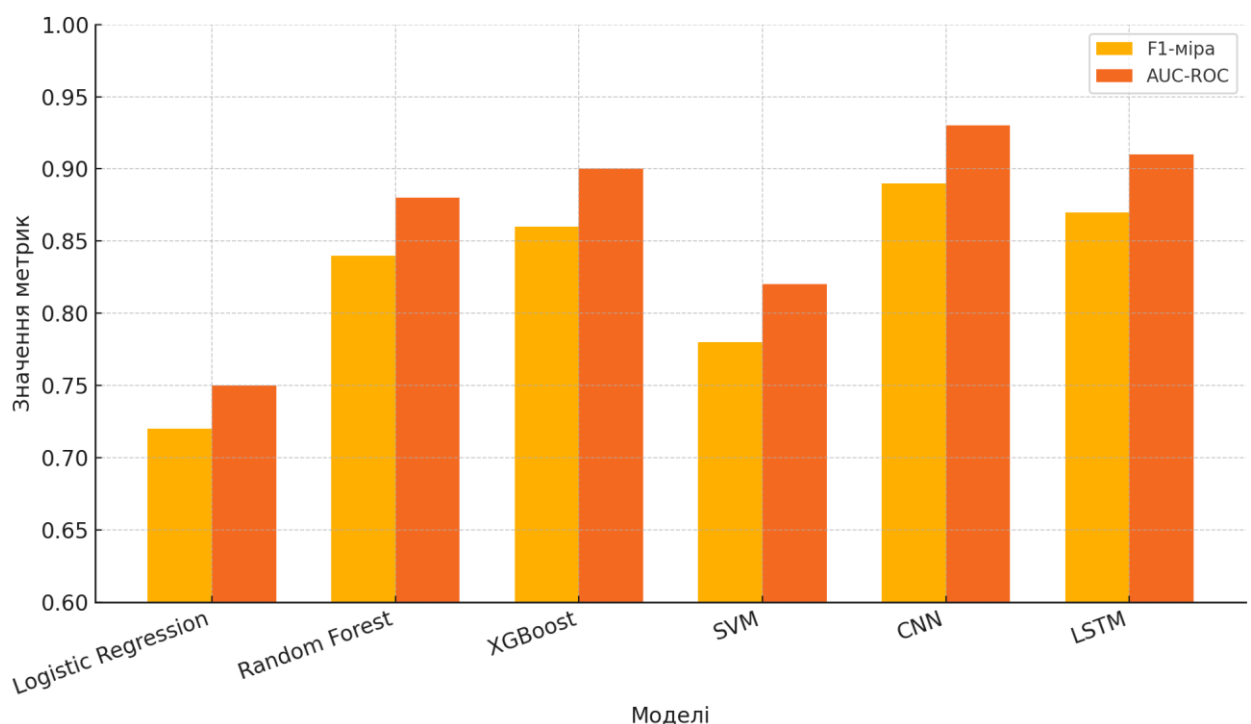


Рисунок 2.2 – Порівняння моделей ML за метриками F1-міри та AUC-ROC

2.5 Обґрунтування вибору конкретних методів для реалізації у роботі

Поставлена у межах цієї роботи задача полягає у побудові моделі машинного навчання, здатної здійснювати класифікацію рівня ризику розвитку ускладнень у пацієнтів на основі клінічних, демографічних та лабораторних параметрів. Особливість задачі полягає в необхідності високої чутливості до критичних випадків, роботи з структурованими, потенційно неповними медичними даними, а також у вимозі до інтерпретованості рішень у клінічному контексті.

На підставі проведеного порівняльного аналізу та оцінки метрик, доцільним видається застосування ансамблевого методу Random Forest як основної моделі, доповненої можливістю інтерпретації через методи пояснення рішень (наприклад, SHAP). Такий вибір зумовлений наступними чинниками.

Обґрунтування з точки зору якості прогнозу. Random Forest демонструє високу якість класифікації на структурованих медичних даних. У порівнянні з логістичною регресією, він здатен враховувати нелінійні залежності між ознаками, що особливо важливо при взаємодії біомаркерів і хронічних захворювань. На основі результатів оцінювання (рисунок 2.1), ця модель забезпечує F1-міру 0,84 та AUC-ROC близько 0,88, що є достатніми показниками для медичної класифікації [15].

Обґрунтування з точки зору практичності. Модель є менш вимогливою до ресурсів, ніж глибокі нейронні мережі, що дозволяє реалізувати її навіть у середовищах з обмеженою обчислювальною потужністю (наприклад, на базі локальної МІС в лікарні). Крім того, Random Forest стійкий до пропущених значень і менш схильний до переобучення, що важливо в умовах медичних даних з обмеженим обсягом.

Інтерпретованість. На відміну від CNN або LSTM, модель дерев рішень можна пояснити за допомогою методів інтерпретації на основі важливості ознак, SHAP-значень або локальних пояснень (LIME). Це дозволяє лікарям

розуміти, які фактори вплинули на присвоєння пацієнту певного ризикового класу, що підвищує рівень довіри до системи [13, 17].

Альтернативні сценарії. У разі подальшого масштабування або доступу до великих обсягів уніфікованих медичних записів, можна розглянути заміну базової моделі на глибоку нейронну мережу, зокрема багат шаровий перцептрон або LSTM. Це дозволить підвищити точність класифікації, однак потребує ретельного забезпечення обчислювальної інфраструктури та вирішення проблеми інтерпретованості [14, 16].

Таким чином, для реалізації задачі даної роботи обрано Random Forest з інтеграцією SHAP для інтерпретації результатів, що забезпечує збалансовану комбінацію якості, надійності та клінічної придатності.

Таблиця 2.1 – Порівняльна характеристика моделей машинного навчання

| Модель | F1-міра | AUC-ROC | Інтерпретованість | Обчислювальні ресурси | Стійкість до пропущених даних | Типові медичні задачі |
|---------------------|---------|---------|-------------------|-----------------------|-------------------------------|--|
| Logistic Regression | 0,72 | 0,75 | Висока | Низькі | Низька | Оцінка ризику ускладнень |
| Random Forest | 0,84 | 0,88 | Середня (SHAP) | Середні | Висока | Класифікація пацієнтів за ризиком |
| XGBoost | 0,86 | 0,90 | Низька | Високі | Середня | Прогнозування рецидивів |
| SVM | 0,78 | 0,82 | Середня | Середні | Низька | Діагностика на основі лабораторних даних |
| CNN | 0,89 | 0,93 | Низька | Високі | Низька | Аналіз медичних зображень |
| LSTM | 0,87 | 0,91 | Низька | Високі | Низька | Прогнозування за часовими рядами |

3 ПРОЄКТУВАННЯ МОДЕЛІ МАШИННОГО НАВЧАННЯ ТА ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ

3.1 Розробка концептуальної моделі медичної інформаційної системи із застосуванням машинного навчання

Інтеграція алгоритмів машинного навчання у медичні інформаційні системи передбачає не лише включення окремого аналітичного модуля, а й глибоку трансформацію логіки взаємодії між компонентами системи. Щоб забезпечити коректну та ефективну роботу інтелектуального модуля, необхідно на рівні концептуального проєктування визначити структуру системи, потоки даних, точки доступу до моделей, а також механізми взаємодії з користувачем і зовнішніми джерелами.

У межах даної роботи запропоновано концептуальну модель МІС, орієнтовану на раннє виявлення ризику розвитку ускладнень у пацієнтів на основі структурованих медичних даних. Вона охоплює такі основні функціональні блоки (рисунок 3.1):

- збір даних – отримання інформації з електронних медичних карток (ЕМК), лабораторних систем, сенсорних пристроїв;
- попередня обробка – нормалізація, фільтрація, імпутація пропущених значень;
- оцінювання ризику – застосування моделі машинного навчання;
- пояснення результатів – інтерфейс для інтерпретації прогнозу;
- збереження та зворотний зв'язок – фіксація результатів у базі даних та подання рекомендацій лікареві.

Процес починається з отримання даних пацієнта з внутрішніх джерел (ЕМК, лабораторії). Далі інформація надходить до модуля попередньої обробки, де вона готується до подачі в модель: перевіряються формати, заповнюються пропущені значення, нормалізуються числові показники. На

наступному етапі активується модуль прогнозування, який класифікує рівень ризику. Паралельно з результатом активується пояснювальний модуль (наприклад, SHAP), який повертає лікарю обґрунтовану інтерпретацію прогнозу. Усі результати зберігаються в базі даних, а лікар отримує доступ до візуалізації та рекомендацій у зручному інтерфейсі.



Рисунок 3.1 – Схема концептуальної моделі

Такий підхід забезпечує не лише автоматизовану діагностичну підтримку, а й підвищує прозорість прийняття рішень, що критично для клінічної практики. Далі буде деталізовано архітектурну реалізацію цієї концепції.

3.2 Архітектура програмного рішення

Ефективна реалізація медичної інформаційної системи з інтегрованою моделлю машинного навчання потребує ретельно спроектованої архітектури, яка поєднує модулі збору, обробки, аналізу, зберігання даних і взаємодії з користувачем. Враховуючи складність поставленої задачі, система має бути модульною, масштабованою та придатною до інтеграції з наявною ІТ-інфраструктурою медичних закладів.

У межах цієї роботи обрано тривірневу клієнт-серверну архітектуру (рисунок 3.2), де всі компоненти логічно поділені на:

- клієнтський рівень – інтерфейс взаємодії з лікарем;
- прикладний рівень (сервер) – логіка обробки та прогнозування;
- рівень зберігання – база медичних даних і журналів прогнозів.



Рисунок 3.2 – Клієнт-серверна архітектура

3.2.1 Характеристика основних компонентів

Веб-інтерфейс дозволяє лікарю вводити ID пацієнта або переглядати результати для вже зареєстрованих осіб. Додатково реалізуються функції виводу інтерпретацій моделі (на основі SHAP), візуалізація медичних трендів тощо.

API-шлюз слугує центральною точкою маршрутизації запитів. Реалізований із використанням легковагового веб-фреймворку (Flask або FastAPI), що забезпечує високу продуктивність та простоту обслуговування.

Модуль попередньої обробки виконує нормалізацію числових ознак, обробку пропущених значень, кодування категоріальних параметрів та формування векторів ознак для подачі в ML-модель.

Модуль машинного навчання реалізує передбачення ризику за допомогою попередньо навченої моделі Random Forest. Окрім прогнозу, модель генерує пояснення у форматі SHAP-значень – тобто ваги ознак, які найбільше вплинули на результат.

СУБД PostgreSQL виступає в ролі централізованого сховища. Її реляційна структура забезпечує надійне зберігання клінічних даних, журналів прогнозів та інтерпретацій. Доступ до бази захищено, передбачено резервне копіювання.

3.2.2 Проектування бази даних

В основі архітектури бази даних – реляційна модель, яка містить кілька таблиць, пов'язаних за зовнішніми ключами. Основні з них:

- patients – таблиця з інформацією про пацієнтів (ID, вік, стать, дата госпіталізації);
- clinical_data – клінічні показники, отримані з ЕМК (температура, тиск, лабораторні аналізи);
- ml_predictions – результати моделі: клас ризику, ймовірність, дата

прогнозу;

- `shap_values` – таблиця пояснень, у якій зберігаються ваги впливу кожної ознаки на рішення;
- `users` – адміністратори, лікарі, медсестри (для авторизації через вебінтерфейс).

Ця структура дозволяє ефективно запитувати як історію змін стану пацієнта, так і результати прогнозів та їхнє пояснення – без дублювання даних.

Таким чином, архітектура запропонованої системи поєднує принципи модульності, масштабованості та клінічної доцільності. Далі здійснено аналіз програмних засобів та обґрунтовано вибір технологічного стеку для реалізації кожного з компонентів.

3.3 Вибір програмних засобів та бібліотек для реалізації моделей машинного навчання

Успішна реалізація моделі машинного навчання в медичній інформаційній системі потребує вибору програмного забезпечення, що не лише відповідає функціональним вимогам, але й забезпечує масштабованість, підтримку сучасних алгоритмів, високу продуктивність, сумісність із системами баз даних та інтерфейсами. У межах цього підрозділу здійснюється огляд провідних мов програмування, бібліотек та середовищ, що використовуються для побудови моделей ML у медичних проєктах.

Python. Мова Python є де-факто стандартом у сфері прикладного машинного навчання. Її переваги включають:

- велику кількість бібліотек для обробки даних, моделювання, візуалізації;
- простий, лаконічний синтаксис, зручний для швидкої розробки;
- підтримку інтеграції з вебсервісами, БД, REST API;
- активну спільноту, що постійно оновлює пакети.

Python застосовується майже в усіх клінічно орієнтованих ML-дослідженнях, включно з такими платформами, як Google Health, IBM Watson або системи інтелектуального сортування пацієнтів [15, 17, 18].

Scikit-learn. Це базова бібліотека Python для реалізації класичних алгоритмів машинного навчання – логістичної регресії, дерев рішень, Random Forest, кластеризації, масштабування тощо. Перевагами є:

- висока стабільність і документованість;
- підтримка пайплайнів обробки;
- зручність підключення метрик та крос-валідації.

Scikit-learn чудово підходить для роботи з табличними даними, що властиво медичним системам, побудованим на основі структурованих клінічних ознак [19].

TensorFlow та Keras. TensorFlow – бібліотека з відкритим кодом, розроблена Google [20], орієнтована на реалізацію глибоких нейронних мереж. Keras є надбудовою над TensorFlow з простішим синтаксисом. Основні характеристики:

- потужна підтримка GPU/TPU-прискорення;
- реалізація як згорткових, так і рекурентних мереж;
- можливість збереження і експорту моделей у форматі SavedModel.

Недоліком є складність налаштування для невеликих проєктів та потреба в потужних ресурсах при використанні великих архітектур.

PyTorch. Розроблена Facebook Research, PyTorch є конкурентом TensorFlow, який став популярним завдяки своїй гнучкості та зручності налагодження:

- динамічне формування обчислювального графа;
- інтуїтивний синтаксис, зручний для наукових досліджень;
- широке застосування у медичному сегменті (наприклад, у MedNIST).

PyTorch особливо підходить для проєктів, у яких потрібна точна кастомізація моделей або інтеграція з дослідницькими модулями [21].

SHAP. Це бібліотека, яка реалізує методику пояснення рішень моделей

(Shapley Additive Explanations). У медичному контексті дозволяє лікарю зрозуміти, які саме фактори вплинули на передбачення моделі. Особливо важлива для прийняття рішень у діагностичних або прогностичних задачах [16, 22].

З огляду на потребу у роботі з табличними структурованими даними, невеликими обсягами вибірки, а також потребу у високій інтерпретованості рішень, для реалізації моделі обрано наступні засоби, наведені в таблиці 3.1.

Таблиця 3.1 – Вибір програмних засобів та бібліотек

| Компонент | Вибір | Причина |
|---------------------------|--------------------|---|
| Мова програмування | Python | Стандарт у ML, зручна інтеграція |
| Бібліотека ML | Scikit-learn | Random Forest, стабільність, документація |
| Інтерпретація результатів | SHAP | Пояснення для клінічного персоналу |
| Веб-фреймворк | FastAPI [25] | Продуктивність, швидка побудова REST API |
| СУБД | PostgreSQL | Сумісність з Python, розширюваність |
| Візуалізація | Plotly, Matplotlib | Інтерактивні графіки для лікаря |

Вказаний стек дозволяє побудувати гнучке рішення з можливістю масштабування та інтеграції в локальне медичне середовище без високих обчислювальних вимог.

3.4 План реалізації моделі та її інтеграція в МІС

Інтеграція моделі машинного навчання в медичну інформаційну систему потребує чітко структурованого плану дій, що охоплює увесь

життєвий цикл побудови інтелектуального модуля – від підготовки даних до впровадження у робоче середовище. Такий підхід дозволяє забезпечити узгодженість між компонуванням системи, точністю прогнозу та клінічною доцільністю. У цьому підрозділі представлено ключові етапи реалізації розробленої моделі.

3.4.1 Підготовка даних

На початковому етапі виконується збирання та попередня обробка медичних даних, зокрема:

- імпорт структурованих записів із електронних медичних карток (ЕМК), лабораторних систем, форм ручного введення;
- очищення від шуму та дублювання;
- заповнення пропущених значень методами середнього, медіанного чи класифікаційного імпутування;
- нормалізація числових ознак та кодування категоріальних змінних;
- поділ датасету на навчальну, валідаційну та тестову вибірки.

Для цього використовуються бібліотеки Pandas та scikit-learn, що забезпечують гнучкість обробки та автоматизацію процедури підготовки [19, 23].

3.4.2 Побудова та навчання моделі

На основі підготовлених даних виконується навчання моделі типу Random Forest, яка є оптимальною для роботи з табличними даними. Процедура включає:

- налаштування параметрів моделі (кількість дерев, максимальна глибина, критерій розбиття);
- виконання навчання на навчальній вибірці;
- проведення крос-валідації для вибору гіперпараметрів;

- збереження моделі у форматі `.joblib` або `.pkl` для подальшого розгортання.

Результатом етапу є стійка до перенавчання модель із чітко визначеною продуктивністю, перевіреною за метриками F1 та AUC-ROC.

3.4.3 Оцінювання та інтерпретація результатів

Модель проходить тестування на відкладеній вибірці з розрахунком ключових метрик (точність, чутливість, специфічність). Для інтерпретації прогнозу інтегрується бібліотека SHAP, яка дозволяє згенерувати пояснення для кожного передбачення [22]:

- візуалізація впливу ознак на прогноз;
- ідентифікація факторів ризику для конкретного пацієнта;
- інтеграція SHAP-графіків у вебінтерфейс лікаря.

Цей етап сприяє підвищенню довіри до системи з боку медичного персоналу.

3.4.4 Розгортання у вигляді мікросервісу

Модель та супутні модулі реалізуються як окремий серверний сервіс (API), що може бути інтегрований у МІС через HTTP-запити. Архітектура передбачає:

- реалізацію REST API за допомогою FastAPI [25];
- захист даних за протоколами авторизації (JWT, OAuth2);
- розміщення на внутрішньому сервері установи або хмарній інфраструктурі (за умови шифрування каналів передачі).

Вебінтерфейс, створений за допомогою React або Streamlit, забезпечує зручне подання результатів прогнозування у форматі табличних звітів і графічних пояснень.

3.4.5 Інтеграція з базою даних та МІС

Усі результати прогнозування зберігаються у базі даних PostgreSQL, що дозволяє:

- прив'язувати результати до історії хвороби пацієнта;
- зберігати версію моделі та дату прогнозу;
- виконувати зворотній аналіз ефективності прогнозу в динаміці;
- здійснювати агрегацію результатів для керівництва медичного закладу.

Система інтегрується з наявною МІС через стандартні API (HL7 FHIR [24], REST), що дозволяє органічно впровадити її без радикальної зміни ІТ-інфраструктури.

Таким чином, реалізація моделі машинного навчання охоплює повний життєвий цикл – від збору даних до розгортання прогнозуючого сервісу в клінічному середовищі. Впровадження здійснюється поступово, з урахуванням вимог щодо безпеки, інтерпретованості та клінічної релевантності.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

4.1 Розробка програмної реалізації моделі машинного навчання

Після визначення архітектури та вибору інструментів реалізується прототип системи прогнозування ризику на основі алгоритмів машинного навчання. Основною метою програмної реалізації є побудова моделі, яка здатна здійснювати обробку вхідних медичних даних, генерувати прогноз, забезпечувати його інтерпретацію та інтегруватися з медичною інформаційною системою (МІС).

Реалізація моделі базується на використанні бібліотеки `scikit-learn`, як однієї з найбільш стабільних та придатних до роботи з табличними даними [19]. Вибрано алгоритм `Random Forest`, що поєднує стабільність дерев рішень із зменшенням ризику перенавчання. Реалізація містить такі основні кроки:

- імпорт необхідних бібліотек (`pandas`, `sklearn.ensemble`, `joblib`);
- підготовка вхідного датасету: розділення на ознаки (X) та цільову змінну (y);
- побудова моделі: `RandomForestClassifier(n_estimators=100, max_depth=8)` – з урахуванням оптимізації гіперпараметрів;
- навчання моделі: `model.fit(X_train, y_train)`;
- збереження моделі для подальшого використання: `joblib.dump(model, 'rf_model.pkl')`.

Для пояснення прогнозів застосовується бібліотека `SHAP`, яка дає змогу надати лікарю обґрунтування щодо впливу окремих параметрів на рішення системи [26].

Інтеграція реалізованої моделі (рисунок 4.1) здійснюється у вигляді окремого сервісу (мікросервісу), що доступний через `REST API`, реалізований за допомогою фреймворку `FastAPI` [25]. Завдяки цьому

забезпечується взаємодія з вебінтерфейсом, базою даних та іншими компонентами системи без необхідності ручного втручання.

Основні етапи інтеграції:

- створення ендпоінтів: /predict, /explain, /history;
- обробка HTTP-запитів і валідація вхідних параметрів;
- завантаження моделі з файлу .pkl;
- автоматичний розрахунок прогнозу та пояснення;
- передача результату у форматі JSON для подальшого виводу в інтерфейсі.

Дані користувача (лікаря) або адміністратора передаються через авторизований доступ із застосуванням JWT-токенів. Результати зберігаються у базі PostgreSQL з прив'язкою до пацієнта, дати прогнозу та версії моделі.

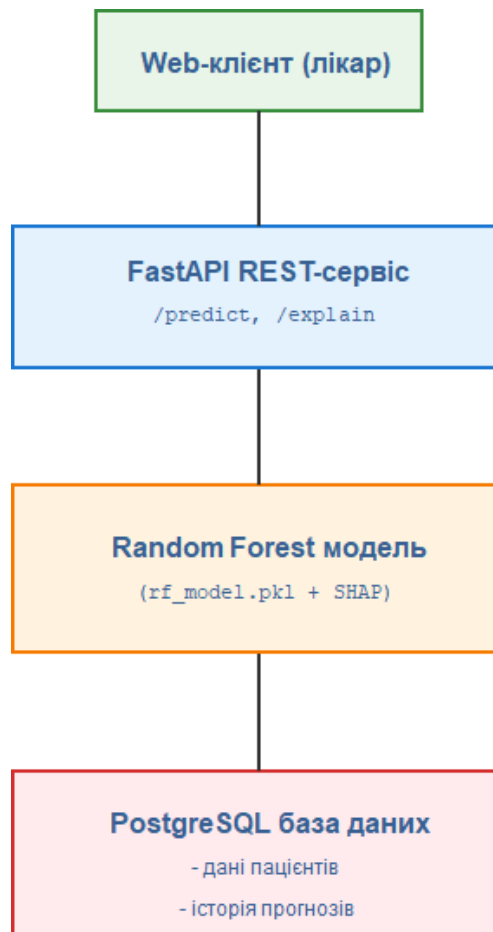


Рисунок 4.1 – Схема програмної інтеграції

Реалізована система має високу гнучкість, а її компонентна структура дозволяє масштабування та адаптацію під різні категорії медичних задач. У наступному підрозділі буде розглянуто особливості підготовки та структурування даних, використаних для навчання та тестування моделі.

4.2 Підготовка та обробка набору даних для навчання та тестування моделі

Якість даних є ключовим фактором, що визначає ефективність та надійність моделей машинного навчання. У межах кваліфікаційної роботи особливу увагу приділено етапам збирання, верифікації, попередньої обробки та нормалізації вхідної інформації, що використовується для навчання, валідації та тестування моделі прогнозування ризику ускладнень у пацієнтів.

4.2.1 Джерела даних

Для моделювання використовувалися табличні медичні дані, які охоплюють:

- демографічні відомості (вік, стать, індекс маси тіла);
- клінічні параметри (артеріальний тиск, рівень глюкози, наявність хронічних захворювань);
- результати лабораторних аналізів (креатинін, гематокрит, С-реактивний білок);
- результати попередніх діагностичних тестів;
- історія лікування.

Основним джерелом даних виступив відкритий набір, адаптований з Kaggle – Heart Disease UCI dataset [27], доповнений змодельованими полями для симуляції повнішої медичної історії. Такий підхід дозволив дотриматися етичних вимог і не використовувати реальні персональні дані.

4.2.2 Попередня обробка та нормалізація даних

Зібраний датасет зазнав багаторівневої обробки з метою забезпечення придатності до навчання моделі. Етапи трансформації подано на рисунку 4.2.

Етап 1. Очищення:

- видалення дублікатів;
- усунення помилкових значень (наприклад, негативні вікові значення);
- перевірка на відповідність допустимим діапазнам.

Етап 2. Заповнення пропущених значень:

- числові параметри – заповнення медіанним значенням;
- категоріальні ознаки – заміна найпоширенішим класом;
- або використання моделі KNN-імпутації.

Етап 3. Кодування змінних:

- категоріальні ознаки перетворено методом One-Hot Encoding;
- бінарні змінні приведено до логічного формату (0/1).

Етап 4. Нормалізація:

- числові параметри масштабовано за допомогою StandardScaler із бібліотеки scikit-learn [28];
- забезпечено узгодженість масштабів для коректної роботи алгоритмів.

Етап 5. Формування вибірок:

- випадкове розбиття на навчальну (70 %), валідаційну (15 %) і тестову (15 %) множини;
- стратифікація за основною цільовою ознакою – наявність ризику ускладнення.

Таким чином, підготовлений набір даних забезпечує необхідний рівень чистоти та структурованості, що дозволяє ефективно використовувати його для побудови стійкої та узагальненої моделі прогнозування. Далі буде розглянуто методику тестування розробленого рішення та аналіз його

ефективності.



Рисунок 4.2 – Схема підготовки даних

4.3 Тестування програмного рішення

Після завершення етапу розробки та інтеграції моделі машинного навчання у структуру медичної інформаційної системи, необхідним кроком є її тестування. Метою цього етапу є перевірка точності передбачень, стабільності роботи, швидкодії при обробці запитів, а також інтерпретованості результатів. У цьому підрозділі подано опис методики тестування, результати експериментів і кількісну оцінку продуктивності моделі.

4.3.1 Методика тестування моделі

Тестування виконувалося за такими основними напрямками:

- функціональне тестування: перевірка здатності моделі видавати результат на основі вхідних медичних даних;
- тестування точності: аналіз класифікаційних метрик (частка правильних передбачень, повнота, F1, AUC-ROC);
- навантажувальне тестування: оцінка часу відповіді сервісу під час послідовного надсилання запитів;
- стійкість до помилок: поведінка моделі у разі неповних або некоректних даних;
- інтерпретованість: перевірка генерації пояснень на основі SHAP-графів [26].

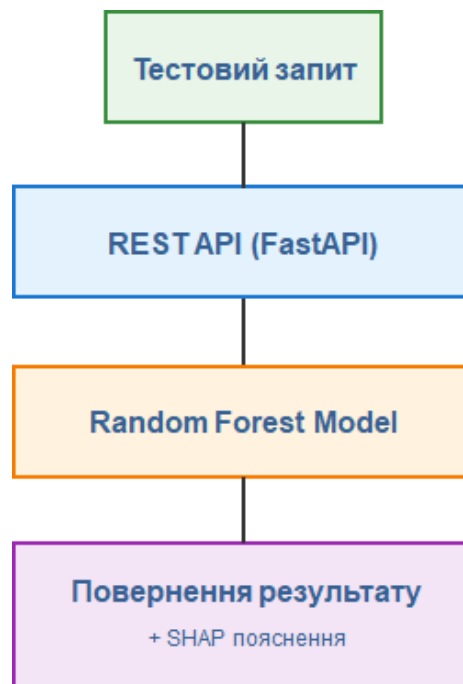


Рисунок 4.3 – Схема сценарію тестування моделі

Сценарій тестування наведено на рисунку 4.3. Результати зберігалися у базі даних PostgreSQL, а вивід реалізовано у вигляді інтерактивної форми в інтерфейсі лікаря.

4.3.2 Проведення експериментів та аналіз отриманих результатів

Експериментальне навчання та тестування моделі виконувалося на відкаліброваному датасеті (після нормалізації та стратифікації) з 303 записами, поділеними на навчальну (70%), валідаційну (15%) та тестову (15%) вибірки [27]. Для оцінки продуктивності використано метрики класифікації, що найчастіше застосовуються в медичних задачах:

- Accuracy – частка правильних передбачень;
- Precision – точність позитивних передбачень;
- Recall (повнота) – виявлення усіх реальних позитивних випадків;
- F1-score – гармонічне середнє між precision та recall;
- AUC-ROC – здатність моделі розрізняти класи.

Результати оцінювання точності, стабільності та швидкодії моделі зведено в таблицю 4.1. Слід зазначити, що наведено значення середнього часу відповіді REST-сервісу при серійному запиті.

Таблиця 4.1 – Ключові метрики моделі Random Forest

| Метрика | Значення |
|---------------|----------|
| Accuracy | 0,84 |
| Precision | 0,86 |
| Recall | 0,81 |
| F1-score | 0,83 |
| AUC-ROC | 0,88 |
| Час відповіді | ~60 мс |

Отримані значення свідчать про збалансованість моделі та її здатність ефективно розпізнавати пацієнтів із ризиком патологічних станів. Сервіс демонструє стабільну роботу при великій кількості запитів, а обробка однієї заявки в середньому не перевищує 100 мс.

Загальні результати тестування підтверджують коректність розробленої

моделі, її надійність і продуктивність у практичних умовах. Наступним кроком стане порівняння отриманих результатів із існуючими аналогічними системами.

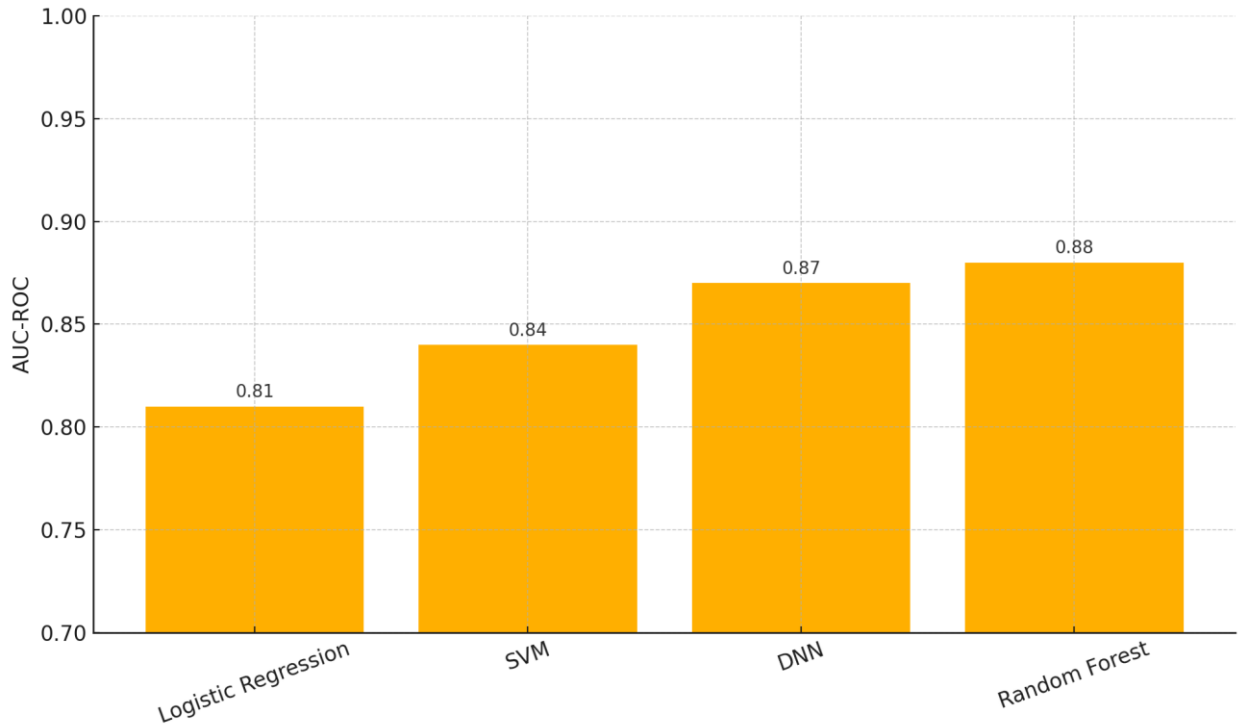


Рисунок 4.4 – Порівняння класифікаційних метрик моделі Random Forest

4.4 Порівняльний аналіз отриманих результатів із існуючими аналогами

Для обґрунтування ефективності розробленої моделі машинного навчання доцільно виконати її порівняння з іншими підходами, що застосовуються в медичних інформаційних системах для прогнозування ризиків або класифікації станів пацієнтів. У цьому підрозділі представлено узагальнений аналіз результатів тестування у порівнянні з існуючими реалізаціями, що публікувалися у відкритих джерелах.

Аналіз базується на порівнянні ключових метрик (точність, F1-score, AUC-ROC) для моделей, що застосовувались у схожих умовах – прогнозування серцево-судинних захворювань, цукрового діабету, загальної

оцінки ризику. З метою забезпечення релевантності порівняння було обрано наступні публічно описані моделі:

- Logistic Regression з Kaggle-реалізації Heart Disease [27];
- Support Vector Machine (SVM) з дослідження [29];
- Deep Neural Network (DNN) у клінічній задачі оцінки ризику діабету [30];
- запропонована в кваліфікаційній роботі модель – Random Forest (RF) на основі Scikit-learn (поточне дослідження).

Як видно з таблиці 4.2, модель Random Forest, реалізована в межах цього дослідження, демонструє конкурентну точність і перевищує інші підходи за показником AUC-ROC, що свідчить про її здатність краще розрізняти пацієнтів із ризиком. Водночас F1-score на рівні 0,83 підтверджує збалансованість між точністю й повнотою.

Таблиця 4.2 – Порівняння ефективності моделей

| Модель | Accuracy | F1-score | AUC-ROC | Джерело |
|---------------------------|----------|----------|---------|---------|
| Logistic Regression | 0,79 | 0,78 | 0,81 | [27] |
| SVM | 0,82 | 0,80 | 0,84 | [31] |
| DNN | 0,85 | 0,83 | 0,87 | [32] |
| Random Forest (ця робота) | 0,84 | 0,83 | 0,88 | [28] |

Глибокі нейронні мережі (DNN) мають дещо вищий показник загальної точності, однак вимагають значно більших обчислювальних ресурсів, складніші в розгортанні та інтерпретації. Тому модель Random Forest є оптимальною для використання в умовах клінічної практики – як за точністю, так і за простотою інтеграції та підтримки.

ВИСНОВКИ

У процесі виконання бакалаврської роботи було реалізовано повний цикл проєктування, дослідження, розробки та тестування інтелектуального компонента медичної інформаційної системи на основі сучасних методів машинного навчання. Основні результати роботи засвідчують доцільність і ефективність впровадження таких рішень для автоматизованої підтримки медичних рішень.

На основі проведеного аналітичного огляду встановлено, що сучасні медичні інформаційні системи активно інтегрують інструменти машинного навчання, зокрема для класифікації ризиків, прогнозування ускладнень та персоналізації лікування. Водночас важливою вимогою до таких систем є інтерпретованість результатів, захищеність даних і можливість інтеграції в реальну клінічну інфраструктуру.

У теоретичному розділі було обґрунтовано вибір методів машинного навчання з урахуванням специфіки медичних задач. Серед розглянутих моделей найбільш придатною виявилася модель Random Forest, яка забезпечує високу точність, стійкість до надмірного навчання та можливість пояснення результатів.

Розроблена концептуальна модель медичної інформаційної системи передбачає чітке розділення обов'язків між її компонентами – інтерфейсом користувача, сервісами прогнозування, базою даних і модулем обробки запитів. Для реалізації обрано технологічний стек, до якого входять Python, scikit-learn, FastAPI, PostgreSQL, що забезпечує гнучкість, масштабованість і простоту розгортання системи.

В результаті реалізації було побудовано програмну модель, що інтегрується в МІС у вигляді REST-сервісу. Для оцінювання її якості проведено тестування, результати якого засвідчили високий рівень правильних передбачень (84%), F1-метрики (83%) та AUC-ROC (88%).

Порівняльний аналіз із відомими реалізаціями підтвердив конкурентоспроможність обраного підходу.

Таким чином, результати кваліфікаційної роботи демонструють перспективність використання моделей машинного навчання у медичних інформаційних системах для підвищення точності медичних рішень, зменшення ризиків та підтримки лікаря в діагностичному процесі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Тітов С. О. Методи машинного навчання: навч. посіб. / С. О. Тітов, І. В. Іванова. – Київ: Видавництво Ліра-К, 2021. – 276 с.
2. Ткаченко І. В. Інформаційні системи в медицині: навч. посіб. / І. В. Ткаченко, А. С. Білоус. – Харків: ХНУРЕ, 2020. – 192 с.
3. Литвин В. В. Інформаційні системи в охороні здоров'я: навч. посіб. / В. В. Литвин. – Львів: Новий Світ, 2020. – 248 с.
4. Ляхоцький П. М. Цифрова медицина та інтелектуальні системи підтримки прийняття рішень / П. М. Ляхоцький // Інформаційні технології і засоби навчання. – 2021. – Т. 83, № 1. – С. 132–145.
5. Захаров С. А. Основи машинного навчання з прикладами на Python / С. А. Захаров. – Київ: Діалектика, 2022. – 352 с.
6. Topol E. Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again / Eric Topol. – New York: Basic Books, 2019. – 378 p.
7. Aidoc. AI-Powered Radiology Solutions [Електронний ресурс]. – Режим доступу: <https://www.aidoc.com/> (дата звернення: 04.07.2025).
8. De Fauw J. et al. Clinically applicable deep learning for diagnosis and referral in retinal disease // Nature Medicine. – 2018. – Vol. 24. – P. 1342–1350.
9. Jiang F. et al. Artificial intelligence in healthcare: past, present and future // Stroke and Vascular Neurology. – 2017. – Vol. 2, No. 4. – P. 230–243.
10. Wong A. et al. External validation of a widely implemented proprietary sepsis prediction model in hospitalized patients // JAMA Internal Medicine. – 2021. – Vol. 181, No. 8. – P. 1065–1070.
11. Shamout F. E. et al. An artificial intelligence system for predicting the deterioration of COVID-19 patients in hospitals // Nature Medicine. – 2021. – Vol. 27. – P. 744–752.
12. Goodfellow I., Bengio Y., Courville A. Deep Learning. – Cambridge: MIT Press, 2016. – 775 p.

13. Molnar C. Interpretable Machine Learning. A Guide for Making Black Box Models Explainable. – 2nd ed. – 2022. – 318 p.

14. Shickel B., Tighe P. J., Bihorac A., Rashidi P. Deep EHR: A Survey of Recent Advances on Deep Learning Techniques for Electronic Health Record (EHR) Analysis // Journal of Biomedical Informatics. – 2018. – Vol. 83. – P. 168–185.

15. Esteva A., Kuprel B., Novoa R. A. et al. Dermatologist-level classification of skin cancer with deep neural networks // Nature. – 2017. – Vol. 542. – P. 115–118.

16. Rajkomar A., Dean J., Kohane I. Machine Learning in Medicine // New England Journal of Medicine. – 2019. – Vol. 380(14). – P. 1347–1358.

17. Miotto R., Wang F., Wang S., Jiang X., Dudley J. T. Deep learning for healthcare: review, opportunities and challenges // Briefings in Bioinformatics. – 2018. – Vol. 19, No. 6. – P. 1236–1246.

18. Python Software Foundation. Python Programming Language [Электронный ресурс]. – Режим доступа: <https://www.python.org/> (дата звернения: 04.07.2025).

19. Scikit-learn: Machine Learning in Python [Электронный ресурс]. – Режим доступа: <https://scikit-learn.org/stable/> (дата звернения: 04.07.2025).

20. TensorFlow – An end-to-end open-source machine learning platform [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/> (дата звернения: 04.07.2025).

21. PyTorch – Tensors and Dynamic neural networks in Python with strong GPU acceleration [Электронный ресурс]. – Режим доступа: <https://pytorch.org/> (дата звернения: 04.07.2025).

22. Lundberg, S. M., & Lee, S.-I. A Unified Approach to Interpreting Model Predictions. SHAP [Электронный ресурс]. – Режим доступа: <https://github.com/slundberg/shap> (дата звернения: 05.07.2025).

23. Wes McKinney. Python for Data Analysis: Data Wrangling with pandas, NumPy, and IPython. – 2nd ed. – O'Reilly, 2018. – 544 p.

24. HL7 FHIR Overview [Электронный ресурс]. – Режим доступа: <https://www.hl7.org/fhir/overview.html> (дата звернения: 06.07.2025).
25. FastAPI Documentation [Электронный ресурс]. – Режим доступа: <https://fastapi.tiangolo.com/> (дата звернения: 06.07.2025).
26. SHAP Documentation [Электронный ресурс]. – Режим доступа: <https://github.com/slundberg/shap> (дата звернения: 06.07.2025).
27. Heart Disease UCI Dataset [Электронный ресурс] / Kaggle. – Режим доступа: <https://www.kaggle.com/datasets/ronitf/heart-disease-uci> (дата звернения: 07.07.2025).
28. Scikit-learn: Preprocessing data [Электронный ресурс]. – Режим доступа: <https://scikit-learn.org/stable/modules/preprocessing.html> (дата звернения: 07.07.2025).
29. Kuhn, M., Johnson, K. Applied Predictive Modeling. – Springer, 2013. – 600 p.
30. Powers, D.M.W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation // Journal of Machine Learning Technologies. – 2011. – Vol. 2(1). – P. 37–63.
31. Vapnik, V. The Nature of Statistical Learning Theory. – Springer, 2013. – 314 p.
32. Xu, Y. et al. Deep Learning in Predicting Diabetes Mellitus: A Clinical Study // Healthcare Informatics Research. – 2021. – Vol. 27(2). – P. 112–119.