**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**Факультет інформаційних технологій**

Кафедра технологій управління

**ЗВІТ**

**ПРО ПРОХОДЖЕННЯ НАУКОВО-ДОСЛІДНОЇ ПРАКТИКИ**

Студента 2 курсу магістратури групи ІАВ-21

спеціальності 122 Комп’ютерні науки освітньої програми «Інформаційна

аналітика та впливи»

Лавриновича Владислава Валерійовича

(прізвище, ім’я, по батькові)

Термін практики з 03.04.2023р. по 14.05.2023р.

База практики ТОВ “Інститут інформаційних технологій «Інтелліас»”

Науковий керівник магістерської роботи

д.т.н., доцент. Хлевна Юлія Леонідівна

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (науковий ступінь, вчене звання керівника, ПІБ)

Керівник практики від кафедри

Мочалова Дарія Сергіївна

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(науковий ступінь, вчене звання керівника, ПІБ)

**Київ-2023**

**Завдання**

Аналіз предметної області та наукової літератури за темою дипломної роботи «Передбачення серцево-судинних хвороб технологіями машинного навчання». Підбір математичних моделей, методи та методики дослідження.

Визначення проблематики установи, де студент проходить практику.

**ІНДИВІДУАЛЬНИЙ ГРАФІК ПРОХОДЖЕННЯ**

**НАУКОВО-ДОСЛІДНОЇ ПРАКТИКИ**

студентом 2 курсу магістратури групи ІАВ-21

спеціальності 122 Комп’ютерні науки, освітньої програми

«Інформаційна аналітика та впливи»

|  |  |  |  |
| --- | --- | --- | --- |
| Завдання за планом | Термін виконання | Фактичне виконання | Підписи наукового керівника та керівника від кафедри |
| Розробка індивідуального графіку проходження практики.  Узгодження його з науковим керівником магістерської роботи та керівником практики від кафедри. | 03.04.2023 | 03.04.2023 |  |
| Визначення предмету та об’єкту дослідження науково-  дослідної практики та їх взаємозв’язок з темою магістерської  роботи | 03.04.2023 | 03.04.2023 |  |
| Ознайомлення з науковими напрямами роботи бази практики | 10.04.2023 | 10.04.2023 |  |
| Ознайомлення з іноземними та вітчизняними науково-інформаційними джерелами за спеціалізацією та темою наукового дослідження, формування постановки задачі  дослідження та розробка бібліографії | 18.04.2023 | 18.04.2023 |  |
| Визначення проблематики бази практики, за темою  магістерського дослідження та розробка програми її рішення | 01.05.2023 | 01.05.2023 |  |
| Формулювання компонент магістерського дослідження: вступ, літературний огляд, огляд проблематики діючих підприємств (бази практики) за темою дослідження (перший розділ), підбір математичних моделей, методів та методики дослідження з врахуванням особливостей діючих підприємств (бази практики) (другий розділ) та розробка програми (технології) її рішення (четвертий розділ) | 04.05.2023 | 04.05.2023 |  |
| Виконання індивідуального завдання з обраної проблеми  досліджень | 09.04.2023 | 09.04.2023 |  |
| Оформлення звіту з практики | 12.05.2023 | 12.05.2023 |  |

Узгоджено: дата \_\_\_\_\_\_\_\_\_\_\_\_\_

**Науковий керівник магістерської роботи**

**д.т.н., професор Хлевна Юлія Леонідівна**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

**Керівник практики від кафедри**

**Мочалова Дарія Сергіївна**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

Характеристика студента-практиканта з об’єкту практики

Студент 2 курсу денної форми навчання 122 спеціальності проходив навчально-дослідну практику в ТОВ «Інститут інформаційних технологій «Інтелліас» з 03.04.2023 по 14.05.2023.

За період проходження практики студентом були виконані зазначені в індивідуальному завданні види робіт, а саме дослідження предметної області та аналіз наукової літератури та досліджень за темою дипломної роботи “Передбачення серцево-судинних хвороб методами машинного навчання”. Роботи були виконані в повному об’ємі без зауважень зі сторони керівника.

Протягом виконання робіт Лавринович В. В. зарекомендував себе як грамотний, відповідальний та пунктуальний співробітник. В процесі роботи практикант проявив себе як кваліфікований спеціаліст, здатний аналізувати факти, збирати необхідну інформацію і на її основі приймати зважені рішення.

За час проходження практики виявив зацікавленість, активність, концентрацію на вирішенні завдань у відповідності з профілем професійної діяльності. При вирішенні складних завдань проявив самостійність і оперативність. Володіє організаторськими здібностями, користується авторитетом у колег і співробітників.

За результатами роботи під час проходження практики, виконання програми практики, індивідуального завдання, які відображені у звіті з практики студент Лавринович Владислав Валерійович заслуговує оцінки відмінно.

М.П. Дата "12" травня 2023р.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ПІП керівника практики від підприємства

**Зміст**

Вступ 7

Основна частина8

1. Напрями та характеристика наукової діяльності бази практики8
2. Літературний огляд наукових джерел9
3. Визначення проблематики бази практики за темою магістерського дослідження19
4. Підбір математичних моделей, методів та методики дослідження 21  
   4.1 Метод опорних векторів (SVM)21

4.2 Наївний Байєсівський алгоритм23

4.3 Дерево рішень24

4.4 Random Forest25

4.5 XGBoost27

4.6 Глибокі нейронні мережі28

Висновок 33

# **Вступ**

В наш час, з стрімким розвитком інформаційних технологій, все більша увага прикута до застосування інформаційних технологій та інтелектуального аналізу даних в передбаченні та лікуванні різних хвороб.

Індустрія охорони здоров’я є одним із найбільш значущих бенефіціарів Data Science. Завдяки науковцям з даних медична діагностика стає ефективнішою та доступнішою, лікування персоналізованим, а медичні дослідження більше орієнтованими на дані.

Медицина може використовувати алгоритми аналізу даних для запобігання поширеним захворюванням за допомогою передбачення захворювань на основі різних метрик. Прогнозні моделі використовують наявні дані, аналізують їх та інтерпретують, щоб встановити кореляції та забезпечити точні прогнози. Наука про дані також допомагає зрозуміти стан здоров’я людини та надає допомогу, щоб запобігти майбутнім ризикам.

Серцево-судинні захворювання (ССЗ) зараз є найбільшою проблемою в медичній галузі. Це одні з найбільш смертельних і хронічних захворювань, які призводять до найбільшої кількості смертей. За останніми статистичними даними Всесвітньої організації охорони здоров’я (ВООЗ), щорічно від серцево-судинних захворювань помирає 20,5 мільйона людей, тобто приблизно 31,5% усіх смертей у світі. Також підраховано, що кількість щорічних смертей зросте до 24,2 мільйонів до 2030 року. Близько 85% смертей від серцево-судинних захворювань пов'язані з серцевими нападами та інсультами.

Застосування аналізу даних в цій області дозволить вчасно діагностувати багато серцевих захворювань та підвищити рівень обізнаності серед громадян.

Мета роботи – аналіз наукових досліджень прогнозування серцево-судинних захворювань з використанням технологій машинного навчання для подальшого використання при написанні дипломної роботи**.**

# **Основна частина**

1. **Напрями та характеристика наукової діяльності бази практики**

Компанія Інтелліас може бути використана як база практики для розробки систем прогнозування серцево-судинних хвороб за допомогою методів машинного навчання. Серцево-судинні хвороби - це одна з провідних причин смертності у світі, і раннє виявлення ризикових факторів та вчасне лікування можуть допомогти у запобіганні небажаних наслідків.

Інтелліас має досвід у розробці систем з прогнозування та класифікації на основі методів машинного навчання, таких як нейронні мережі, алгоритми класифікації та регресії, ансамблі дерев рішень та багатокласова класифікація. Для прогнозування серцево-судинних хвороб можуть бути використані дані, зібрані з різних джерел, таких як здоров'я пацієнта, медичні історії, результати тестів та обстежень, які можна обробляти та аналізувати з допомогою методів машинного навчання.

Інтелліас може використовувати свій досвід у розробці програмного забезпечення та науці даних для створення систем, які допоможуть у попередженні серцево-судинних хвороб та покращенні результатів лікування. Використання методів машинного навчання дозволить побудувати прогностичні моделі, які допоможуть у визначенні ризику розвитку серцево-судинних хвороб у конкретного пацієнта, а також розробити індивідуальні плани лікування.

За допомогою машинного навчання можна створити алгоритми, які допоможуть у визначенні ризикових факторів розвитку серцево-судинних хвороб та розробки індивідуальних планів лікування.

Для розробки таких систем необхідно мати велику кількість даних, зокрема медичних даних пацієнтів, що потребують аналізу. До цих даних можуть належати медичні історії, результати тестів та обстежень, а також біометричні дані, такі як аналіз крові, ЕКГ та інші показники здоров'я.

Компанія Інтеліас має досвід у розробці систем машинного навчання та інтелектуального аналізу даних, що може бути корисним у розробці систем прогнозування серцево-судинних хвороб. Вона використовує різні алгоритми машинного навчання, такі як нейронні мережі, ансамблі дерев рішень та інші, для розв'язання задач класифікації, прогнозування та кластеризації.

Такі системи можуть допомогти лікарям визначати ризик розвитку серцево-судинних хвороб у конкретного пацієнта та розробляти індивідуальні плани лікування, що в свою чергу покращує результати лікування та знижує ризики для здоров'я пацієнта.

1. **Літературний огляд наукових джерел**

Останніми роками індустрія охорони здоров’я пережила значний прогрес у сфері аналізу даних і машинного навчання. Ці методи були широко впроваджені та продемонстрували ефективність у різних сферах охорони здоров’я, зокрема в галузі медичної кардіології. Швидке накопичення медичних даних надає дослідникам безпрецедентну можливість розробити та перевірити нові алгоритми в цій галузі. Захворювання серця залишаються основною причиною смертності в країнах, що розвиваються, і виявлення факторів ризику та ранніх ознак захворювання стало важливою областю досліджень. Використання методів інтелектуального аналізу даних і машинного навчання в цій галузі потенційно може допомогти в ранньому виявленні та профілактиці захворювань серця.

Aadar Pandita, Sarita Yadav зробили дослідження, під назвою “Prediction of Heart Disease using Machine Learning Algorithms”. В цьому досліженні вони використовували UCI датасет, та кілька алгоритмів машинного навчання таких як Naive Bayes, K-Nearest Neighbor, Decision Tree і Random Forest, які взаємодіють, щоб знайти найточнішу модель. Етап попередньої обробки даних вони поділили на чотири підетапи.

Очищення: дані, які вони хочуть обробити, не будуть чистими, тобто вони можуть містити шум або значення, яких не вистачає під час обробки. Вони не можуть отримати хороші результати, тому для отримання хороших і ідеальних результатів їм потрібно все це усунути. Процес усунення всього цього є очищення даних. Вони заповнять відсутні значення та можуть усунути шум за допомогою деяких методів, як-от заповнення найбільш поширеним значенням у пропущеному місці.

Трансформація: це передбачає зміну формату даних одної форми в іншу, що робить їх найбільш зрозумілими за допомогою нормалізації, згладжування та узагальнення, агрегації

Інтеграція: Дані, які їм не потрібно обробляти, можуть походити не з одного джерела, іноді вони можуть надходити з різних джерел, вони їх інтегрують. Під час обробки це може бути проблемою, тому інтеграція є одним із важливих етапів попередньої обробки. Тут розглядаються різні питання для інтеграції.

Зменшення: коли вони працюють над даними, іноді розмірність може бути високою, а дані складними і важко зрозуміти, тому, щоб зробити їх зрозумілими системі, вони зменшать їх до необхідного формату, щоб досягти гарних результатів.

В загальному запропонована дослідниками система зображена на Рис. 1.

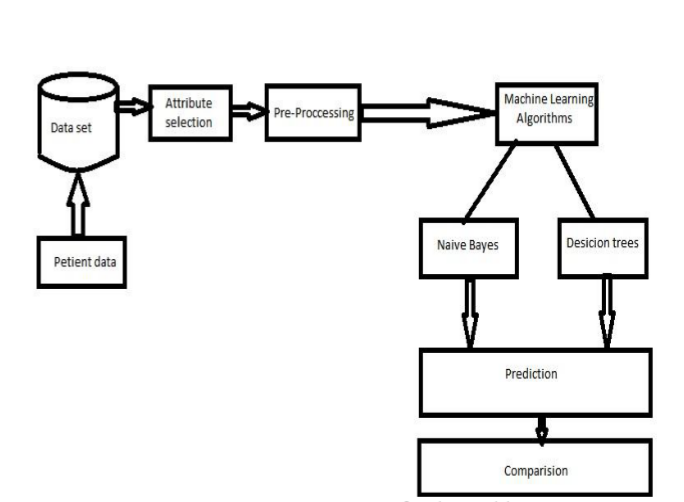


Рис. 1. Запропонована система для вирішення задачі

Після етапів відбору атрибутів та попередньої обробки, дослідники виділили наступні атрибути датасету, на основі яких в подальшому побудували моделі, наведені в Рис. 2.

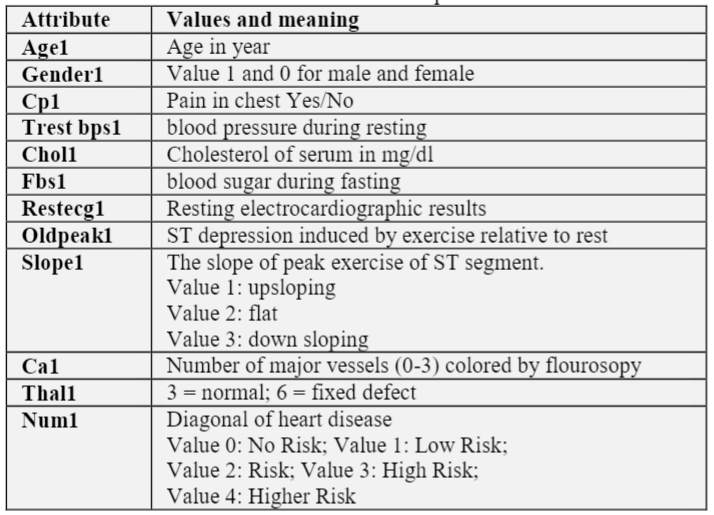


Рис. 2. Вихідні параметри датасету після попередньої обробки

За результатами роботи автори виявили, що поодинці алгоритми дають неточні результати, тому найкращим рішенням є комбінація алгоритмів k-means, ID3 або k-means і Naïve Bayes.

Науковці Saleh Mahdi Muhammed, Ghassan Abdul-Majeed, Mahmoud Shuker Mahmoud зробили дослідження під назвою «Prediction oh heart diseases by using Supervised Machine Learning». У цьому дослідженні вони обговорюють кілька методологій, які мають передбачити наслідки серцевих захворювань за допомогою методів машинного навчання. Були застосовані наступні алгоритми: ANN,Naive Bayes, DT, Random Forest, і Gradient Boosting. Однак точність, отримана в кожному дослідженні, наразі не вважається задовільною, оскільки певні алгоритми працюють краще, ніж інші.

У цьому дослідженні було визначено п’ять алгоритмів, які показали задовільну точність через 10-fold кросс-валідацію, демонструючи їхній потенціал для використання в прогнозуванні. Таким чином, мета дослідження полягала в тому, щоб виявити класифікатори, які можуть ефективно передбачати результати серцево-судинних захворювань і можуть мати практичне значення. За результатами дослідження автори запропонували наступну методологію, зображену на рис. 3.

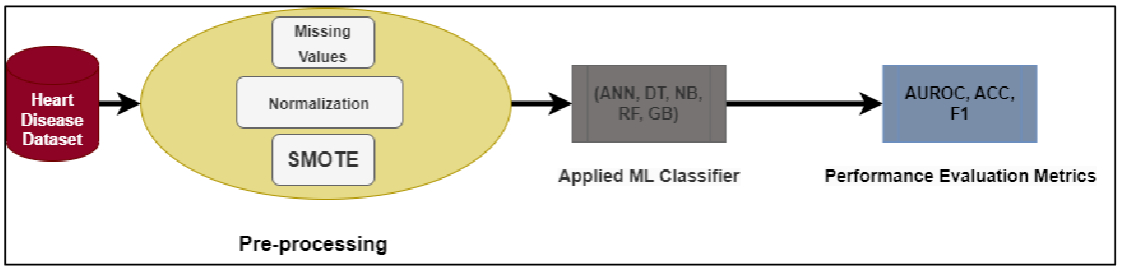


Рис. 3. Методологія запропонована дослідниками

Confusion matrix, створена для кожного алгоритму, включала чотири параметри: істинно позитивний (TP), істинно негативний (TN), хибно позитивний (FP) і хибно негативний (FN). Для оцінки алгоритмів використовувалися чутливість, специфічність і точність. Чутливість відноситься до частки фактичних позитивних результатів, які були правильно визначені класифікатором, тоді як специфічність вказує на здатність класифікатора правильно ідентифікувати негативні результати.

Точність — це відсоток правильно класифікованих екземплярів класифікатором, і це середня наближеність до цілі. Для порівняння продуктивності різних алгоритмів використовувалися різні статистичні вимірювання, такі як точність, відкликання та F-міра. Точність вимірює відсоток очікуваних позитивних результатів, які є фактичними позитивними, запам’ятовування представляє частку позитивних результатів, які правильно класифіковані, а F-вимірювання врівноважує точність і запам’ятовування для класифікатора. AUROC — це метрика продуктивності для розрізнення, яка оцінює здатність моделі розрізняти хворобливі випадки та здорові.

Щоб оцінити модель у цьому дослідженні, як для навчання, так і для тестування використовувався підхід K-fold cross-validation. Ця техніка передбачає поділ набору даних на K груп, або «згорток», де K означає кількість груп. K-fold cross-validation – це метод оцінки машинного навчання, у якому модель навчається на (K-1) групах, тоді як решта групи використовується для оцінки навченої моделі. У цьому методі модель навчається K разів, при цьому кожне згортання використовується для оцінки моделі. У цьому дослідженні була використана техніка 10-кратної перехресної перевірки, щоб уникнути переобладнання в прогнозній моделі.

Для визначення найефективнішого алгоритму класифікації до набору даних було застосовано п’ять різних алгоритмів, а саме ANN, RF, DT, GB і NB, а їх точність та інші статистичні змінні порівнювали за допомогою техніки 10-кратної перехресної перевірки. Алгоритми були оцінені на основі показників продуктивності. Щоб визначити чутливість, специфічність і точність результатів кожного алгоритму, була створена матриця помилок. Для обчислення цих показників були використані відповідні рівняння:

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦 = (𝑇𝑃+𝑇𝑁)/(𝑇𝑃+𝐹𝑃+𝑇𝑁+𝐹𝑁)

𝑃𝑟𝑒𝑐𝑖𝑠𝑖𝑜𝑛 = 𝑇𝑃/(𝑇𝑃+𝐹𝑃)

𝑅𝑒𝑐𝑎𝑙𝑙 = 𝑇𝑃(𝑇𝑃+𝐹𝑁)

𝐹1=2\*(precision\*recall)/ (precision + recall)

AUROC: 𝑇𝑃/(𝑇𝑃+𝐹𝑁)

Продуктивність кожного алгоритму оцінювалася за допомогою різних показників перехресної перевірки, і було визначено найефективніший алгоритм. Весь процес показаний на рис. 1, а рис. 4 відображає результати продуктивності використаних класифікаторів. Використані показники оцінки включають AUROC, точність, F1, точність і запам'ятовування, які були вибрані для забезпечення всебічної оцінки продуктивності кожного класифікатора та забезпечення справедливого порівняння між ними. Результати дослідження показують, що ANN забезпечує максимальну точність, чутливість і специфічність, а потім йдуть Gradient Boosting, Decision Tree, Random Forest, і Naive Bayes. Це дослідження виявило кілька класифікаторів ML з потенціалом для точного виявлення захворювань серця, що може бути цінним для лікарів, які прагнуть передбачити виникнення захворювань серця. у своїх пацієнтів.

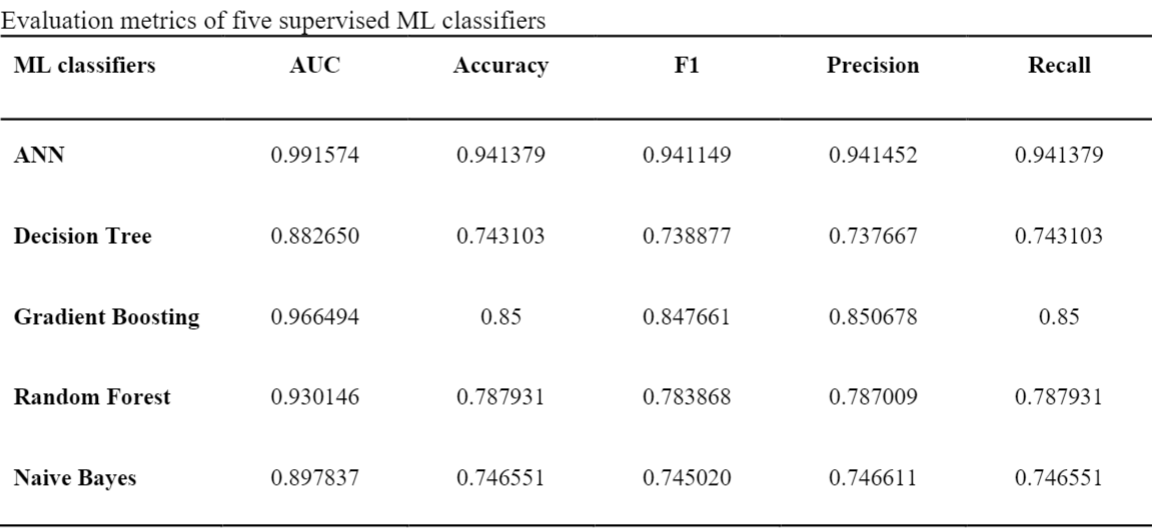


Рис. 4. Результати досліджуваних алгоритмів

Однак слід зазначити, що набір даних, використаний у цьому дослідженні (Cleveland), містив обмежені дані про захворювання серця, і для створення більш надійної моделі прогнозування потрібні додаткові дані та аналіз. Незважаючи на це обмеження, ми очікуємо, що майбутні дослідження покращать наше розуміння переваг і обмежень цього підходу, а використання алгоритмів машинного навчання для аналізу додаткових даних призведе до високоточних прогнозів серцево-судинних захворювань і пов’язаних із ними станів.

Md. Julker Nayeem, Sohel Rana, і Md. Rabiul Islam у своєму дослідженні під назвою «Prediction of Heart Disease Using MachineLearning Algorithms» використали декілька алгоритмів машинного авчання і досягли точності в 95%.

У їх дослідницькій роботі були використані різні типи алгоритмів керованого машинного навчання, щоб передбачити наявність захворювань серця у пацієнтів. Вони також зосередилися на ефективному способі покращення продуктивності своїх прикладних класифікаторів. Для обробки нульових значень у їхньому наборі даних була використана техніка середнього значення. Параметри, які не є необхідними, видалялися за допомогою техніки вибору параметрів з посиленням інформації. Щоб обчислити точність передбачення, вони застосовували K-найближчих сусідів (KNN), Naive Bayes і Random Forest до свого набору даних про хвороби серця. Вони розраховували точність, прецизійність, запам’ятовування, F1-оцінку та ROC, щоб порівняти ефективність своїх моделей класифікації. За допомогою техніки обробки нульових значень та вибору ознак з посиленням інформації вони змогли підвищити точність своїх моделей прогнозування. Зокрема, Random Forest дав найкращу точність класифікації - 95,63%, з точністю, запам’ятовуванням, оцінкою F1 та ROC 0,93, 0,92, 0,92 та 0,9 відповідно. Вони використовували методологію, зображену на рис. 5.

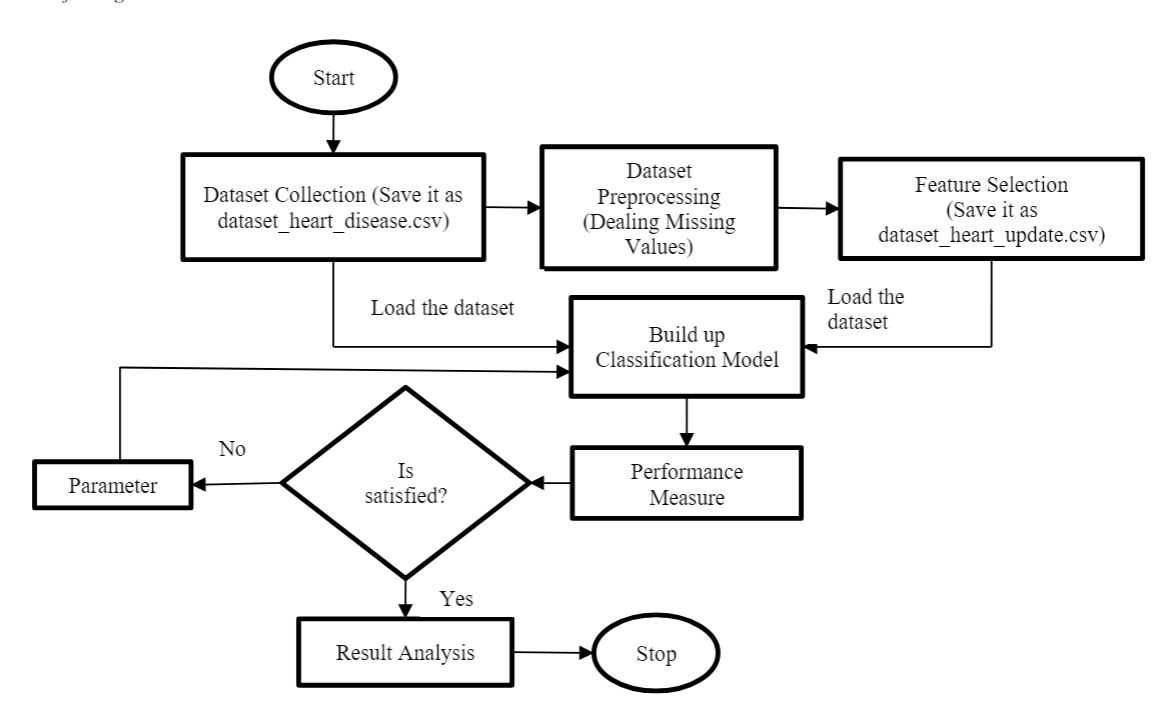


Рис. 5. Алгоритм проведення дослідження.

В загальному вони виділили наступні сім кроків виконання роботи:

Крок 1. Спочатку необхідно створити файл з назвою набору даних серцевих захворювань після збору набору даних із репозиторію машинного навчання UCI. Формат файлу - CSV (значення, розділені комами).

Крок 2: Наступним кроком є перевірка наявності нульових значень у кожному стовпці.

Крок 3: Якщо виявляються нульові значення, вони обробляються за допомогою техніки середньої вказівки.

Крок 4: Потім створюється новий файл CSV з назвою "heart update" після перевірки висококорельованих функцій за допомогою підходу до вибору функції отримання інформації для виявлення висококорельованих функцій.

Крок 5: Для визначення наявності серцевих захворювань завантажується набір даних "heart.csv" (містить усі атрибути та нульові значення) і класифікується за допомогою трьох класифікаторів.

Крок 6: Для визначення наявності серцевих захворювань завантажується набір даних "heart update.csv" (який містить лише сильно корельовані ознаки та виключає спостереження з нульовими значеннями) і класифікується за допомогою трьох класифікаторів.

Крок 7: Насамкінець, продуктивність моделі класифікації, отриманої на кроках 5 і 6, порівнюється за допомогою наборів даних "heart.csv" та "heart update.csv" відповідно.

Крок 8: Для порівняння точності моделі класифікації з результатами попередніх досліджень.

За результатами досліджень отримали результати, відображені на рис. 6.

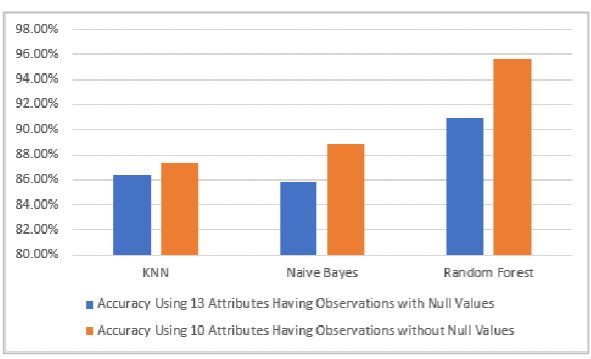


Рис. 6. Результати проведеного дослідження.

Arooj S, Rehman Su, Imran A, Almuhaimeed A, Alzahrani AK, Alzahrani A. у науковій публікації «A Deep Convolutional Neural Network for the Early Detection of Heart Disease» випробували використання згорткових нейронних мереж для передбачення серцево-судинних хвороб. Завдяки різноманітним технологіям і методам, розробленим для виявлення серцевих захворювань, використання класифікації зображень може ще більше покращити результати. Класифікація зображень є важливою проблемою в наш час. Це одна з найпростіших робіт з ідентифікації візерунків і комп’ютерного зору, яка стосується призначення однієї або кількох міток зображенням. Ідентифікація шаблонів із зображень стала легшою завдяки використанню машинного навчання, а глибоке навчання зробило його більш точним, ніж традиційні методи класифікації зображень. Це дослідження спрямоване на використання підходу глибокого навчання з використанням класифікації зображень для виявлення захворювань серця.

Глибока згорточна нейронна мережа (DCNN) наразі є найпопулярнішим методом класифікації для розпізнавання зображень. Запропонована модель оцінюється на загальнодоступному наборі даних про захворювання серця UCI, що включає 1050 пацієнтів і 14 атрибутів. Зібравши набір ознак, які можна отримати безпосередньо з набору даних про хвороби серця, автори вважають цей вектор ознак вхідним для DCNN, щоб визначити, чи належить примірник до класу здорових чи серцевих захворювань. Для оцінки ефективності запропонованого методу використовувалися різні показники ефективності, запам’ятовування та показник F1, і модель досягла точності перевірки 91,7%. Результати експерименті свідчать про ефективність запропонованого підходу в реальних умовах.

Архітектура запропонованої моделі DCNN представляє собою мережу прямого зв'язку з послідовною моделлю, в якій кожен рівень з'єднаний з одним входом і одним виходом. Атрибут класифікації захворювань серця є бінарним атрибутом, який класифікується як «1» для пацієнтів з захворюванням серця та «0» для пацієнтів без захворювання серця. Модель мала 2 згорткових шари, за якими слідували 8 щільних шарів. 14 вибраних атрибутів були об’єднані в повністю пов'язаний щільний шар. Загалом для побудови каркасу CNN було використано 8 повністю з'єднаних щільних шарів. Перші чотири шари містили 128 нейронів, наступні - 64, а останній - 1 нейрон. Перед нелінійним перетворенням ці шари нормалізують змінні. Експоненціальна лінійна одиниця (ELU) використовувалася як функція активації, за винятком останнього шару. Сигмоїдна функція була використана як функція активації в останньому шарі моделі CNN. Оптимізатор Nadam використовувався зі швидкістю навчання 0,001. Для функції втрат було встановлено двійкову крос-ентропію. Під час фази навчання кількість епох було встановлено на 100 для кращої класифікації. Рівень випадання становив 3%, щоб уникнути перенавчанням. Графіки тренування моделі та її точність зображені на рис. 7.

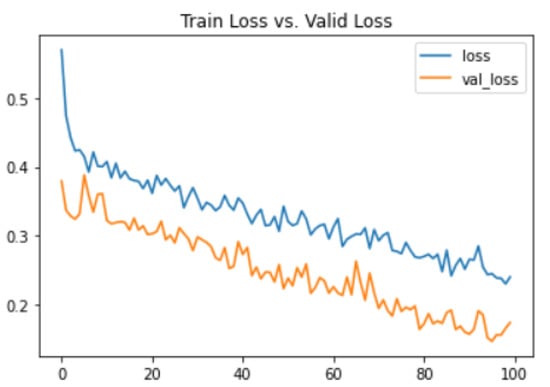
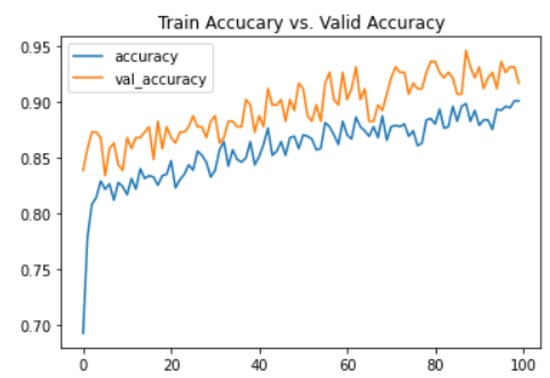
 

Рис. 7. Графіки навчання нейронної мережі.

Попередньо оброблені дані були використані з алгоритмом CNN для прогнозування захворювань серця на Google Collab. Запропоновану систему оцінювали щодо таких показників ефективності, як точність, точність, запам’ятовування та оцінка F1, і вона досягла 91,71%, 88,88%, 82,75% та 85,70% відповідно.

З проаналізованих досліджень ми можемо визначити, що найчастіше дослідники використовуються наступні алгоритми машинного навчання: Naive Bayes, Decision Trees, Random Forest, Gradient Boosting та нейронні мережі. Всі проаналізовані роботи використовували доволі застрілий датасет UCI та показали непогані показники точності. В подальшій роботі ми спробуємо застосувати ті ж самі алгоритми на іншому датасеті та побудувати свою нейронну мережу. Таким чином ми спробуємо покращити точність роботи алгоритмів для передбачення серцево-судинних хвороб.

Syed Nawaz Pasha, Dadi Ramesh, Sallauddin Mohmmad, A. Harshavardhan і Shabana написали наукову статтю під назвою «Cardiovascular disease prediction using deep learning techniques». В цій статті вони використовували датасет з Kaggle. Вони застосували 3 алгоритми машинного навчання, а саме SVM, KNN та Decision Tree, а також побудували нейронну мережу. За результатами порівняння точностей алгоритмів, зображених на рис. 8 можна сміливо стверджувати що нейронна мережа має найбільшу точність з усіх представлених алгоритмів в більшості випадків. При цьому SVM проявляє кращу точність на менших датасетах.

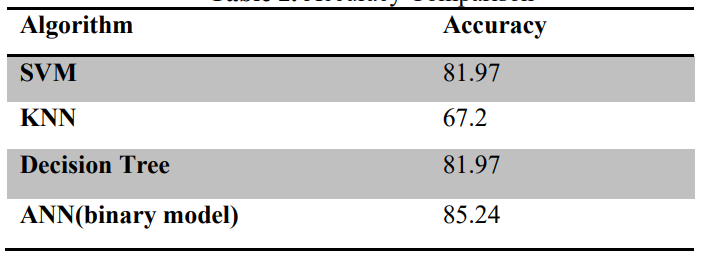


Рис. 8. Порівняння точності використаних моделей.

1. **Визначення проблематики бази практики за темою магістерського дослідження**

Проблематика компанії «Інтелліас» як бази практики відповідно до теми дослідження має багатогранний характер. З одного боку компанія могла б запропонувати своїм клієнтам медичної сфери автоматизовані системи засновані на алгоритмах штучного інтелекту, що покращило б репутацію компанії, збільшило клієнтську базу та прибуток загалом. Однак при цьому слід враховувати ряд наступних чинників, які можуть завадити реалізації таких медичних систем в компанії. Першим чинником є вузька спеціалізація, а отже – обмежений попит на системи такого класу. Недостатність точної та широкої бази даних щодо хвороб серця та судин. Також потреба у встановленні зв'язку між медичними дослідженнями та даними з машинного навчання для забезпечення максимальної ефективності прогнозування серцево-судинних хвороб.

Діагностика є важливою частиною циклу догляду за пацієнтом, оскільки вона визначає характер лікування, яке буде надано. Діагностика є першим кроком для правильного дослідження будь-якої хвороби, а також може визначити характер лікування, яке має бути надано. Навіть у епоху пікових технологій методика діагностики 21 століття далека від досконалості.

Помилки, спричинені лише в клінічній практиці в США, були причиною від 40 000 до 80 000 смертей у 2018 році. Наука про дані, та аналітики з методами глибокого навчання можуть поставити більш точний діагноз. Завдяки певним функціям аналіз даних може дозволити виявити ранні ознаки та дати підказки щодо профілактики завчасно.

Інструменти Data Science та Big Data показали позитивні результати в прогнозуванні, профілактиці та розробці планів лікування. Мобільні пристрої, розумні пристрої, датчики разом із Data Science можуть стати засобами попередження серцевого нападу. Наука про дані та хмарні обчислення зменшили відстань між лікарнями та світовими експертами з серцевих захворювань. Ці технології можуть бути використані для розробки системи охорони здоров’я на основі даних, де медсестри можуть допомогти надати першу допомогу, поки не будуть доступні лікарі-експерти. Наука про дані може зробити революцію в лікуванні серця і підвищити обізнаність щодо нездорового способу життя який може призвести до серцевих захворювань.

Отже, медична галузь потребує автоматизованої інтелектуальної системи для точного прогнозування захворювань серця. Це може бути досягнуто шляхом використання величезної кількості даних про пацієнтів, які є в медичній галузі, разом з алгоритмами машинного навчання. Останнім часом дослідницькі групи Data Science приділили велику увагу прогнозуванню захворювань. Це пояснюється швидким розвитком передових комп’ютерних технологій у сфері охорони здоров’я, а також доступністю масивних баз даних охорони здоров'я.

Поєднання глибокого навчання та розумної системи прийняття рішень мають великий потенціал для покращення медичної допомоги в нашому суспільстві. Дані є найціннішим ресурсом для отримання нових або додаткових знань і збирання важливої ​​інформації. Існує величезна кількість даних (великих даних) різних секторів, таких як наука, технології, сільське господарство, бізнес, освіта та охорона здоров'я.

Це повністю необроблені дані, у структурованій або неструктурованій формі. Наразі в секторі охорони здоров’я інформація, що стосується пацієнтів із медичними висновками, легко доступна в базах даних і з кожним днем ​​швидко зростає. Це сирі дані дуже надлишкові та незбалансовані. Потрібна попередня обробка для вилучення важливих метрик, що скорочують час виконання алгоритмів навчання та покращують класифікацію

Останні досягнення в області обчислювальних можливостей і програмування та можливості машинного навчання покращують ці процеси та відкривають двері для досліджень в секторі охорони здоров’я, особливо щодо раннього прогнозування захворювань, таких як ССЗ та рак, для підвищення рівня виживання.

Машинне навчання використовується в широкому діапазоні застосувань, від виявлення факторів ризику захворювань до розробки передових систем безпеки для автомобілів. Одним із потужних методів машинного навчання для передбачення є класифікація. Класифікація є ефективним методом машинного навчання під наглядом для виявлення захворювання при навчанні з використанням відповідних даних.

1. **Підбір математичних моделей, методів та методики дослідження**

**4.1 Метод опорних векторів (SVM)**

Мета машинного алгоритму опорних векторів — знайти гіперплощину в N-вимірному просторі (N — кількість ознак), яка чітко класифікує точки даних. Щоб розділити два класи точок даних, можна вибрати багато можливих гіперплощин. Наша мета — знайти площину, яка має максимальний запас, тобто максимальну відстань між точками даних обох класів. Збільшення межі відстані надає деяке посилення, щоб майбутні точки даних можна було класифікувати з більшою впевненістю. Загальна схематична візуалізація алгоритму наведена на рисунку 9.



Рисунок 9 – Схема роботи алгоритму опорних векторів

Як видно з рисунка – оптимальна гіперплощина – це та, яка має найбільшу відстань між обома ознаками класів. В нашому випадку у нас бінарна класифікація, тому ця ілюстрація також ілюструє необхідний нам результат.

Гіперплощини — це межі рішень, які допомагають класифікувати точки даних. Точки даних, що падають по обидва боки від гіперплощини, можна віднести до різних класів. Крім того, розмір гіперплощини залежить від кількості ознак. Якщо кількість вхідних об’єктів дорівнює 2, то гіперплощина – це просто лінія. Якщо кількість вхідних об’єктів дорівнює 3, то гіперплощина стає двовимірною площиною. Важко уявити, коли кількість функцій перевищує 3.

Опорні вектори — це точки даних, які знаходяться ближче до гіперплощини і впливають на положення та орієнтацію гіперплощини. Використовуючи ці опорні вектори, ми максимізуємо запас класифікатора. Видалення опорних векторів змінить положення гіперплощини. Це характеристики, які допомагають нам будувати нашу модель SVM. Лінійний класифікатор методу опорних векторів описується формулою:

Де – функція класифікатора для поділяючої гіперплощини, b – параметр зсуву (точка перетинання з віссю x), – вектор нормалі до поділяючої гіперплощини, – гіперплощина. Для задачі нелінійної класифікації використовується формула:

Де – функція класифікатора для поділяючої гіперплощини, b – параметр зсуву (точка перетинання з віссю x), – вектор нормалі до поділяючої гіперплощини, – гіперплощина, – мітка класу, – відповідний зв’язаний множник Лагранжа, - функція ядра, що відповідає скалярному добутку ознак в розширеному просторі.

**4.2 Наївний Байєсівський алгоритм**

Найпростіші рішення зазвичай є найпотужнішими, і Наївний Байєсівський алгоритм є хорошим прикладом цього. Незважаючи на досягнення в області машинного навчання за останні роки, застосування алгоритму виявилося не тільки простим, але й швидким, точним і надійним. Він успішно використовується для багатьох цілей, але він особливо добре працює з проблемами обробки природної мови (NLP).

Наївний Байєсівський алгоритм — це імовірнісний алгоритм машинного навчання, заснований на теоремі Байєса, який використовується в широкому спектрі завдань класифікації. Наївний класифікатор Байєса — це імовірнісний класифікатор, заснований на теоремі Байєса, яка передбачає, що кожна ознака робить незалежний і рівний внесок у цільовий клас. Класифікатор NB передбачає, що кожна ознака незалежна і не взаємодіє один з одним, так що кожна ознака незалежно і однаково впливає на ймовірність належності вибірки до певного класу. Класифікатор NB простий у реалізації та швидкий у обчислювальному відношенні та добре працює на великих наборах даних, які мають високу розмірність. Класифікатор NB сприятливий для додатків у реальному часі та не чутливий до шуму. Класифікатор NB обробляє навчальний набір даних для обчислення ймовірностей класів P(yi) та умовних ймовірностей, які визначають частоту кожного значення ознаки для заданого значення класу, поділену на частоту екземплярів із цим значенням класу. NB класифікатор найкраще працює, коли корельовані ознаки видаляються, оскільки корельовані ознаки будуть голосувати двічі в моделі, що призведе до надмірного підкреслення важливості корельованих ознак. Формули виражають імовірність для деякого класу.

Де P(c|x) - апостеріорна ймовірність класу, c - клас, x - атрибути, P(c) - попередня ймовірність класу, P(x|c) - ймовірність предиктора даного класу, P(x) - попередня ймовірність предиктора. Наївний алгоритм Байєса відомий своєю високою продуктивністю на великомасштабованих даних і іноді перевершує навіть більш складні методи.

**4.3 Дерево рішень**

У світі машинного навчання розробники можуть легко створювати незалежне середовище для проектів. Щоб досягти надійних результатів, потрібно лише кілька кліків, щоб встановити та підігнати моделі. Проте багато алгоритмів може бути досить важко зрозуміти, не кажучи вже про пояснення. Дерева рішень, незважаючи на погану продуктивність у своїй базовій формі, легко зрозуміти, i в складі комплексних алгоритмів (Random Forest, XGBoost) досягають чудових результатів.

Ідея досить проста і нагадує людський розум. Якби ми спробували розділити дані на частини, наші перші кроки були б засновані на запитаннях. Крок за кроком дані будуть розділятися на унікальні частини, і, нарешті, ми розділяли б вибірки. Це суть всієї концепції. Щоб краще пояснити це поняття, ми скористаємося популярною термінологією:

* Вузол: кожен об’єкт в дереві. Вузли містять підмножини даних, і за винятком листових вузлів питання розбиває підмножину.
* Батьківський вузол: Питання, яке розбиває дані.
* Дочірній вузол: результуючий вузол. Він також може бути батьком для своїх дітей.
* Листковий вузол: Останній вузол без додаткових запитань. Лише частина даних, що представляють відповіді на попередні запитання.
* Відділення: Унікальний рядок запитань з відповідями, які надходять до листкового вузла.
* Корінь: верхній вузол.

Можна використовувати дерева рішень з регресією або класифікацією. Методики дещо відрізняються, тому ми розглянемо обидві, починаючи з класифікації. Як правило, для класифікації модель слід вивчати на навчальних даних із заздалегідь визначеним набором міток. Він передбачав би мітку (тобто клас) для нових зразків. Отже, ми маємо набір даних з різними атрибутами (функціями). Кожен зразок має свою комбінацію значення ознак.

Листки представляють цільові класи, кожен вузол - тестовий набір для певного атрибута даних, а ребра є результатом тестового випадку.

Дерева рішень утворюють вкладені оператори if-else, і чим глибше дерево, тим краще модель. Коротка ілюстрація компонентів алгоритму показана на малюнку 10.

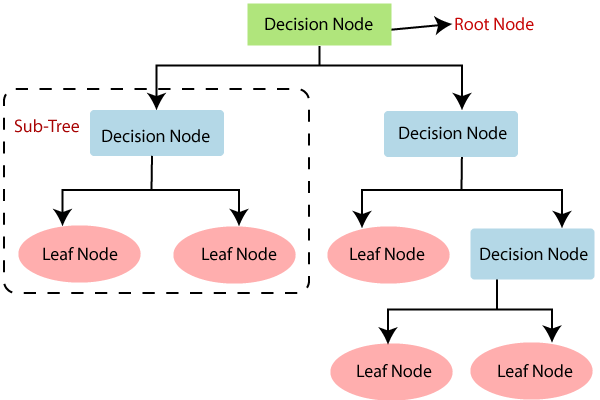


Рисунок 10 – Компоненти дерева рішень

**4.4 Random Forest**

Random Forest — це алгоритм навчання під наглядом. «Ліс», який він будує, являє собою комбінацію дерев рішень, зазвичай навчених методом «мішкування». Загальна ідея методу мішків полягає в тому, що комбінація моделей навчання підвищує загальний результат.

Однією з великих переваг Random Forest є те, що його можна використовувати як для задач класифікації, так і для задач регресії, які формують більшість сучасних систем машинного навчання. Random Forest має майже ті самі гіперпараметри, що й дерево рішень. Random Forest додає моделі додаткову випадковість під час зростання дерев. Замість того, щоб шукати найважливішу характеристику під час розбиття вузла, він шукає найкращу характеристику серед випадкової підмножини ознак. Це призводить до широкого розмаїття, що зазвичай призводить до кращої моделі. Отже, у Random Forest алгоритмом розбиття вузла до уваги враховується лише випадкова підмножина ознак. Ви навіть можете зробити дерева більш випадковими, додатково використовуючи випадкові пороги для кожної функції, а не пошук найкращих можливих порогових значень (як це робить звичайне дерево рішень).

Однією з найбільших переваг Random Forest є його універсальність. Його можна використовувати як для завдань регресії, так і для завдань класифікації, а також легко побачити відносну важливість, яку він надає вхідним функціям.

Random Forest також є дуже зручним алгоритмом, оскільки гіперпараметри за замовчуванням, які він використовує, часто дають хороший результат прогнозу. Зрозуміти гіперпараметри досить просто, і їх також не так багато.

Однією з найбільших проблем машинного навчання є перенавчання, але в більшості випадків цього не станеться завдяки класифікатору Random Forest. Якщо в лісі достатньо дерев, класифікатор не переповнить модель.

Основним обмеженням Random Forest є те, що велика кількість дерев може зробити алгоритм занадто повільним і неефективним для передбачення в реальному часі. Загалом, ці алгоритми швидко навчаються, але досить повільно створюють передбачення після навчання. Для більш точного прогнозу потрібно більше дерев, що призводить до повільнішої моделі. У більшості реальних додатків алгоритм Random Forest досить швидкий, але, безсумнівно, можуть виникнути ситуації, коли продуктивність під час виконання є важливою, а інші підходи були б кращими. Для знаходження ваги кожного дерева використовується наступна формула

Де – пріоритет дерева j, – зважена кількість екземплярів класу які доходять до цього вузла (дерева), – значення домішки Gini, – лівий дочірній вузол (дерево). Формула для значення домішки Gini наведена на формулі:

Де pj – ймовірність належності до деякого класу j.Звичайно, Random Forest є інструментом прогнозного моделювання, а не інструментом опису, тобто, якщо ви шукаєте опис взаємозв’язків у ваших даних, інші підходи були б кращими.

**4.5 XGBoost**

Спочатку XGBoost розпочався як дослідницький проект Tianqi Chen’a як частина групи Distributed (Deep) Machine Learning Community (DMLC). Спочатку він починався як термінальна програма, яку можна було налаштувати за допомогою файлу конфігурації libsvm. Він став добре відомим у колах змагань з машинного навчання після його використання в переможному рішенні Higgs Machine Learning Challenge. Незабаром після цього були створені пакети Python і R, і XGBoost тепер має реалізації пакетів для Java, Scala, Julia, Perl та інших мов. Це принесло бібліотеку більше розробників і сприяло її популярності серед спільноти Kaggle, де вона використовувалася для великої кількості конкурсів.

XGBoost працює як метод Ньютона-Рафсона у функціональному просторі, на відміну від градієнтного прискорення, яке працює як градієнтний спуск у функціональному просторі, у функції втрат використовується наближення Тейлора другого порядку для встановлення зв’язку з методом Ньютона-Рафсона.

При використанні градієнтного підвищення для регресії слабкі підмоделі є деревами регресії, і кожне дерево регресії відображає точку вхідних даних на один зі своїх листків, що містить безперервну оцінку. XGBoost мінімізує регуляризовану (L1 і L2) цільову функцію, яка поєднує опуклу функцію втрат (на основі різниці між прогнозованими та цільовими результатами) та штрафний термін за складність моделі (іншими словами, функції дерева регресії). Навчання відбувається ітераційно, додаючи нові дерева, які передбачають залишки або помилки попередніх дерев, які потім об’єднуються з попередніми деревами, щоб зробити остаточний прогноз. Це називається посиленням градієнта, оскільки він використовує алгоритм градієнтного спуску, щоб мінімізувати втрати при додаванні нових моделей. Коротку ілюстрацію того, як працює підсилення градієнтного дерева наведено на рисунку.

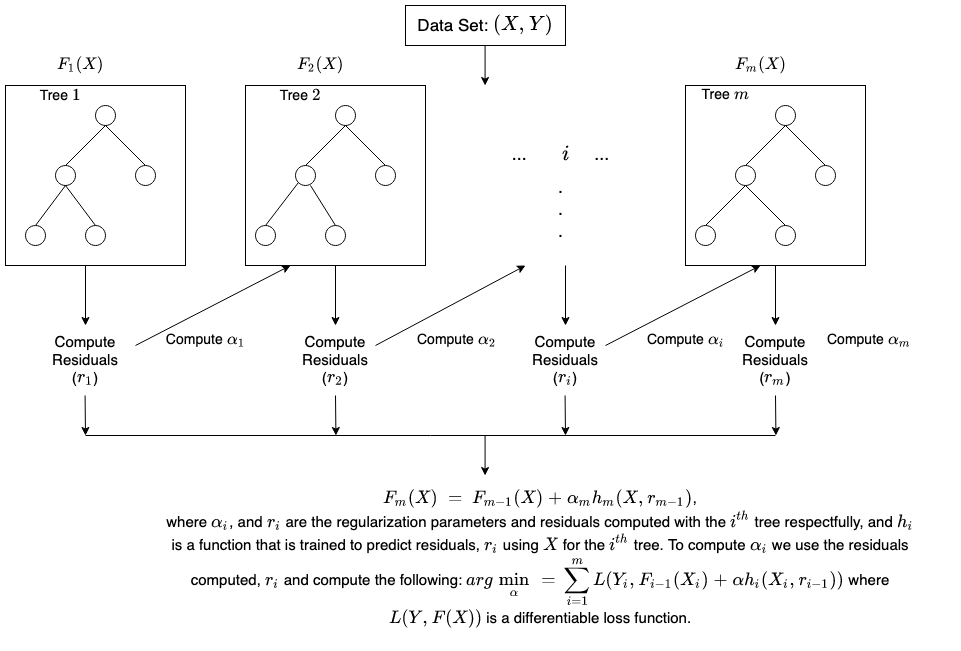


Рисунок 11 – Ілюстрація роботи XGBoost

**4.6 Глибокі нейронні мережі**

Нейронна мережа з шарів, а шар складається з невеликих окремих одиниць, які називаються нейронами. Нейрон в нейронній мережі можна краще зрозуміти за допомогою біологічних нейронів. Штучний нейрон схожий на біологічний нейрон. Він отримує вхідні дані від інших нейронів, виконує певну обробку та виробляє вихід. Математичний опис нейрона наведений на формулі :

Де – значення нейрона, – зміщення, – вага, або бета-коефіцієнт, .

Кожному зі своїх вхідних з’єднань вузол призначає значення, відоме як «вага». Коли мережа активна, вузол отримує інший елемент даних — інше значення — над кожним своїм з’єднанням і множить його на відповідну вагу. Потім отримані результати додаються разом, у результаті чого виходить єдине число. Якщо це число нижче порогового значення, вузол не передає дані наступному шару. Якщо число перевищує порогове значення, вузол «спрацьовує», що в сучасних нейронних мережах зазвичай означає надсилання числа — суми зважених вхідних даних — по всіх своїх вихідних з’єднаннях.

Коли нейронну мережу навчають, усі її ваги та пороги спочатку встановлюються на випадкові значення. Дані навчання надходять до нижнього шару — вхідного — і проходять через наступні шари, множачись і складаючись разом, доки не потрапляють, трансформовані, на вихідний шар. Під час тренування ваги та пороги постійно коригуються, доки дані навчання з однаковими мітками постійно дають подібні результати.

Функція активації в нейронній мережі визначає, як зважена сума вхідних даних перетворюється на вихід з вузла або вузлів у шарі мережі. Іноді функцію активації називають «функцією передачі». Якщо діапазон виходу функції активації обмежений, це можна назвати «функцією здавлення». Багато функцій активації є нелінійними, і їх можна назвати «нелінійністю» в шарі або конструкції мережі.

Вибір функції активації має великий вплив на можливості та продуктивність нейронної мережі, і різні функції активації можуть використовуватися в різних частинах моделі. Технічно функція активації використовується в межах або після внутрішньої обробки кожного вузла в мережі, хоча мережі призначені для використання однієї і тієї ж функції активації для всіх вузлів у шарі. Мережа може мати три типи шарів: вхідні шари, які отримують необроблені вхідні дані з домену, приховані шари, які беруть вхідні дані з іншого шару і передають вихід на інший шар, і вихідні шари, які роблять прогноз.

Усі приховані шари зазвичай використовують ту саму функцію активації. Вихідний рівень зазвичай використовує функцію активації, відмінну від прихованих шарів, і залежить від типу передбачення, якого вимагає модель.

Функції активації також зазвичай диференційовані, що означає, що похідну першого порядку можна обчислити для заданого вхідного значення. Це необхідно з огляду на те, що нейронні мережі зазвичай навчаються за допомогою алгоритму зворотного поширення помилки, який вимагає похідної від помилки передбачення для оновлення ваг моделі.

Існує багато різних типів функцій активації, які використовуються в нейронних мережах, хоча, можливо, лише невелика кількість функцій використовується на практиці для прихованих і вихідних шарів.

Функція випрямленої лінійної активації, або функція активації ReLU, є, мабуть, найпоширенішою функцією, яка використовується для прихованих шарів.

Вона є поширеною, оскільки проста у реалізації та ефективна у подоланні обмежень інших раніше популярних функцій активації, таких як Sigmoid і Tanh. Зокрема, вона менш чутлива до зникаючих градієнтів, які заважають навчанню глибоких моделей, хоча і може страждати від інших проблем, наприклад, насичених або «мертвих» одиниць.

Функція ReLU розраховується наступним чином: max(0.0, x). Це означає, що якщо вхідне значення (x) є негативним, то повертається значення 0.0, інакше повертається дане значення.

Сигмовидна функція активації також називається логістичною функцією.

Це та сама функція, що використовується в алгоритмі класифікації логістичної регресії. Функція приймає будь-яке реальне значення в якості вхідних даних і виводить значення в діапазоні від 0 до 1. Чим більше вхідний сигнал (більш позитивний), тим ближче вихідне значення буде до 1,0, тоді як чим менше вхід (більш негативний), тим ближче вихід буде 0,0. Сигмовидна функція активації розраховується наступним чином: 1,0 / (1,0 + e^-x), де e — математична константа, яка є основою натурального логарифма.

Функція активації гіперболічного тангенса також називається просто функцією Tanh (також «tanh» і «TanH»). Вона дуже схожа на функцію активації сигмоїда і навіть має таку ж S-подібну форму. Функція приймає будь-яке реальне значення в якості вхідних даних і виводить значення в діапазоні від -1 до 1. Чим більше вхідний сигнал (більш позитивний), тим ближче вихідне значення буде до 1,0, тоді як чим менше вхід (більш негативний), тим ближче вихід буде до -1,0.

Функція активації Tanh розраховується наступним чином:

(e^x – e^-x) / (e^x + e^-x), де e — математична константа, яка є основою натурального логарифма.

Нейронна мережа майже завжди матиме однакову функцію активації у всіх прихованих шарах. Дуже незвичайно змінювати функцію активації за допомогою мережевої моделі. Традиційно сигмовидна функція активації була функцією активації за замовчуванням у 1990-х роках. Можливо, з середини до кінця 1990-х до 2010-х років функція Tanh була функцією активації за замовчуванням для прихованих шарів.

Як сигмовидна, так і Tanh функції можуть зробити модель більш сприйнятливою до проблем під час навчання через так звану проблему зникаючих градієнтів. Функція активації, що використовується в прихованих шарах, зазвичай вибирається на основі типу архітектури нейронної мережі.

Сучасні моделі нейронних мереж із загальною архітектурою, такими як MLP і CNN, будуть використовувати функцію активації ReLU або розширення.

Рекурентні мережі досі зазвичай використовують функції активації Tanh або сигмовидної форми, або навіть обидві. Наприклад, LSTM зазвичай використовує сигмоїдну активацію для повторних з'єднань і активацію Tanh для виведення.

Вихідний шар — це шар моделі нейронної мережі, який безпосередньо виводить прогноз. Є три функції активації, які ви можна розглянути для використання на вихідному рівні.

Лінійна функція активації також називається «ідентичність» (помножена на 1,0) або «без активації». Це пов’язано з тим, що функція лінійної активації жодним чином не змінює зважену суму вхідних даних і натомість повертає значення безпосередньо.

Сигмовидна функція логістичної активації була описана вище.

Функція softmax виводить вектор значень, які досягають суми 1,0, які можна інтерпретувати як ймовірності приналежності до класу. Це пов’язано з функцією argmax, яка виводить 0 для всіх параметрів і 1 для вибраного параметра. Softmax — це «м’яка» версія argmax, яка дозволяє отримати схожий на ймовірність вихід функції «переможець отримує все». Таким чином, вхід до функції є вектором дійсних значень, а вихідним є вектор такої ж довжини зі значеннями, які сумують до 1,0, як ймовірності.

Функція softmax розраховується наступним чином: e^x / sum(e^x)

Де x — вектор виходів, а e — математична константа, яка є основою натурального логарифма.

Нейронна мережа використовувалась для передбачення серцево-судинних хвороб і досягла високої точності, але ми спробуємо розробити власну модель і підвищити точність.

**4.6 Методологія вирішення задачі**

Для початку нам необхідно ознайомитись з датасетом. Для цього з використанням pandas та бібліотек для малювання графіків ми зробимо візуалізації для кращого розуміння наших даних.

Після цього ми якщо необхідно, попередньо обробимо дані, вилучимо прогалини та замінимо назви класів чисельними значеннями. Збалансуємо датасет.

Далі розіб’ємо датасет на тренувальну та тестувальну частини, проведемо навчання і порівняємо результати за допомогою матриці невідповідностей.

Зробимо висновки та проаналізуємо можливі шляхи впровадження найефективнішої моделі. Реалізуємо систему із використанням моделі.

Весь процес зображено на рисунку 12.

Рисунок 12 – Методика проведення дослідження.

# **Висновок**

З огляду на наукові дослідження, прогнозування серцево-судинних хвороб методами машинного навчання є перспективним напрямком досліджень. Було проведено значну кількість робіт, які показали високу точність і надійність таких моделей прогнозування.

Методи машинного навчання, зокрема нейронні мережі, дозволяють враховувати велику кількість факторів, які можуть впливати на розвиток серцево-судинних хвороб. Такі фактори, як вік, стать, історія захворювань, стиль життя та інші, можуть бути враховані при створенні прогностичних моделей.

Дослідження також показали, що застосування машинного навчання може допомогти в попередженні серцево-судинних захворювань та зменшенні ризику їх розвитку. Аналіз даних, зібраних в рамках прогностичних моделей, може допомогти виявити певні закономірності та тенденції, що забезпечать можливість раннього діагностування хвороби.

Таким чином, можна стверджувати, що прогнозування серцево-судинних хвороб методами машинного навчання має великий потенціал у покращенні діагностики та лікування цих захворювань. Однак, необхідно продовжувати проводити дослідження та вдосконалювати методи, щоб забезпечити їх максимальну ефективність та точність.

Оглянуті дослідження показують що з усіх використаних моделей нейронні мережі показують найвищу точність. Також успіх показують комплексні моделі які складаються з декількох алгоритмів. В цілому застосування алгоритмів машинного навчання для прогнозування серцево-судинних хвороб вже довело свою успішність і може бути використано для лікування пацієнтів.