МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

СНІЖИНСЬКИЙ МАКСИМ ВАДИМОВИЧ

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій
канд. техн. наук, доцент

_____ О. В. Зелінська
«____»_____ 20__ р.

СИСТЕМА ОБРОБКИ ТА АНАЛІЗУ ДАНИХ У МЕДИЧНІЙ СФЕРІ НА ОСНОВІ ТЕХНОЛОГІЙ BIG DATA

Спеціальність 122 Комп’ютерні науки

Кваліфікаційна (магістерська) робота

Науковий керівник:
Січко Т.В., к.т.н., доцент
_____
(підпис)

Оцінка: _____ / _____ / _____
(бали/за шкалою ЄКТС/за національною шкалою)
Голова ЕК: _____
(підпис)

Вінниця — 2024

АНОТАЦІЯ

Сніжинський М.В. Розробка системи обробки та аналізу даних в медичній сфері на основі технологій Big Data. Спеціальність 122 «Комп’ютерні науки», Освітня програма «Комп’ютерні технології обробки даних (Data science)». Донецький національний університет імені Василя Стуса, Вінниця, 2024.

У кваліфікаційній (магістерській) роботі розроблено систему обробки та аналізу даних у медичній сфері за допомогою технологій Big Data. Описана актуальність даної роботи, проаналізовано існуючі аналоги. Показані кроки розробки, отримані результати, використані інструменти, підходи до розробки, перспективи подальшого розвитку даної системи.

Ключові слова: Big Data, Healthcare, обробка даних, аналіз даних, веб-додаток, система для аналізу.

ANNOTATION

Snizhynskyi M.V. Development of a system for processing and analyzing data in the medical field using Big Data technologies Specialty 122 "Computer Science", Educational program "Data science". Vasyl' Stus Donetsk National University, Vinnytsia, 2024.

In the qualification (master's) work, a system for processing and analyzing data in the medical field using Big Data technologies was developed. The relevance of this work is described and existing analogues are analyzed. The steps of development, the results obtained, the tools used, the approaches to development, and the prospects for further development of this system are shown.

Keywords: Big Data, Healthcare, data processing, data analysis, web application, system for analysis.

ЗМІСТ

ВСТУП

У медичній галузі спостерігається експоненціальне зростання обсягу та різноманітності даних, що генеруються з різних джерел, таких як: електронні медичні картки (ЕМК), медичні зображення, носимі пристрої, геномне секвенування та дані, створені пацієнтами. Серед цього потоку даних значна частина існує в неструктурованих форматах, що включають текстові нотатки, зображення, аудіозаписи та розповіді у вільній формі. Неструктуровані дані, на відміну від структурованих, організованих у заздалегідь визначені поля або бази даних, не мають заздалегідь визначеної моделі, що робить їх складними для інтерпретації, обробки та вилучення значущої інформації. У медичній сфері такі дані містять багато цінної інформації, зокрема: клінічні записи лікарів, звіти радіологів, звіти про патологію, рукописні рецепти та медичні зображення, такі як рентгенівські знімки, МРТ-сканування та КТ-сканування.

Серед основних труднощів із якими стикається сьогоднішня медицина в контексті ІТ-технологій можна виділити наступні:

- неоднорідність і складність даних. Неструктуровані медичні дані існують у різних форматах, починаючи від тексту у вільній формі в клінічних записах і закінчуючи складними медичними зображеннями та нестандартизованими даними з натільних пристроїв. Така неоднорідність ускладнює інтеграцію, гармонізацію та аналіз, вимагаючи складних методів обробки для отримання значущих висновків;
- обсяг і масштабованість. Обсяги неструктурованих медичних даних продовжують стрімко зростати, перевантажуючи традиційні інструменти та інфраструктури обробки даних. Традиційні системи намагаються ефективно масштабуватися, щоб впоратися з величезним потоком даних, що призводить до виникнення "вузьких місць" в обробці та неефективності використання часу;
- якість даних і шум: неструктуровані дані часто містять шум, невідповідності, скорочення, орфографічні помилки та несуттєву

інформацію, що впливає на точність і надійність аналізу. Очищення, попередня обробка та нормалізація цих даних зі збереженням важливого клінічного контексту є складним завданням;

- конфіденційність і безпека: захист конфіденційності пацієнтів і дотримання суворих нормативних вимог (наприклад, HIPAA, GDPR) при обробці та обміні конфіденційними медичними даними додає складнощів. Процеси анонімізації та деідентифікації повинні бути надійними, щоб забезпечити відповідність вимогам без шкоди для корисності даних.

Вирішення такого спектру задач вимагає повного використання потенціалу технологій Big Data. Такі підходи є перспективним шляхом подолання цих перешкод, пропонуючи масштабовані, розподілені обчислювальні фреймворки, передову аналітику та алгоритми машинного навчання для обробки, аналізу та отримання інсайтів з цього складного ландшафту даних.

Актуальність роботи обумовлена в зростаючій потребі в ефективному аналізі та обробці великих обсягів даних. Технології Big Data (BD) набувають все більшої актуальності, оскільки вони дозволяють обробляти великі обсяги даних, вилучати з них корисну інформацію та використовувати її для різних цілей, особливо в такій важливій галузі як медицині. Це все може значно підвищити якість, ефективність і результативність надання медичних послуг і медичних досліджень, а також вирішити важливі питання управління даними, конфіденційності та етики в умовах швидкозмінного ландшафту технологій охорони здоров'я.

Мета роботи полягає в розробці системи обробки та аналізу даних у медичній сфері задля покращення результатів охорони здоров'я, вдосконалення медичних досліджень та створення персоналізованої медицини. Крім того, оцінюється вплив технологій великих даних на надання медичних послуг, прийняття клінічних рішень та догляду за пацієнтами.

Об’єктом дослідження є процес обробки та аналізу даних у медичній галузі, зокрема, за допомогою технологій Big Data. Ця сфера представляє величезний

практичний інтерес, оскільки знаходиться на перетині охорони здоров'я та інформаційних технологій - двох галузей, що стрімко розвиваються. У сфері охорони здоров'я зростаючий обсяг і складність даних - від електронних медичних записів і геномних даних до систем моніторингу пацієнтів у режимі реального часу - є одночасно і викликом, і можливістю. Ефективне управління та аналіз цих даних мають вирішальне значення для покращення догляду за пацієнтами, вдосконалення медичних досліджень та підвищення загальної ефективності системи охорони здоров'я.

Предметом дослідження є методи аналітики великих даних у медичній сфері. Зокрема, досліджується, як інтеграція аналітики великих даних може покращити обробку, інтерпретацію та використання великих обсягів даних у сфері охорони здоров'я.

Завдання роботи зумовлене метою розробки комплексної системи обробки та аналізу медичних даних з використанням технологій великих даних, включає низку конкретних послідовних кроків, спрямованих на вирішення проблем та використання можливостей на перетині медичних даних та передових обчислювальних технологій. Для досягнення поставленої мети потрібно вирішити такі завдання:

- всебічний огляд і аналіз існуючих технологій і методів роботи з великими даними, які зараз застосовуються в медичній галузі. Це передбачає визначення їхніх сильних сторін, обмежень і придатності для різних типів медичних даних.;
- проектування та розробка системи обробки даних;
- аналіз та порівняння різних методів обробки та аналізу даних, включаючи збір, зберігання, інтеграцію та аналітику даних, підкреслюючи їх значення в медичній сфері. Ця система повинна бути здатна ефективно керувати великими обсягами різноманітних медичних даних, забезпечуючи швидкість і точність обробки та аналізу даних;
- впровадження передових аналітичних, пристосованих до конкретних потреб і нюансів медичних даних. Система буде ретельно протестована і

оцінена в реальних сценаріях щоб оцінити її продуктивність, масштабованість і практичну корисність в медичних установах. Такий комплексний підхід гарантує, що завдання, які виконуються в рамках цієї роботи, систематично ведуть до досягнення основної мети - покращення обробки та аналізу медичних даних за допомогою технологій великих даних.

Апробація результатів дослідження. Результати кваліфікаційної (магістерської) роботи апробовано на IV Всеукраїнській науково-практичній конференції «Комп'ютерні технології обробки даних» (КТОД 2023), яка відбулась 8 грудня 2023 року на базі кафедри інформаційних технологій Донецького національного університету імені Василя Стуса. Тези на тему: «Система обробки та аналізу даних у медичній сфері за допомогою технологій Big Data» були опубліковані в електронному збірнику наукових праць, який можна переглянути за посиланням https://jktod.donnu.edu.ua .

РОЗДІЛ 1
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ. СУЧАСНИЙ СТАН ОБРОБКИ ТА АНАЛІЗУ МЕДИЧНИХ ДАНИХ

1.1 Опис актуальності предметної області

Інтеграція технологій великих даних у медичну сферу є предметом першорядної важливості в сучасній системі охорони здоров'я та наукових дослідженнях. Актуальність зумовлена стрімким зростанням обсягу, різноманітності та швидкості поширення даних у медичному секторі. Епоха цифрових медичних записів, геномного секвенування та носимих медичних пристроїв вивільнила безпрецедентний потік даних [1]. Традиційні методи обробки та аналізу даних все частіше виявляються недостатніми, щоб впоратися з цим потоком, що робить потребу в надійних технологіях великих даних більш критичною, ніж будь-коли.

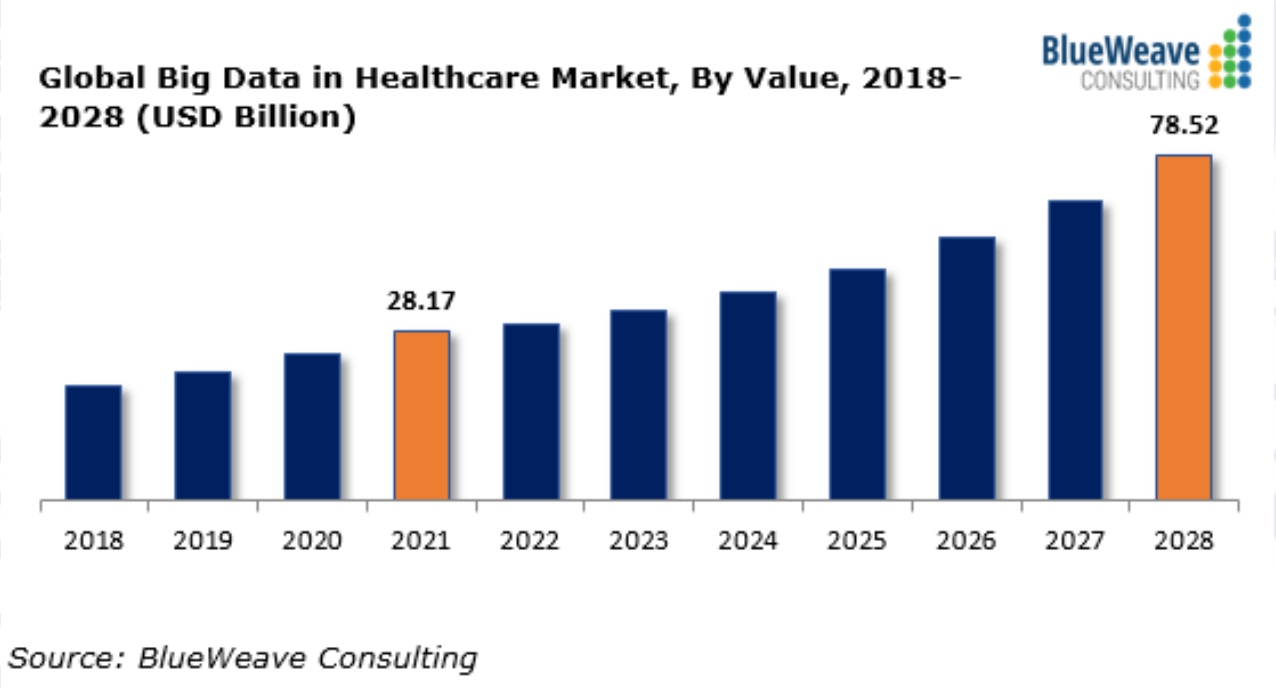


Рисунок 1.1 – Графік зростання ринку Big Data у медичній сфері [2]

Важливість цієї теми полягає в кількох ключових аспектах. По-перше, здатність ефективно обробляти та аналізувати великі обсяги даних про стан здоров'я може значно покращити результати лікування пацієнтів. Технології Big Data дозволяють аналізувати складні масиви даних для виявлення закономірностей і кореляцій, які раніше було неможливо виявити. Наприклад,

предиктивна аналітика може передбачити погіршення стану пацієнта, що дозволяє вжити заходів на більш ранній стадії, або персоналізувати плани лікування на основі унікального профілю здоров'я пацієнта [3].

Актуальність великих даних в охороні здоров'я, особливо в обробці медичних зображень та геноміці, підкреслюється їхньою здатністю значно вдосконалити традиційні методи. Традиційні підходи часто покладаються на ручний аналіз даних медичними працівниками, який, хоч і є ефективним, але може пропустити складні деталі, особливо у величезних обсягах даних, що генеруються в сучасній медицині. На противагу цьому, технології великих даних, використовуючи машинне навчання (ML) та розпізнавання образів, можуть виокремлювати біомаркери критичних захворювань з медичних зображень, трансформуючи діагностику, лікування та моніторинг пацієнтів. Такий підхід не лише ефективніший, але й компенсує брак спеціалістів у певних галузях медицини [4].

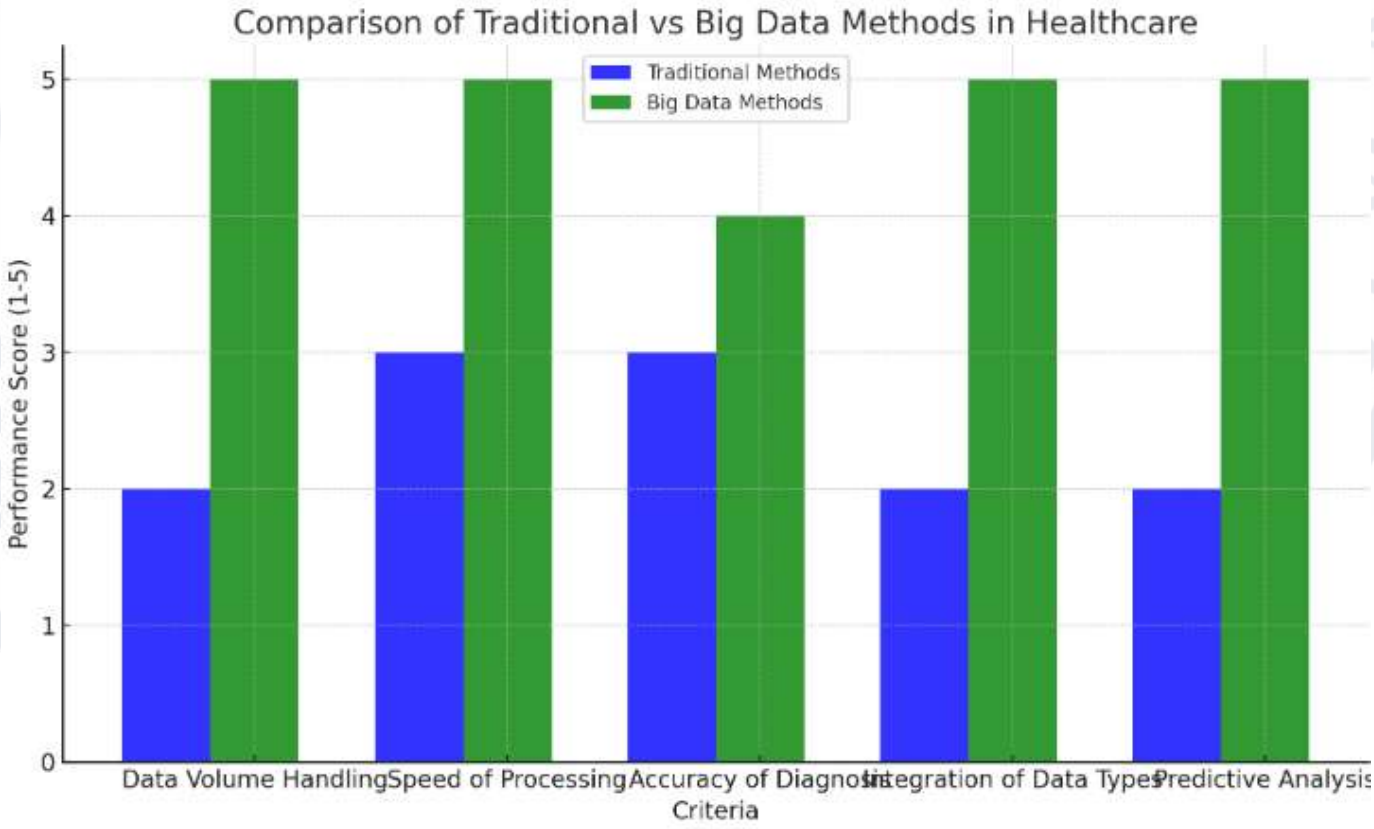


Рисунок 1.2 – Гістограма порівняння традиційних методів та методів великих даних в медицині

Наведена гістограма ілюструє порівняння між традиційними методами та методами великих даних у різних аспектах обробки та аналізу даних у сфері охорони здоров'я.

На діаграмі порівнюються п'ять ключових критеріїв:

- обробка великих обсягів даних: наскільки добре кожен метод може обробляти великі обсяги даних;
- швидкість обробки: ефективність і швидкість, з якою обробляються дані;
- точність діагностики: точність і правильність медичної діагностики;
- інтеграція типів даних: здатність інтегрувати та аналізувати різні типи медичних даних;
- прогностичний аналіз: ефективність у прогнозуванні медичних результатів або тенденцій.

Кожен критерій оцінюється за шкалою від 1 до 5, де 5 означає відмінну ефективність, а 1 - погану. Як показано на графіку, методи великих даних загалом перевершують традиційні методи за всіма критеріями, особливо в обробці великих обсягів даних, швидкості обробки та інтеграції різних типів даних. Графік наочно демонструє значні переваги методологій Big Data в охороні здоров'я над традиційними підходами.

Ці досягнення демонструють перехід від традиційних ручних і розрізнених методів до більш інтегрованого, ефективного і глибокого підходу, який пропонують технології великих даних в охороні здоров'я. Традиційно аналіз даних у сфері охорони здоров'я обмежувався проблемами, пов'язаними з їхнім обсягом, різноманітністю та швидкістю. Однак з появою технологій Big Data з'явилася безпрецедентна можливість розкрити потенціал цих великих і складних наборів даних. Інтеграція технологій великих даних у медичну сферу відкриває багатообіцяючі перспективи для революційних змін у наданні медичної допомоги, медичних дослідженнях та прийнятті рішень. Ефективно використовуючи можливості аналітики, медичні працівники можуть отримати цінну інформацію про популяції пацієнтів, закономірності перебігу захворювань, результати лікування та тенденції у сфері охорони здоров'я. Ці знання можуть

сприяти прийняттю рішень на основі доказової бази, вдосконаленню клінічних практик та покращенню результатів лікування пацієнтів.

Крім того, використання таких технологій може дозволити розвивати підходи персоналізованої медицини, підбираючи лікування для окремих пацієнтів на основі їхніх унікальних особливостей та історії хвороби. Це може призвести до більш цілеспрямованих та ефективних втручань, мінімізації несприятливих наслідків та оптимізації ресурсів охорони здоров'я. Однак впровадження аналітики великих даних у медичній сфері також пов'язане зі значними викликами. Питання конфіденційності та безпеки, пов'язані з обробкою конфіденційних даних пацієнтів, інтеграцією даних з різних джерел, забезпеченням якості даних, а також потреба в просунутих навичках аналітики є одними з ключових перешкод, які необхідно вирішити [5].

Сфера дослідження охоплює систему обробки та аналізу даних у медичній сфері з використанням технологій Big Data. Основна увага приділяється застосуванню передової аналітики та підходів, заснованих на даних, до великих і різноманітних наборів медичних даних.

Дослідження вивчатиме переваги, виклики та практичне застосування аналітики великих даних у медичній галузі. Будуть розглянуті методи і методології, пов'язані зі збором, зберіганням, інтеграцією, попередньою обробкою та аналітикою даних, специфічних для медичної сфери. Однак важливо зауважити, що це дослідження має певні обмеження. По-перше, основна увага буде зосереджена на технічних аспектах обробки та аналізу великих даних у медичній галузі, і дослідження не буде широко охоплювати клінічні або медичні аспекти. Воно передбачає базове розуміння процесів і термінології в галузі охорони здоров'я. По-друге, хоча дослідження має на меті надати всебічний огляд, воно не може охопити всі потенційні застосування, методи та виклики, пов'язані з даною темою. Стрімкий розвиток технологій і практики охорони здоров'я може призвести до появи нових розробок, які виходять за рамки цього дослідження. Крім того, для ілюстрації потенціалу технологій великих даних у медичній сфері дослідження спиратиметься насамперед на наявну літературу, тематичні дослідження та гіпотетичні сценарії. Дослідження може не

передбачати великого збору первинних даних або проведення експериментів у реальному часі.

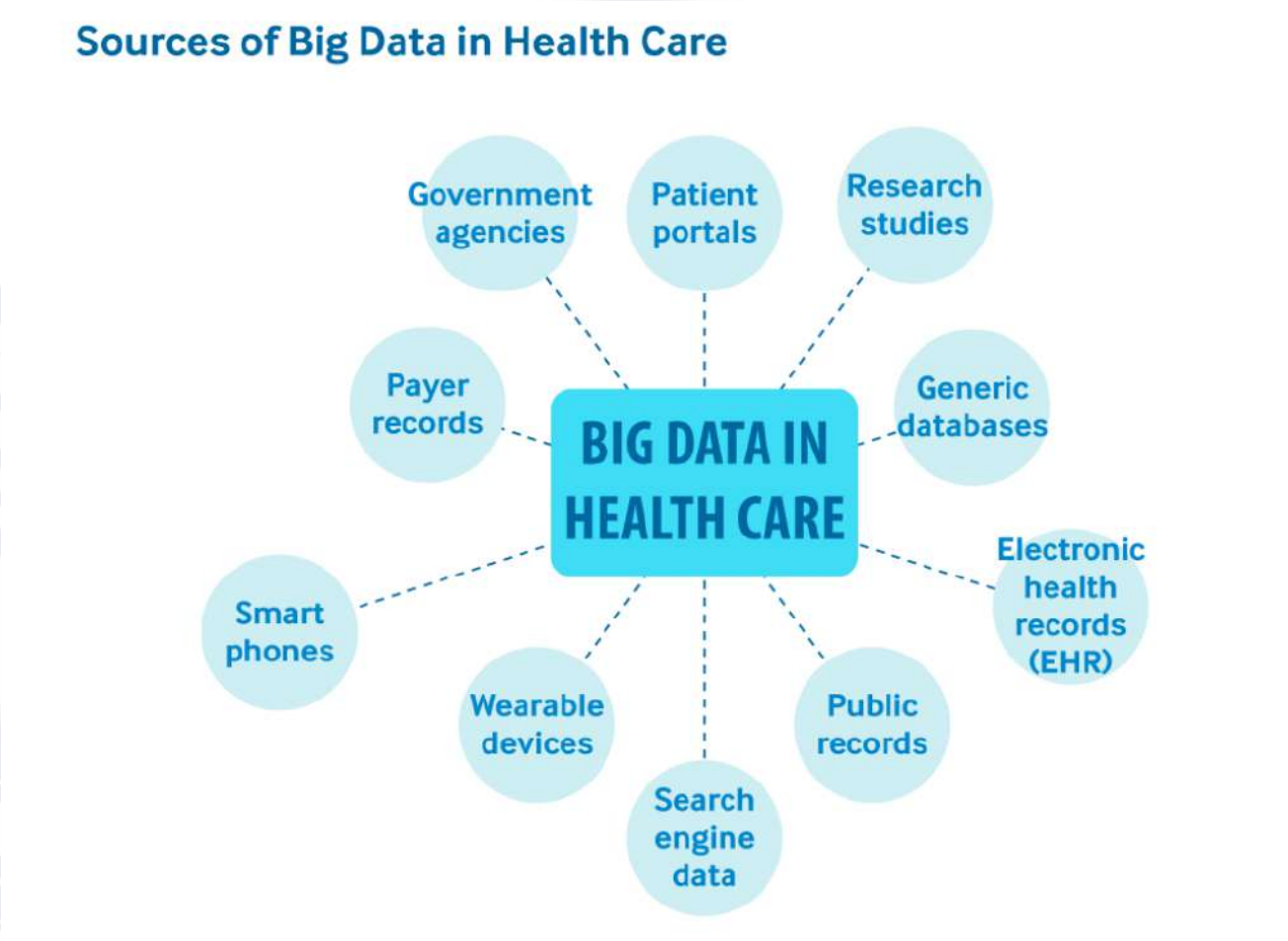


Рисунок 1.3 – Джерела великих даних в медицині [6].

Отже, сучасні підходи все більше зосереджуються на інтеграції та аналізі різноманітних і складних наборів даних. Ця інтеграція включає геномну інформацію, електронні медичні картки (ЕМК) та дані з різних біометричних пристроїв моніторингу. Сучасні методи обробки даних відіграють ключову роль у розвитку персоналізованої медицини, пропонуючи більш цілеспрямовані та ефективні плани лікування, засновані на індивідуальних профілях пацієнтів.

Однак ця галузь стикається зі значними проблемами, особливо в інтеграції багатотомних наборів даних, які включають складні шари біологічної інформації. Іншим важливим викликом є забезпечення можливості узагальнення результатів медичних досліджень для різних груп населення. Складність даних також викликає занепокоєння з точки зору етики та конфіденційності, що вимагає ретельного управління та регулювання [7].

Незважаючи на ці виклики, потенціал великих даних у медицині величезний. Вони обіцяють покращену прогностичну аналітику, що може призвести до покращення стратегій профілактики захворювань та більш ефективного надання медичної допомоги. Галузь продовжує розвиватися, а поточні дослідження спрямовані на подолання існуючих обмежень і повне використання можливостей великих даних в охороні здоров'я.

Потреба в аналізі Big Data

Сучасний світ переповнений даними, що генеруються з різних джерел, включаючи цифрові медіа, соціальні мережі та інтернет речей. Обробка цих величезних обсягів даних стає викликом, але водночас відкриває нові можливості для зростання продуктивності та інновацій [8].

Термін "великі дані" використовується для опису наборів даних, які настільки великі та складні, що їх важко обробити за допомогою традиційних інструментів управління базами даних або додатків обробки даних. Ці дані характеризуються трьома основними атрибутами [9]:

- обсягом,
- швидкістю
- структурованістю.

Обсяг даних відноситься до кількості даних, які потрібно зберігати та обробляти. Це може включати системні журнали, облікові записи та інші дані, накопичені протягом багатьох років.

Швидкість даних відноситься до темпу, з яким дані надходять та потребують обробки. Це може бути особливо важливо для онлайн-магазинів та інтернет-маркетингових організацій, які потребують швидкої обробки даних для надання рекомендацій та прогнозування інформації.

Структурованість даних відноситься до формату даних, які можуть бути структурованими або неструктурованими. Неструктуровані дані можуть включати текстові документи, відео, аудіо дані, зображення та інше.

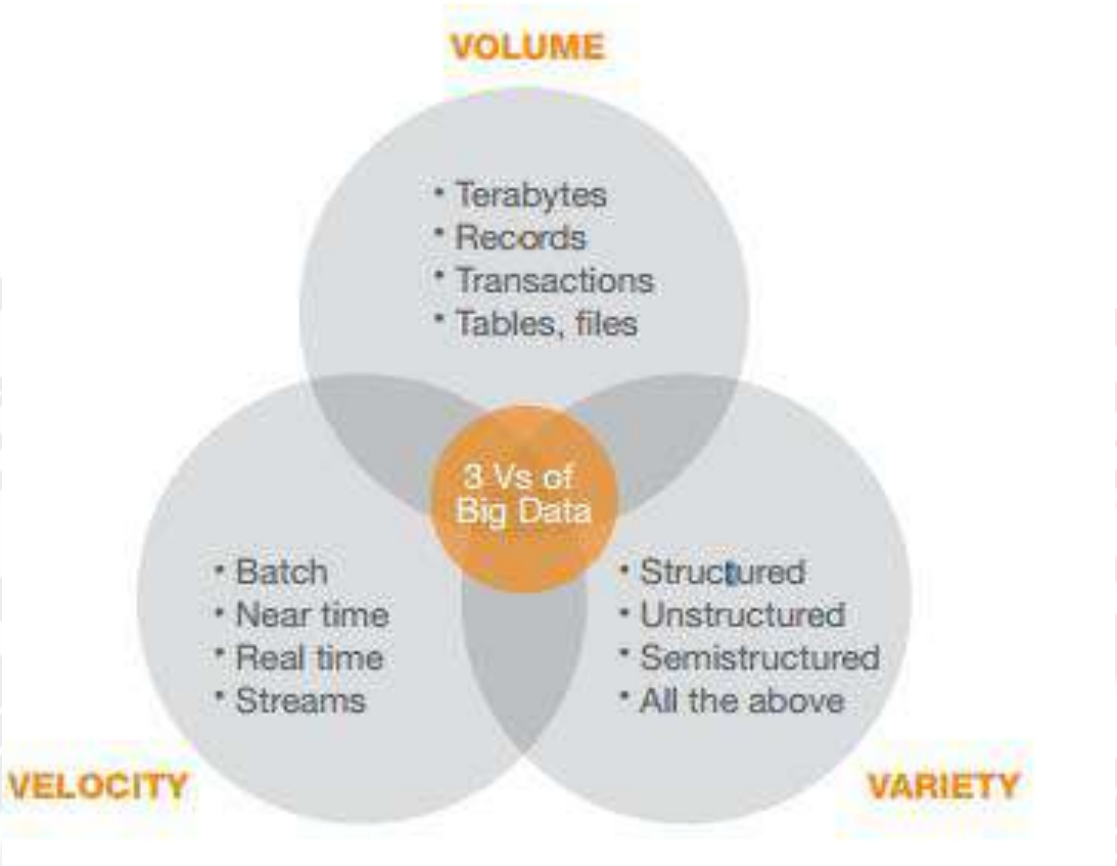


Рисунок 1.4 – Атрибути великих даних [10]

Аналіз великих даних вимагає нового підходу, оскільки традиційні реляційні бази даних можуть не справлятися з великими обсягами. Технології великих даних, які використовують хмарні обчислення та відкрите програмне забезпечення, дозволяють ефективно обробляти великі обсяги даних. Вони дозволяють зберігати всі дані для подальшого аналізу, враховуючи вірування, що в кожному біті даних може бути прихована важлива і цінна інформація.

Технології великих даних перевершують реляційні бази даних по низці критеріїв, включаючи потребу в більш складних резервних копіях, відновленні та більш швидких алгоритмах пошуку. Завдяки більш дешевому технічному обладнанню, хмарним обчисленням та відкритому програмному забезпеченню обробка великих обсягів даних стала набагато простішою.

Сектор охорони здоров'я генерує величезну кількість даних з різних джерел, включаючи електронні медичні картки (ЕМК), медичну візуалізацію, геноміку, носимі пристрої, датчики, соціальні мережі та медичні додатки. Ці дані

охоплюють широкий спектр інформації, включаючи демографічні дані пацієнтів, історії хвороби, діагностичні тести, плани лікування та нінше. Обсяг медичних даних продовжує зростати в геометричній прогресії завдяки технологічному прогресу та все ширшому впровадженню цифрових технологій в дану сферу

Використання технологій Big Data в охороні здоров'я пропонує кілька потенційних переваг [11]. Аналізуючи великі та різноманітні набори даних, медичні працівники та дослідники можуть отримати цінну інформацію про популяції пацієнтів, моделі захворювань, результати лікування та тенденції у сфері охорони здоров'я. Ці знання можуть допомогти у прийнятті рішень на основі фактичних даних, покращити клінічну практику та результати лікування пацієнтів. Одне з ключових застосувань великих даних в охороні здоров'я - предиктивна аналітика. Аналізуючи історичні дані про пацієнтів, можна розробити прогностичні моделі для прогнозування перебігу хвороби, виявлення пацієнтів з високим ризиком розвитку певних станів і передбачення потенційних несприятливих подій. Це дає змогу здійснювати проактивні втручання та розробляти персоналізовані плани лікування, що в кінцевому підсумку призводить до покращення догляду за пацієнтами та покращення результатів їхнього здоров'я.

У сфері великих даних у медицині виділяються кілька сучасних рішень, кожне з яких має свої переваги та виклики [12]:

- носимі пристрої та мобільні додатки для здоров'я. Ці технології трансформують спосіб моніторингу та управління здоров'ям. Носимі пристрої забезпечують безперервний потік даних про стан здоров'я, що дозволяє створювати більш точні профілі здоров'я та моделі прогнозування захворювань. Наприклад, схвалені FDA глюкометри взаємодіють з цифровими додатками, безпосередньо зв'язуючись з медичними працівниками. Такий підхід дозволяє втручатися на ранній стадії та покращити управління здоров'ям. Однак ці пристрої все ще розвиваються від рекреаційних до клінічних, і з часом виникає потреба у підвищенні їхньої точності та надійності;

- геноміка та глибоке молекулярне профілювання. Сучасні технології, такі як геноміка, сприяють глибшому розумінню захворювань на молекулярному рівні. Аналізуючи послідовності ДНК і РНК, дослідники можуть будувати прогностичні моделі та визначати ключові фактори розвитку таких захворювань, як хвороба Альцгеймера. Цей підхід змістив фокус з традиційних гіпотез на розуміння, засноване на даних, що призвело до появи нових терапевтичних стратегій. Однак, доступність залишається проблемою, оскільки таке просунуте секвенування здебільшого доступне лише у великих онкологічних центрах;
- пацієнто-партнерські дослідницькі ініціативи. Такі ініціативи, як проект "Count Me In", є прикладом досліджень за участю пацієнтів, що дозволяють пацієнтам ділитися своїми медичними записами, зразками та генетичною інформацією з дослідниками. Такий підхід прискорює дослідження рідкісних захворювань і сприяє більш ефективному лікуванню. Він також демонструє потенціал використання даних пацієнтів для стимулювання досліджень та покращення охорони здоров'я.

Хоча ці рішення пропонують революційні можливості в охороні здоров'я, вони також несуть з собою певні виклики. Однією з найважливіших проблем є забезпечення того, щоб обчислювальні моделі, які використовуються для оцінки стану здоров'я, були вільні від упереджень, які можуть поглибити нерівність у сфері охорони здоров'я. Крім того, безпека та етичне поводження з персональними медичними даними мають першорядне значення для запобігання зловживанням.

Обробка даних

Обробка даних включає перетворення вихідних даних у формат, який є зручним та бажаним для подальшого використання. Цей процес може включати ряд операцій, виконаних вручну або автоматично, зазвичай за допомогою комп'ютерів. Оброблені дані можуть бути представлені у різних формах, включаючи зображення, графіки, таблиці, векторні файли, аудіо, діаграми та інші формати, в залежності від використовуваного програмного забезпечення

або методу обробки даних. Основна мета обробки даних - це синхронізація вводу даних у програмне забезпечення для відбору найбільш корисної інформації. Це важливий процес для будь-якої організації, оскільки це дозволяє видобувати найбільш відповідний контент для подальшого використання. Всі ключові сектори, включаючи банки, освітні установи та великі компанії, використовують обробку даних для зберігання найбільш цінної інформації в своїх системах. Ручна обробка даних може бути затратною по часу і вимагати значного вкладу людських ресурсів, особливо при роботі з великими обсягами даних. Сьогодні багато секторів залежать від потужних та ефективних програмних інструментів для обробки цих даних, що допомагає підвищити точність та ефективність [13].

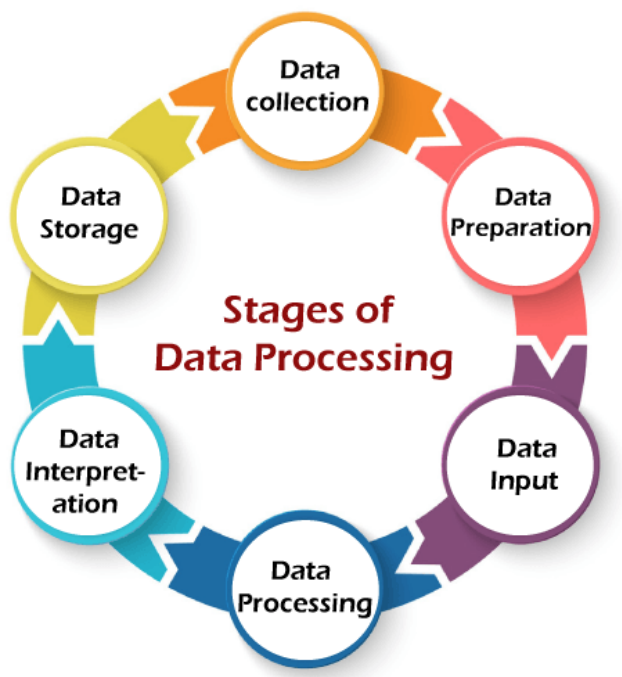


Рисунок 1.5 – Основні етапи обробки даних

Обробка даних, в контексті комп'ютерних наук, визначається як серія операцій над даними, виконаних комп'ютером, з метою отримання, трансформації або класифікації інформації. Цей процес може включати різні підходи, такі як адміністративна обробка даних, автоматична обробка даних, обробка даних для бізнесу, дата-майнінг, розподілена обробка даних, обробка списків, багатопроцесорна обробка, операції в реальному часі, обробка з високим пріоритетом, обробка віддалених даних, послідовна обробка та інші.

1.2 Практичне застосування технологій Big Data в медицині

Стосовно практичного застосування технологій Big Data у медицині слід виділити такі:

- предиктивна аналітика в онкології. Застосування: технології великих даних використовують для аналізу величезних масивів даних з історій хвороб та геномних даних онкологічних пацієнтів. Результат: Виявлення закономірностей і кореляцій, що дозволило розробити персоналізовані плани лікування. Наприклад, використовуючи предиктивну аналітику, онкологи можуть визначити, які методи лікування раку, найімовірніше, будуть ефективними для конкретних профілів пацієнтів на основі генетичних маркерів. Вплив: цей підхід значно покращив показники успішності лікування та зменшив побічні ефекти від неефективних методів лікування;
- переносна технологія для лікування хронічних захворювань. Застосування: використання натільних пристроїв для моніторингу пацієнтів з хронічними захворюваннями, такими як діабет і хвороби серця. Результат: безперервний збір даних з цих пристроїв забезпечує розуміння стану здоров'я пацієнтів у реальному часі, що дозволяє вчасно вжити медичних заходів. У лікуванні діабету носимі глюкометри зробили революцію в тому, як пацієнти контролюють рівень цукру в крові. Вплив: Покращення управління захворюванням, підвищення якості життя пацієнтів та зменшення кількості госпіталізацій через ускладнення;
- великі дані у реагуванні на пандемію. Застосування: під час пандемії COVID-19 великі дані відіграли вирішальну роль у відстеженні рівня інфікування, моделюванні поширення вірусу та формуванні політики у сфері охорони здоров'я. Результат: аналіз даних дозволив органам охорони здоров'я ефективно розподіляти ресурси, прогнозувати потреби в госпіталізації та впроваджувати ефективні стратегії стримування. Вплив: Ці зусилля відіграли важливу роль в управлінні пандемією та мінімізації її впливу на громадське здоров'я та системи охорони здоров'я.

Аналіз цих прикладів демонструє трансформаційну силу технологій великих даних у медицині [14]. В онкології предиктивна аналітика відкрила еру персоналізованої медицини, коли лікування підбирається з урахуванням генетичних особливостей людини, що значно покращує результати. Це означає зміну парадигми від традиційних, універсальних підходів до більш нюансованого та ефективного лікування. Використання натільних технологій у лікуванні хронічних захворювань підкреслює перехід до превентивної медицини. Забезпечуючи безперервний моніторинг, ці технології дозволяють втручатися на більш ранніх стадіях, що може запобігти ускладненням і зменшити потребу в госпіталізації. Це не лише покращує результати лікування пацієнтів, але й зменшує навантаження на системи охорони здоров'я. Реакція на пандемію COVID-19 демонструє критично важливу роль великих даних у сфері охорони здоров'я. Здатність швидко обробляти великі обсяги даних дозволила оперативно реагувати на ситуацію, що розвивається. Таке застосування великих даних підкреслює їхню важливість в епідеміології та формуванні політики громадського здоров'я.

Однак ці застосування також висвітлюють виклики, пов'язані з використанням великих даних у медицині. Покладання на якісні дані має першорядне значення, оскільки неточності можуть призвести до помилкових висновків і шкідливих наслідків. Крім того, існують значні занепокоєння щодо конфіденційності даних та етичного використання інформації про пацієнтів, які необхідно вирішити для збереження суспільної довіри.

Отже, реальне застосування великих даних у медицині продемонструвало значні переваги у покращенні догляду за пацієнтами, просуванні медичних досліджень та допомозі в управлінні громадським здоров'ям. Подальший розвиток і етичне застосування цих технологій мають вирішальне значення для реалізації їхнього повного потенціалу в трансформації.

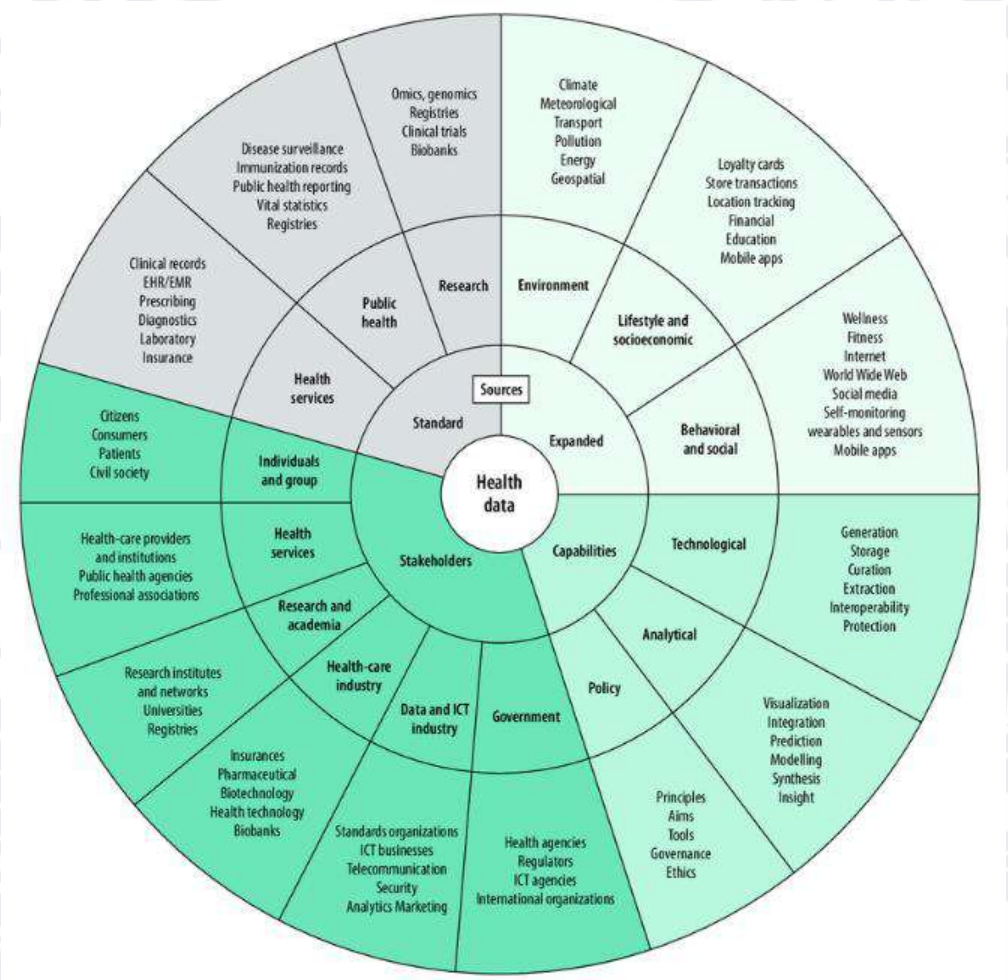


Рисунок 1.6 - Екосистема даних, що розвивається в сфері охорони здоров’я

1.3 Аналіз існуючих рішень

Система IBM Watson

Однією з визначних систем, призначених для обробки та аналізу даних у сфері великих даних, зокрема в охороні здоров'я, є IBM Watson Health [15]. Ця система використовує передові технології штучного інтелекту та машинного навчання для обробки величезних обсягів даних про здоров'я для різних застосувань.

IBM Watson Health, як провідна система в галузі великих даних та охорони здоров'я, може похвалитися кількома ключовими особливостями, які визначають її корисність та ефективність. Центральною функцією є можливість інтеграції даних. Ця система вміє об'єднувати широкий спектр джерел даних, від електронних медичних записів і геномних даних до інформації, отриманої в результаті клінічних випробувань і наукових досліджень. Такий цілісний підхід до агрегації даних є фундаментальним для створення комплексної аналітичної платформи в галузі охорони здоров'я. Доповнюючи можливості інтеграції даних, IBM Watson Health використовує когнітивні обчислення, зокрема, обробку природної мови (NLP). Ця передова технологія дозволяє системі аналізувати та осмислювати неструктуровані дані, які є поширеним форматом у медичних записах та літературі. Інтерпретуючи клінічні записи, наукові роботи та інші текстові дані, система витягує цінну інформацію, яка інакше залишилася б прихованою в складних неструктурованих форматах [16].

Прогностична аналітика є ще одним наріжним каменем IBM Watson Health, в основі якого лежать складні алгоритми машинного навчання. Ці алгоритми просіюють величезні обсяги інтегрованих даних для виявлення закономірностей і тенденцій. Ця функція особливо важлива для прогнозування результатів лікування пацієнтів, розуміння прогресування хвороби та визначення ймовірної ефективності різних варіантів лікування. Прогностичні можливості системи відкривають двері для проактивних медичних втручань і більш обґрунтованого прийняття рішень медичними працівниками.

Рисунок 1.7 – IBM Watson Health Systems

Основні переваги такого комплексного рішення:

- комплексний аналіз даних. Може обробляти та аналізувати великі обсяги різноманітних медичних даних, забезпечуючи глибоке розуміння;
- швидкість та ефективність. Прискорює процес аналізу даних, заощаджуючи цінний час для прийняття клінічних рішень та проведення досліджень;
- підтримка прецизійної медицини. Розширює можливості надання персоналізованих планів лікування на основі індивідуальних даних пацієнта.

Недоліки:

- складність і вартість. Система є складною і може вимагати значних інвестицій, що робить її менш доступною для невеликих медичних закладів;
- Занепокоєння щодо конфіденційності даних. Обробка конфіденційних медичних даних викликає питання щодо конфіденційності та безпеки даних;
- надійність і довіра. Точність рекомендацій, наданих штучним інтелектом, є предметом дискусій, а також занепокоєння щодо надійності ШІ у прийнятті складних медичних рішень.

Apple Watch HealthCare

Apple Watch - флагманський продукт від технологічного гіганта Apple Inc., - це смарт-годинник, який все більше відіграє важливу роль у сфері особистого здоров'я та благополуччя. Цей пристрій є чудовим прикладом в сфері носимих пристроїв та мобільних додатків для здоров’я. Спочатку запущений як пристрій, орієнтований на спілкування та спосіб життя, Apple Watch перетворився на потужного помічника у сфері здоров'я та фітнесу. Його роль у сфері охорони здоров'я полягає у здатності відстежувати та контролювати різні показники здоров'я, трансформуючи спосіб взаємодії людей зі своїм здоров'ям [17].

Одним з ключових аспектів Apple Watch є його можливості моніторингу здоров'я. Він постійно відстежує частоту серцевих скорочень, що є ключовою функцією у виявленні потенційних проблем із серцем. Деякі моделі Apple Watch також включають функцію електрокардіограми (ЕКГ), що дозволяє користувачам проводити оцінку серцевого ритму на місці, що може мати вирішальне значення для виявлення ознак фібриляції передсердь. Цей вид моніторингу в режимі реального часу може забезпечити раннє попередження про серцеві захворювання, які в іншому випадку можуть залишитися непоміченими [18].Окрім моніторингу здоров'я серця, Apple Watch пропонує набір функцій, спрямованих на покращення загального самопочуття та фізичної форми. Він заохочує фізичну активність завдяки щоденному відстеженню активності, підрахунку кроків, оцінці спалених калорій та встановленню персоналізованих рухових цілей. Для багатьох користувачів ці функції стали невід'ємною частиною їхніх фітнес-процедур, пропонуючи зручний і мотивуючий інструмент для підтримки активного способу життя.

Основні функції даного пристрою:

- відстеження загального стану здоров'я та фітнесу. Відстежує щоденні дії, такі як кроки, спалені калорії та загальну фізичну активність;
- моніторинг серцевого ритму. Відстежує частоту серцевих скорочень і може попередити користувача про нерегулярні серцеві ритми, що потенційно можуть свідчити про серйозні проблеми зі здоров'ям;

- функція ЕКГ. Новіші моделі включають функцію електрокардіограми (ЕКГ) для виявлення ознак фібриляції передсердь;
- виявлення падіння та екстрений SOS. виявляє, якщо користувач падає, і може автоматично викликати екстрені служби, якщо користувач не реагує;
- відстеження сну. Відстежує режим сну, щоб допомогти користувачам покращити якість сну;
- дослідження та збір даних. Сприяє дослідженням у сфері охорони здоров'я шляхом збору даних у дослідженнях здоров'я серця, жіночого здоров'я, руху та слуху.

Переваги:

- проактивне управління здоров'ям. Надає дані про стан здоров'я в режимі реального часу, що дозволяє користувачам робити проактивні кроки в управлінні своїм здоров'ям;
- зручність та зв'язок. Інтегрується з iPhone для безперебійної роботи користувача, включаючи сповіщення та інші інтелектуальні функції;
- залучення та мотивація користувачів. Заохочує користувачів бути більш активними та уважними до свого здоров'я;
- внесок у дослідження. Допомагає науковим дослідженням завдяки масштабному збору даних.

Недоліки:

- точність і надійність: Незважаючи на вдосконалення, точність вимірювань стану здоров'я, особливо для медичної діагностики, може змінюватися;
- залежність і тривога: Постійний моніторинг може призвести до надмірної залежності або тривоги, пов'язаної зі здоров'ям, у деяких користувачів;
- конфіденційність і безпека даних. Збирає конфіденційні дані про стан здоров'я, що викликає занепокоєння щодо того, як ці дані зберігаються, використовуються та поширюються;

- вартість і доступність. Ціна може бути надто високою для деяких користувачів, що обмежує їхню доступність.

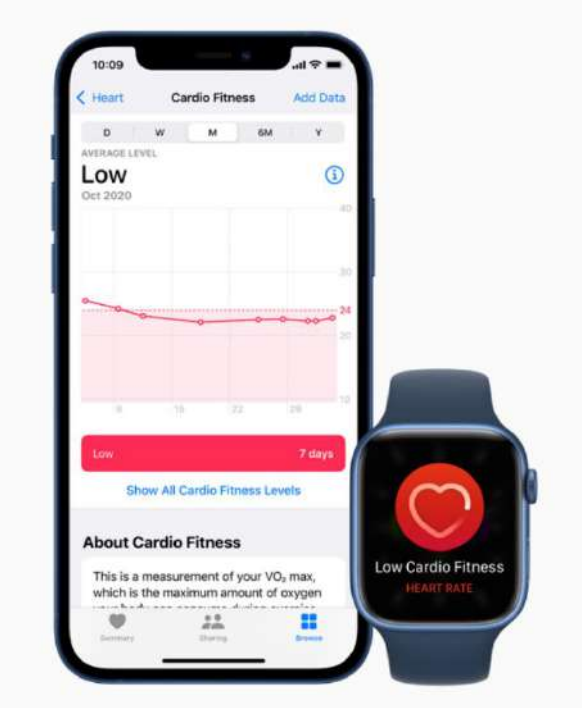


Рисунок 1.8 – Apple Watch Healthcare

Годинник Apple Watch є яскравим прикладом того, як технології, що носяться, стають невід'ємною частиною сучасного управління здоров'ям. Він демонструє потенціал носимих пристроїв для значного впливу на особистий моніторинг здоров'я та профілактику захворювань. Однак його впровадження і використання також висуває виклики, пов'язані з конфіденційністю, точністю і доступністю даних, які необхідно вирішувати в міру розвитку технології.

23andMe

23andMe - провідний гравець в індустрії споживчої генетики, пропонує цікавий погляд на те, як перетинаються великі дані та генетичне тестування. Заснована у 2006 році, компанія зробила генетичне тестування доступним для споживачів, надаючи інформацію про родовід, успадковані риси та потенційні ризики для здоров'я [19]. Дані, які збирає 23andMe відіграють важливу роль у різних дослідницьких проектах. Компанія співпрацює з дослідниками та фармацевтичними компаніями, використовуючи свою генетичну базу даних для виявлення генетичних маркерів, пов'язаних із захворюваннями та розробкою

нових ліків. Ця співпраця спрямована на використання можливостей великих даних у генетиці для більш широких переваг у сфері охорони здоров'я.

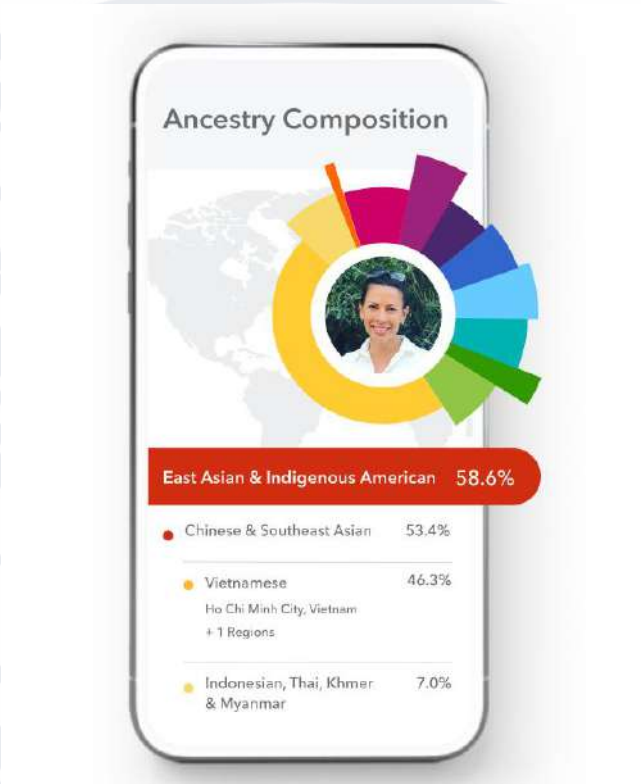


Рисунок 1.9 – сервіс 23andMe

Переваги:

- внесок у дослідження. Компанія накопичила величезну генетичну базу даних, яка надає цінні дані для медичних досліджень і вивчення генетичних зв'язків із захворюваннями;
- профілактична охорона здоров'я. виявляючи потенційні ризики для здоров'я, компанія заохочує людей до проактивних кроків в управлінні своїм здоров'ям;
- залучення та освіта користувачів. Сервіс навчає користувачів про генетику і здоров'я, сприяючи більшій обізнаності та залученню до питань особистого здоров'я.

Недоліки [20]:

- інтерпретація та точність. Розуміння генетичного ризику є складним. Існує ризик неправильного тлумачення, а точність прогнозування ризику може бути різною;

- занепокоєння щодо конфіденційності. Робота з чутливими генетичними даними піднімає значні питання конфіденційності, особливо щодо того, як ці дані зберігаються, використовуються або потенційно поширюються;
- психологічний вплив. Отримання інформації про генетичну схильність до серйозних захворювань може мати психологічний вплив на людей;
- регуляторні проблеми. Індустрія генетичного тестування безпосередньо для споживачів стикається з різними регуляторними перешкодами, особливо щодо інформації, пов'язаної зі здоров'ям, яка надається.

Аналіз сучасних рішень на основі великих даних у сфері охорони здоров'я, показує, що, хоча кожна з цих систем пропонує революційні досягнення в обробці та аналізі медичних даних, жодна з них не позбавлена проблем, що підкреслює тезу про те, що ідеальних систем не існує. Підсумовуючи, можна сказати, що хоча ці системи є значним кроком у використанні великих даних для покращення охорони здоров'я, кожна з них має свої обмеження. Від технічних та інфраструктурних проблем до етичних міркувань і питань конфіденційності - всі ці системи підкреслюють складність інтеграції технологій великих даних в охорону здоров'я. Розвиток цих систем продовжує залишатися балансуванням між використанням їхніх можливостей для покращення результатів у сфері охорони здоров'я та вирішенням незліченних проблем, які вони створюють.

1.4 Постановка задачі

Потрібно розробити комплексну систему обробки та аналізу даних у медичній галузі, використовуючи можливості технологій великих даних. Система призначена для перетворення великих і складних масивів медичних даних у практичні рішення, тим самим покращуючи догляд за пацієнтами, підтримуючи медичні дослідження та сприяючи розвитку персоналізованої медицини. Потрібно створити платформу, яка не тільки ефективно управляє та аналізує медичні дані з різних джерел, але й представляє цю інформацію у зручний та доступний спосіб.

Функції, які повинні бути доступні у системі обробки та аналізу даних в медичній сфері на основі технологій Big Data:

- збирання даних з різних джерел, таких як електронні медичні картки (ЕМК), бази даних;
- зберігання та управління отриманими даними, забезпечення їх доступності та цілісності;
- механізм обробки та трансформації даних. Система, що обробляє та перетворює необроблені дані у формат, придатний для аналізу;
- аналіз оброблених дані за допомогою алгоритмів машинного навчання та статистичних моделей для вилучення корисної інформації;
- інтерактивний інтерфейс для доступу, візуалізації та інтерпретації проаналізованих даних.

Основні компоненти, які є частиною системи обробки та аналізу даних в медичній сфера на основі технологій Big Data:

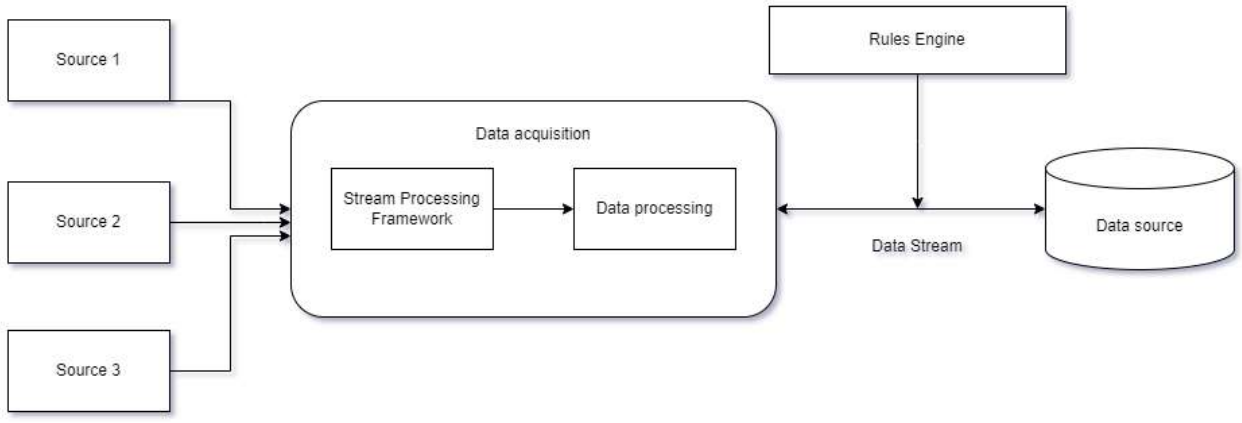


Рис. 1.10 – Основні компоненти системи

- модуль збору та зберігання даних. Цей компонент відповідає за збір та первинну обробку даних з різних джерел. Він включає в себе функції перевірки, очищення та попередньої обробки даних, також дбає про безпечне зберігання оброблених даних, використовуючи комбінацію реляційних і NoSQL баз даних;
- модуль обробки даних. Цей компонент займається перетворенням і нормалізацією необроблених даних, роблячи їх придатними для аналізу.

Він включає в себе більш складні завдання обробки даних, такі як очищення, трансформація та нормалізація даних. Призначений для ефективної роботи з великими наборами даних;

- модуль аналізу даних. Цей модуль є основним аналітичним мозком системи, де дані аналізуються для вилучення значущих інсайтів;
- інтерфейс користувача та компонент звітування. Цей модуль надає користувачам інтерактивний і зручний інтерфейс для доступу та візуалізації проаналізованих даних. Містить інструменти для створення звітів і візуалізації даних у реальному часі, що полегшує інтерпретацію результатів аналізу даних.

Висновки до розділу

У розділі були всебічно розглянуті різні аспекти, виклики та потенціал використання великих даних у галузі медицини. Цей розділ підтверджує актуальність і зростаюче значення технологій великих даних в охороні здоров'я, підкреслюючи, як вони трансформують догляд за пацієнтами, медичні дослідження та управління охороною здоров'я. Дослідження включало детальний аналіз існуючих систем, кожна з яких пропонує унікальне розуміння того, як великі дані використовуються для покращення надання медичної допомоги та медичних досліджень. Були детально розглянуто переваги цих систем, а також їхні недоліки. Поставлена задача роботи та описано основні компоненти системи.

РОЗДІЛ 2
ОПИС ОСНОВНИХ МОДУЛІВ СИСТЕМИ. МЕТОДИ ПОШУКУ АСОЦІАТИВНИХ ПРАВИЛ

2.1 Методи пошуку асоціативних правил

Методи пошуку асоціативних правил - це набір методів, що використовуються в інтелектуальному аналізі даних для виявлення зв'язків, закономірностей або асоціацій між змінними у великих базах даних. Ці методи особливо корисні в ситуаціях, коли зв'язок між різними елементами даних не є очевидним [21].

Як працюють методи пошуку асоціативних правил [22]:

- визначення наборів елементів, що часто зустрічаються. Перший крок - знайти всі набори елементів (itemsets), які часто з'являються в наборі даних;
- створення асоціативних правил. На основі цих наборів елементів генеруються правила. Ці правила припускають, що наявність певних елементів у транзакції означає наявність інших елементів;
- оцінка правил. Потім правила оцінюються на основі таких показників, як підтримка (наскільки часто правило застосовується) і впевненість (наскільки часто правило є точним).

В контексті системи обробки та аналізу медичних даних методи пошуку асоціативних правил можуть застосовуватися різними способами:

- аналіз даних пацієнтів. Аналізуючи записи пацієнтів, методи асоціативних правил можуть виявити закономірності, такі як кореляція між певними симптомами і діагнозами або між конкретними демографічними даними пацієнта і поширеністю захворювання;
- результати лікування. Ці методи можна використовувати для аналізу результатів лікування, допомагаючи визначити, які методи лікування є найбільш ефективними для певних станів або груп пацієнтів;

- взаємодія ліків та побічні ефекти. Асоціативні правила можуть виявити потенційну взаємодію між різними ліками або визначити загальні побічні ефекти ліків у певних популяціях пацієнтів;
- прогностична медицина. Виявляючи закономірності та асоціації в даних пацієнта, система може прогнозувати ризики для здоров'я або прогресування хвороби, допомагаючи в профілактиці та ранньому втручанні;
- індивідуальні медичні рекомендації. Система може використовувати асоціативні правила для надання пацієнтам персоналізованих рекомендацій щодо здоров'я та способу життя на основі їхньої історії хвороби та факторів способу життя.

Методи пошуку за асоціативними правилами, є потужним інструментом для виявлення прихованих закономірностей і асоціацій в медичних даних. Це може призвести до покращення догляду за пацієнтами, більш ефективного лікування та покращення процесу прийняття рішень у сфері охорони здоров'я.

Алгоритм APRIORI

Алгоритм Apriori є одним з найвідоміших і поширених алгоритмів аналізу асоціативних правил у галузі добування даних з баз даних. Він використовується для виявлення комбінацій елементів, які часто зустрічаються (так званих асоціативних правил) в наборах даних, таких як транзакційні бази даних [23].

Алгоритм Apriori працює на основі поняття підтримки. Він робить ітерації по базі даних, починаючи з одноелементних наборів (тобто окремих елементів), обчислюючи їх підтримку (частоту входження) у транзакціях. За допомогою цієї інформації він генерує кандидатські набори елементів більшої довжини. Після цього він сканує базу даних, щоб визначити підтримку цих кандидатських наборів. Алгоритм продовжує ітерувати цей процес, збільшуючи довжину наборів на кожній ітерації, доки не будуть вичерпані кандидатські набори або не буде досягнута задана межа підтримки. На виході алгоритм Apriori повертає набір асоціативних правил, які задовольняють заданим критеріям підтримки та правдоподібності.

Алгоритм Apriori дозволяє виявити часті комбінації елементів у великих наборах даних, що може бути корисним для виявлення прихованих зв'язків та залежностей між елементами в базі даних. Він знаходить застосування в таких областях, як маркетингові дослідження, рекомендаційні системи, аналіз покупок та інше [24].

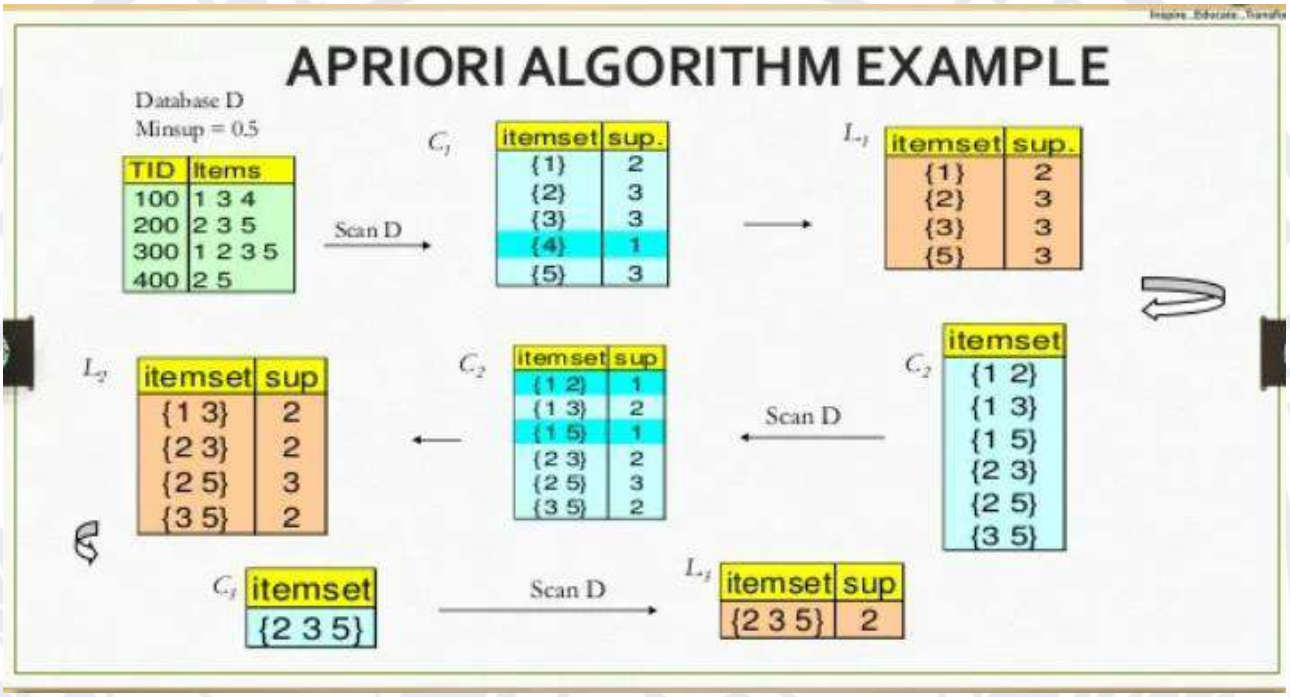


Рисунок 2.1 – Алгоритм Apriori

Алгоритм роботи Apriori:

- встановлюється мінімальний поріг підтримки. Алгоритм починається зі встановлення мінімального порогу підтримки. Цей поріг визначає мінімальну частоту, з якою набір елементів повинен з'являтися в наборі даних, щоб вважатися значущим;
- створення наборів елементів-кандидатів. Починається з визначення всіх окремих елементів у наборі даних, які відповідають мінімальному порогу підтримки. Ці елементи називаються частими наборами з 1 елемента;
- створення більших наборів елементів. Алгоритм об'єднує ці часті набори з 1 елемента для створення наборів з 2 елементів, потім з 3 елементів і так далі. На кожному етапі набори елементів, які не відповідають порогу підтримки, відсікаються;

- відсікання нечастих наборів елементів. Алгоритм використовує принцип, що підмножини частих наборів елементів також повинні бути частими. Якщо виявляється, що будь-яка підмножина набору елементів не є частою, набір елементів відкидається;
- створення правил асоціацій з частих наборів елементів: Після того, як визначено всі часті набори елементів, алгоритм генерує правила на їх основі.

Переваги алгоритму Apriori:

- простота і легкість розуміння. Алгоритм Apriori є простим і легким для реалізації;
- широке застосування. Його можна застосувати до будь-якого набору транзакційних даних, щоб знайти часті набори елементів і правила асоціацій;
- корисний для великих наборів даних. Ефективно працює з великими наборами даних, що робить його ідеальним для додатків на основі великих даних.

Недоліки:

- проблеми з масштабуванням. Алгоритм може бути дорогим в обчислювальному плані, оскільки може вимагати багаторазового сканування бази даних;
- інтенсивне використання пам'яті. Генерування великої кількості наборів кандидатів може споживати значний обсяг пам'яті;
- продуктивність. Продуктивність алгоритму може погіршуватися зі збільшенням кількості елементів і транзакцій

Алгоритм AIS

AIS (Artificial Immune System) [25] алгоритм відноситься до групи біоінспірованих алгоритмів, які були розроблені на основі принципів і процесів імунної системи. Ці алгоритми використовуються в різних областях, включаючи машинне навчання, оптимізацію та системи розпізнавання шаблонів. Алгоритм AIS проводить ітерації по всій базі даних з метою аналізу. На кожній ітерації він

сканує всі транзакції. Під час першої ітерації алгоритм обчислює підтримку окремих елементів і визначає, які з них є великими або частими у базі даних. Великі набори елементів кожної ітерації розширюються для створення кандидатських наборів елементів. Після сканування транзакції визначаються спільні набори елементів між великими наборами елементів з попередньої ітерації та елементами поточної транзакції. Алгоритм AIS був першим опублікованим алгоритмом, створеним для виявлення всіх великих наборів елементів у базі даних транзакцій. Основний акцент був зроблений на розширенні баз даних з необхідною функціональністю для обробки запитів щодо підтримки рішень. Алгоритм AIS був розроблений для виявлення частих наборів елементів у базах даних транзакцій. Ці набори потім використовуються для створення правил асоціації, які корисні для розуміння зв'язків між різними елементами в базі даних. Це має застосування в аналізі ринкових кошиків, управлінні запасами та інших сферах, де розуміння асоціацій між елементами є корисним [26].

Алгоритм роботи AIS:

- ініціалізація. Алгоритм починається з порожнього набору наборів елементів-кандидатів;
- сканування бази даних: На кожному проході база даних сканується і підраховується частота наборів елементів-кандидатів;
- генерація кандидатів: Після кожного сканування генеруються нові набори елементів-кандидатів на основі частих наборів елементів, знайдених під час попереднього проходу;
- відсікання: Алгоритм відсікає кандидати, які не відповідають мінімальному порогу підтримки;

- ітеративний процес: Цей процес повторюється до тих пір, поки не буде знайдено жодного нового частого набору елементів.

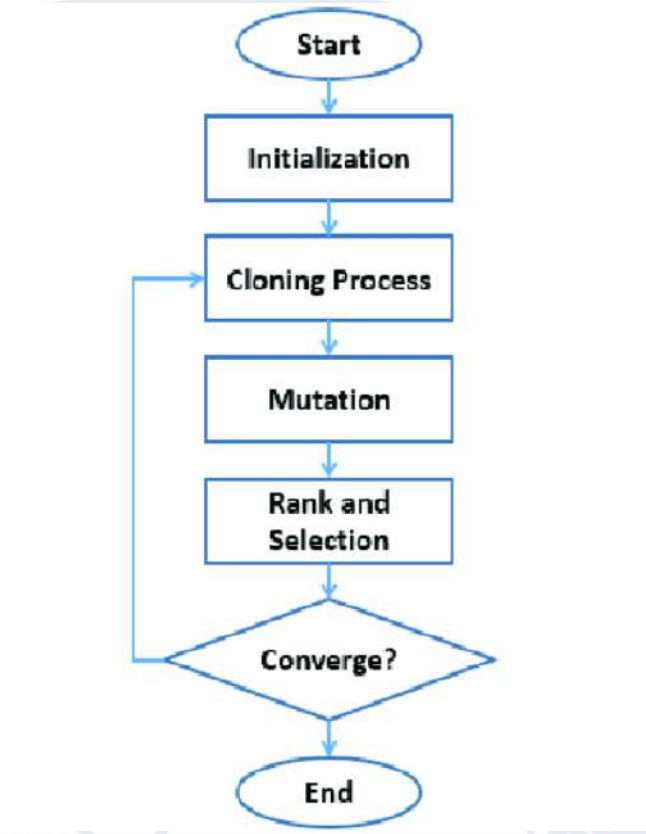


Рисунок 2.2 – схема алгоритму AIS

Переваги:

- новаторський підхід. AIS був одним з перших алгоритмів, який вирішив проблему пошуку частих наборів елементів у великих базах даних;
- фундамент для майбутніх алгоритмів. Він заклав основу для наступних алгоритмів видобутку асоціативних правил, таких як алгоритм Apriori;
- простота. Алгоритм AIS має відносно простий підхід до генерації та відсікання наборів елементів-кандидатів.

Недоліки:

- неефективність. Алгоритм може бути неефективним, особливо для великих баз даних, оскільки він може генерувати велику кількість наборів-кандидатів, які потрібно перевіряти на всій базі даних;
- масштабованість. Алгоритм погано масштабується залежно від розміру бази даних і кількості елементів, що може призвести до проблем з продуктивністю;

- багаторазове сканування бази даних. AIS вимагає декількох проходів по базі даних, що може зайняти багато часу.

Хоча алгоритм AIS став важливим кроком у галузі інтелектуального аналізу даних, його обмеження в ефективності та масштабованості призвели до розробки більш досконалих алгоритмів, таких як Apriori та FP-Growth. Сьогодні AIS слугує радше концептуальною основою та історичною довідкою в еволюції методів інтелектуального аналізу на основі асоціативних правил.

Алгоритм SETM

Алгоритм SETM, подібно до алгоритму AIS, працює з базою даних, виконуючи кілька проходів. Під час першого проходу він обчислює підтримку окремих елементів та визначає, які з них є великими або частими в базі даних. Наступною дією є генерація наборів кандидатів шляхом розширення великих наборів елементів з попереднього проходу. Крім цього, алгоритм SETM зберігає інформацію про TID (ідентифікатор транзакції) для генеруючих транзакцій, пов'язаних з набором елементів. Для генерації наборів елементів в алгоритмі SETM може бути використана реляційна операція злиття-з'єднання [27]. Розроблений як альтернатива алгоритмам AIS та Apriori, SETM був сконструйований з акцентом на використанні можливостей реляційних систем управління базами даних (RDBMS).

Алгоритм SETM інтегрує процес пошуку частих наборів елементів зі стандартними операціями реляційних баз даних.

Загальна схема роботи алгоритму:

- перетворення даних транзакції. Спочатку дані транзакції перетворюються у формат, придатний для обробки за допомогою SQL-запитів. Кожен елемент транзакції перетворюється в окремий запис, позначений ідентифікатором транзакції;
- генерація кандидатів на основі SQL-запитів. Алгоритм використовує SQL-запити для генерації наборів елементів-кандидатів. Це передбачає приєднання бази даних до себе і використання операцій групування та підрахунку для знаходження частот наборів елементів;

- відсікання та ітерації. Набори елементів, які не відповідають мінімальному порогу підтримки, відсікаються. Процес є ітеративним, де на кожній наступній ітерації набори елементів стають більшими на один елемент, а їхні частоти знову визначаються за допомогою SQL-запитів;
- виявлення частих наборів елементів. Алгоритм продовжується до тих пір, поки не буде згенеровано жодного нового частого набору елементів, таким чином завершуючи процес майнінгу.

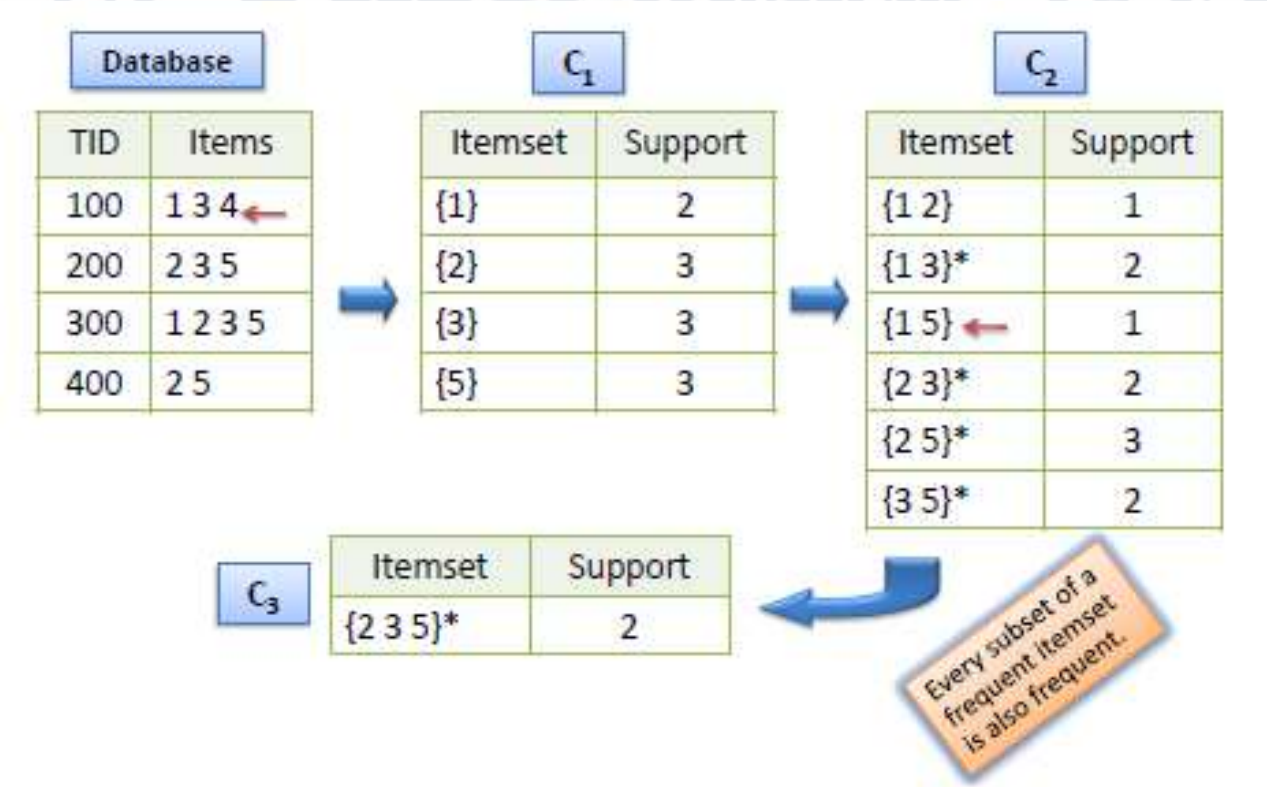


Рисунок 2.3 – Алгоритм SETM

SETM спеціально розроблений для використання можливостей СУБД для видобування наборів елементів, що часто зустрічаються. Він має на меті спростити процес, використовуючи стандартні операції SQL, що робить його привабливим вибором у середовищах, де реляційна база даних вже використовується для зберігання та управління даними [28].

Переваги:

- інтеграція з СКБД. використовує існуючі системи баз даних, уникаючи необхідності в спеціалізованих інструментах інтелектуального аналізу даних;

- робота на основі SQL. Використовує знайомі SQL-запити, що робить його доступним для користувачів з досвідом роботи з базами даних;
- ефективність у певних контекстах. Може бути ефективним у сценаріях, де система баз даних оптимізована під тип запитів, що використовуються SETM.

Недоліки:

- проблеми з масштабуванням. Як і попередні алгоритми, SETM може зіткнутися з проблемами продуктивності з дуже великими базами даних через широке використання операцій з'єднання;
- залежність від продуктивності бази даних. Ефективність алгоритму SETM значною мірою залежить від здатності системи баз даних обробляти складні SQL-запити та великі обсяги даних;
- потенційно інтенсивне навантаження на базу даних. повторювані операції з'єднання та групування можуть бути ресурсоємними для системи баз даних.

Алгоритм SETM представляє цікавий підхід до видобутку частих наборів елементів шляхом тісної інтеграції з технологіями реляційних баз даних. Хоча він пропонує певні переваги, особливо з точки зору використання існуючих систем на основі SQL, його продуктивність і масштабованість сильно залежать від можливостей базової системи баз даних. У сценаріях, де для управління даними вже використовується СУБД, SETM може бути практичним рішенням, хоча її використання може бути обмеженим у контекстах, де доступні більш ефективні, спеціалізовані інструменти інтелектуального аналізу даних.

2.2 Вибір алгоритму для методу пошуку асоціативних правил

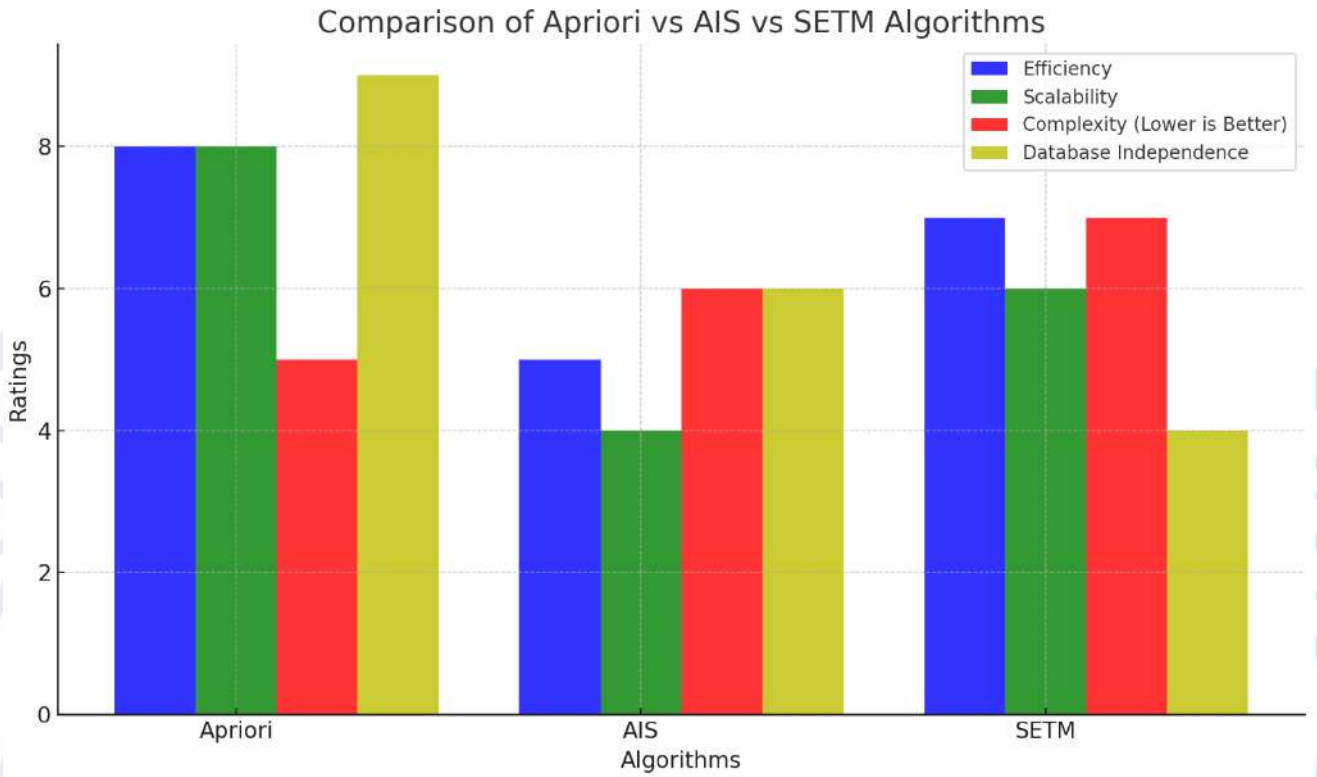


Рисунок 2.4 – Порівняння алгоритмів Apriori, AIS, SETM.

Порівняльна діаграма демонструє відмінності між алгоритмами Apriori, AIS і SETM, заснована на таких ключових характеристиках, як ефективність, масштабованість, складність і незалежність від бази даних. Ця таблиця відображає відносні сильні та слабкі сторони кожного алгоритму, ілюструючи. Можна зробити висновки, що Apriori може бути кращим у сценаріях, де ефективність, масштабованість та незалежність від бази даних є вирішальними факторами [29].

Для системи обробки та аналізу даних в медичній сфері за допомогою технологій Big Data було вирішено обрати алгоритм Apriori. Вибір алгоритму Apriori для системи обробки та аналізу медичних даних, особливо в порівнянні з такими альтернативами, як AIS та SETM, значною мірою зумовлений його ефективністю, масштабованістю та перевіреним досвідом роботи зі складними наборами даних. Apriori пропонує значне покращення порівняно з AIS з точки зору обчислювальної ефективності, що особливо важливо при роботі з великими та різноманітними даними, які зазвичай зустрічаються в охороні здоров'я. На відміну від AIS, яка може створювати громіздку кількість наборів елементів-

кандидатів, Apriori використовує ефективний механізм відсікання, що зменшує обчислювальне навантаження і робить його більш придатним для великих медичних баз даних.

Крім того, незалежність Apriori від продуктивності баз даних дає йому перевагу над SETM, який значною мірою покладається на специфічні можливості баз даних. Ця незалежність гарантує, що алгоритм Apriori може бути реалізований і ефективно функціонувати незалежно від використовуваної технології баз даних. Крім того, простота і прозорість алгоритму Apriori роблять його практичним вибором для системи аналізу медичних даних. Його зрозуміла природа і простота реалізації сприяють більш плавній розробці та усуненню несправностей, що є важливим для динамічної і критичної сфери аналізу даних в охороні здоров'я.

2.3 Модуль аналізу даних

Модуль аналізу даних системи обробки та аналізу медичних даних відіграє ключову роль у вилученні цінної інформації з оброблених даних. Цей модуль інтегрує складні аналітичні методи, включаючи алгоритм Apriori, для виявлення закономірностей і полегшення розширеного аналізу даних. При використанні алгоритму Apriori для пошуку асоціативних правил у системі, модуль в першу чергу буде працювати на основі двох ключових метрик: підтримки та впевненості. Ці метрики мають вирішальне значення для визначення значущості та надійності асоціативних правил, знайдених у наборі даних. Вони формулюються наступним чином.

Підтримка. Підтримка вимірює, як часто набір елементів з'являється в наборі даних. Це частка, яка вказує на присутність набору елементів у всіх транзакціях. Підтримка набору елементів обчислюється наступним чином:

$$\text{Support}(x) = \frac{\text{Кількість транзакцій,що містить X}}{\text{Загальна кількість транзакцій}}$$

Де X – множина елементів.

Для правил асоціацій підтримка обчислюється для комбінації елементів у правилі. Наприклад, для правила A⇒ B підтримка розраховується як:

$$\text{Support}(A \Rightarrow B) = \frac{\text{Кількість транзакцій,що містить A та } B}{\text{Загальна кількість транзакцій}}$$

У медичному контексті поняття довіри до видобутку асоціативних правил, наприклад, при використанні алгоритму Apriori, використовується для вимірювання сили імплікацій, знайдених в медичних даних. Розрахунок достовірності відбувається так само, як і в інших галузях, але інтерпретація адаптована до сфери охорони здоров'я.

Наприклад, розглянемо правило асоціації, отримане на основі даних про пацієнтів: Діабет ⇒ Високий кров’яний тиск. В даному випадку "Діабет" є попереднім (A), а "Високий кров'яний тиск" - наступним (B).

Підтримка цього правила обчислюється як:

$$\text{Confidence}(A \Rightarrow B) = \frac{Support(A \text{ and } B)}{Support(A)}$$

Підтримка (Діабет і високий кров'яний тиск) - частка пацієнтів у наборі даних, які мають і діабет, і високий кров'яний тиск.

Підтримка(Діабет) - відсоток пацієнтів у наборі даних, які мають діабет.

Алгоритм Apriori використовує ітеративний метод поетапного пошуку. Його основна ідея полягає у тому, що набір елементів рівня k будується на основі попередньо створених частих наборів елементів рівня k-1. Починаючи з генерації частого набору елементів P1, він використовує цей набір P1 для створення P2 (тобто частого набору з двох елементів) [10]. Потім P2 використовується для генерації P3 і так далі, доки не з'являються нові часті набори елементів рівня k.

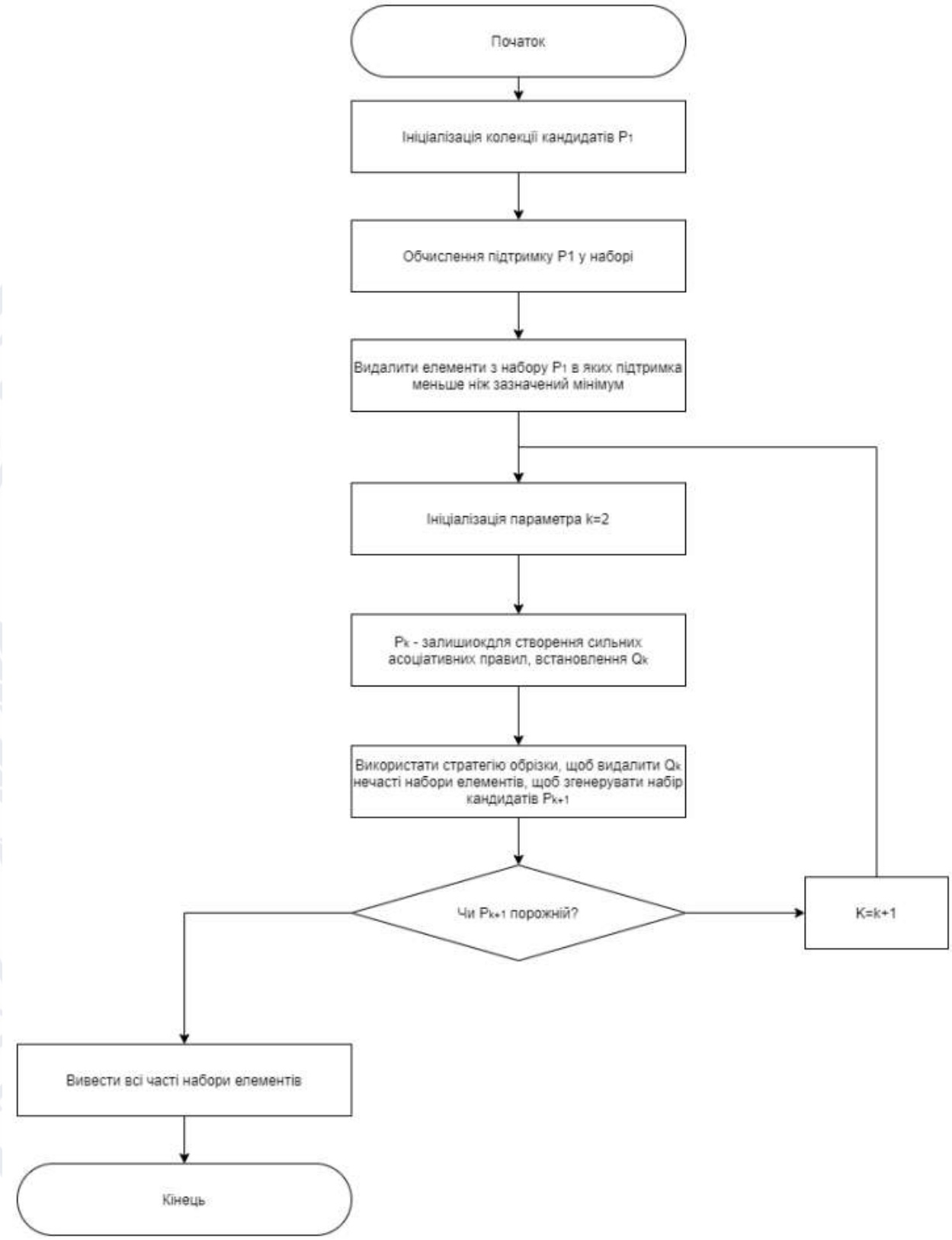


Рисунок 2.5 – Блок-схема алгоритму пошуку асоціативних правил.

Алгоритм Apriori спочатку визначає часті набори елементів у наборі даних. Це комбінації елементів (наприклад, симптомів, ліків або характеристик пацієнта), які зустрічаються разом частіше, ніж певний поріг. З цих наборів

елементів, що часто зустрічаються, алгоритм генерує правила асоціацій. Ці правила вказують на взаємозв'язки, такі як певна комбінація симптомів, що часто призводить до певного діагнозу. Кожне правило оцінюється на основі його підтримки (як часто воно з'являється в наборі даних) і достовірності (як часто воно відповідає дійсності). Це допомагає визначити силу та надійність асоціацій.

2.4 Модуль збору та зберігання даних

Модуль збору та зберігання даних є важливим компонентом системи обробки та аналізу медичних даних. Він відповідає за первинний збір, обробку та безпечне зберігання медичних даних. Основні його компоненти:

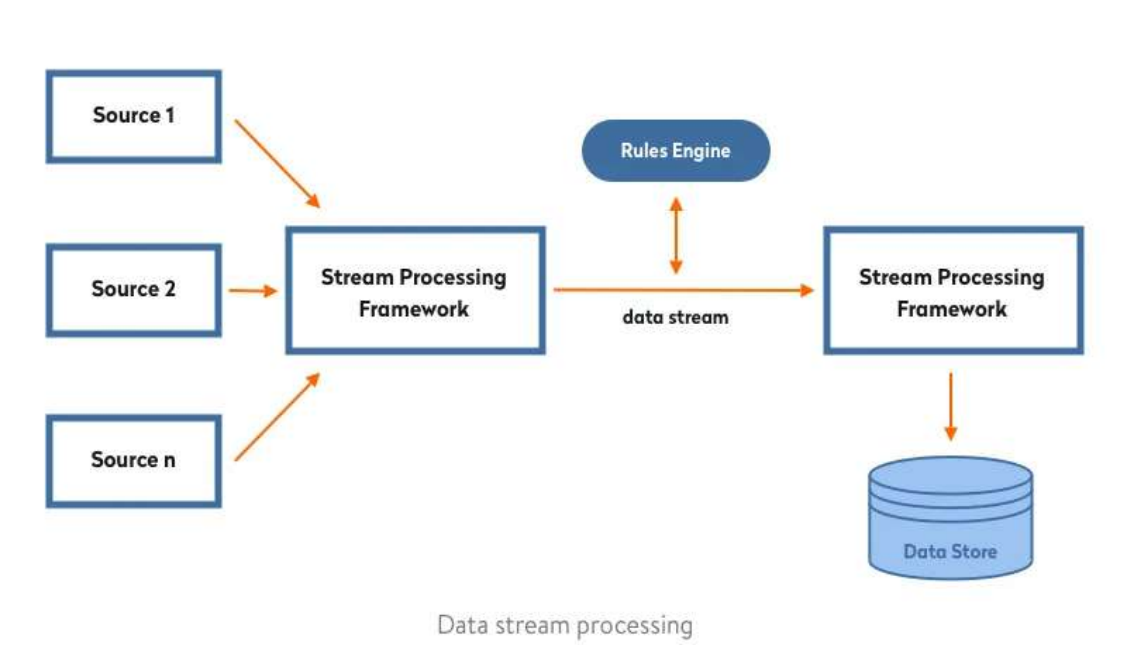


Рисунок 2.6 – Схема обробки потоку даних.

Компонента збору даних. Можливість завантажувати великі датасети та дані в різних форматах (JSON, XML, CSV, тощо).

- інтерфейсні адаптери: Індивідуальні адаптери для різних джерел даних для роботи з різними форматами даних і протоколами;
- обробка в реальному часі та пакетна обробка: Підтримує як отримання даних в режимі реального часу для негайного аналізу (наприклад, з пристроїв моніторингу пацієнта), так і пакетну обробку історичних даних(наприклад із бази даних);

- перевірка та очищення даних: Початкова обробка для перевірки, очищення та попередньої обробки даних для забезпечення точності та узгодженості;
- нормалізація та стандартизація: Перетворення даних в узгоджений формат, дотримуючись стандартів медичних даних.

Модуль зберігання даних:

- реляційні та NoSQL бази даних: Використовує комбінацію реляційних баз даних (для структурованих даних) і баз даних NoSQL (для неструктурованих або напівструктурованих даних) для ефективного зберігання різноманітних даних;
- сховища даних: Реалізує сховище даних для консолідації різноманітних даних для полегшення доступу та аналізу;
- шифрування та захист даних: Впроваджує надійні протоколи шифрування даних у стані запису та читання;
- зберігання метаданих: Зберігає метадані разом з даними, що має вирішальне значення для категоризації, пошуку та управління даними.

Висновки до розділу

В даному розділі було розглянуто поняття методів пошуку асоціативних правил та їх застосування в даній роботі. Проведено аналіз різних алгоритмів задля вирішення цієї задачі. Описано та проілюстровано схеми роботи основних компонентів системи.

РОЗДІЛ 3
РОЗРОБКА ТА АНАЛІЗ СИСТЕМИ

Вибір правильного технологічного стеку є життєво важливим для успіху реалізації будь-якого проекту, насамперед тому, що обрані технології визначатимуть функціонування системи, її ефективність, здатність до адаптації та масштабування з плином часу. Правильно підібраний стек технологій гарантує, що система не тільки відповідає поточним функціональним вимогам, але й здатна впоратися з майбутніми викликами і розширеннями.

Добре підібраний стек технологій гарантує функціональну адекватність. Це означає, що технологія ідеально відповідає специфічним потребам обробки, зберігання та аналізу великих обсягів складних медичних даних. Це гарантує, що система працює на необхідному рівні продуктивності, легко і точно виконуючи такі завдання, як аналіз великих даних, обробка даних в режимі реального часу і складні обчислення, що особливо важливо в такій чутливій сфері, як охорона здоров'я.

3.1 Використані технології

Платформа .NET

Платформа .NET — це безкоштовна платформа для створення програмних продуктів з відкритим кодом, що дозволяє створювати різні типи додатків. З .NET можна використовувати кілька мов, редакторів і бібліотек для створення веб-, мобільних, настільних додатків, ігор, додатків у сфері машинного навчання [30].

Одним з найпомітніших досягнень .NET є кросплатформеність, особливо з появою .NET Core. Ця розробка гарантує, що додатки, створені на платформі .NET, можуть безперешкодно працювати на різних операційних системах, включаючи Windows, Linux та macOS. Різноманітні бібліотеки та фреймворки .NET, такі як ASP.NET для веб-додатків, Entity Framework для доступу до даних та ML.NET для машинного навчання, надають комплексний інструментарій, який спрощує процес розробки. Ці бібліотеки пропонують вбудовані

функціональні можливості, що зменшують обсяг кодування та підвищують ефективність розробки.

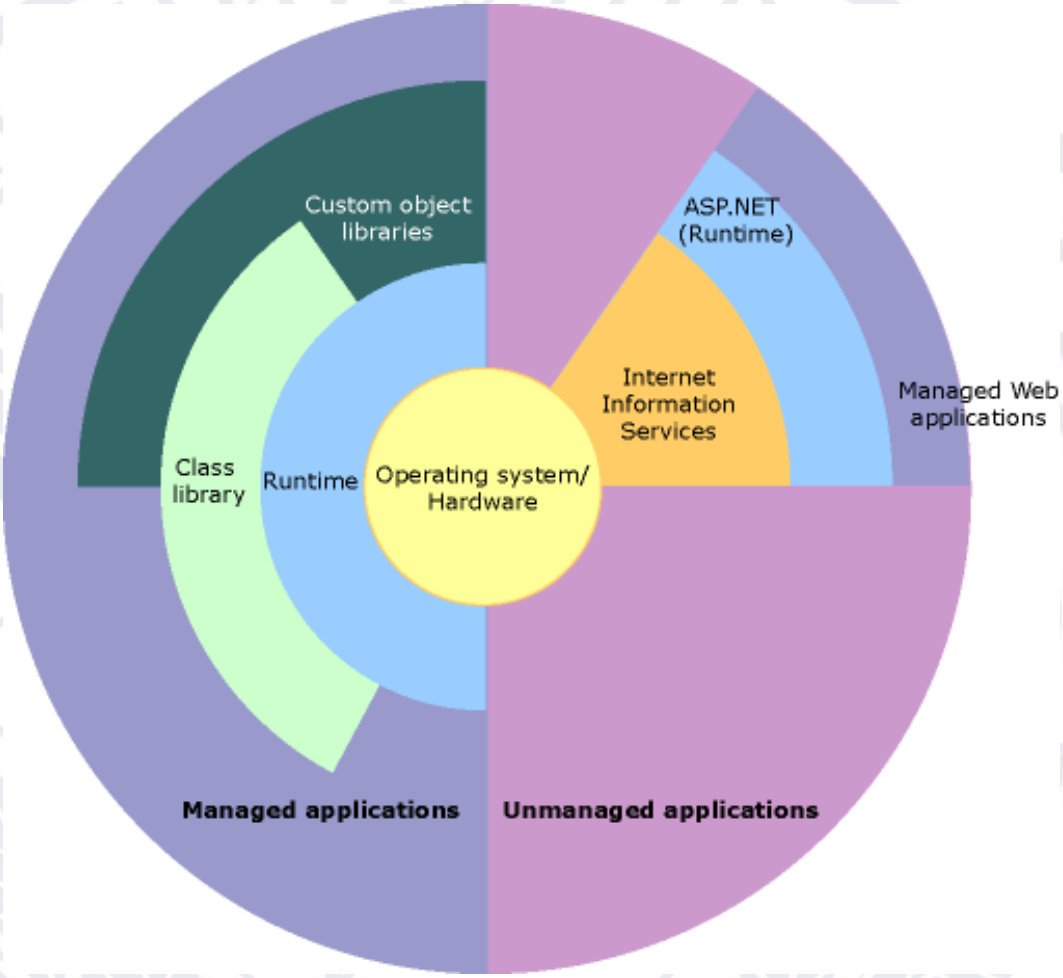


Рисунок 3.1 – Структура .NET

Продуктивність та масштабованість лежать в основі дизайну .NET. Відомий своєю високою продуктивністю, фреймворк вміє ефективно обробляти великі обсяги даних. Крім того, додатки .NET можна масштабувати як горизонтально, так і вертикально, пропонуючи гнучкість, щоб пристосуватися до зростання системи з точки зору обсягу даних і користувацького трафіку. Інтегроване середовище розробки (IDE) .NET, в першу чергу Microsoft Visual Studio, збагачує досвід розробки завдяки своїм розширеним можливостям, інструментам налагодження та інтегрованим засобам тестування. Потужна підтримка спільноти та обширна документація, доступна для .NET, також

надають цінну підтримку, роблячи вирішення проблем та навчання більш доступним для розробників [31].

Для створення додатків на цій платформі можна використовувати одну із трьох підтримуваних мов програмування:

Рисунок 3.2 – Можливості платформи .NET

- C# — це проста, сучасна, об’єктно-орієнтована та безпечна для типів мова програмування;
- F# — це мова програмування, яка дозволяє легко писати стислий, надійний і продуктивний код у функціональному стилі;
- Visual Basic – це доступна мова з простим синтаксисом для створення безпечних для типів об’єктно-орієнтованих програм.

Поєднання продуктивності, гнучкості, безпеки та підтримки спільноти робить .NET зразковим вибором для розробки складної та надійної системи в секторі охорони здоров'я, здатної задовольнити динамічні та критичні вимоги до обробки та аналізу медичних даних.

Технологія ASP.NET Core

ASP.NET (Active Server Pages для .NET) — платформа розробки веб-застосунків, до складу якої входять: веб-сервіси, програмна інфраструктура, модель програмування, від компанії Майкрософт. ASP.NET входить до складу

платформи .NET і є логічним продовженням старої технології Microsoft ASP. ASP.NET Core характеризується розширюваністю. Фреймворк побудований із набору відносно незалежних компонентів. І можна або використати вбудовану реалізацію цих компонентів, або розширити їх за допомогою механізму наслідування, або створити і застосовувати свої компоненти зі своїм функціоналом [32].

Моделі розробки веб-додатків за допомогою ASP.NET Core[13]:

- базовий ASP.NET Core, який підтримує всі основні моменти, необхідні для роботи сучасного веб-додатку: маршрутизація, конфігурація, логування, можливість роботи з різними системами баз даних і т.д.;
- ASP.NET Core MVC представляє у загальному вигляді побудову програми навколо трьох основних компонентів - Model (моделі), View (представлення) та Controller (контролери), де моделі відповідають за роботу з даними, контролери представляють логіку обробки запитів, а представлення визначають візуальну складову;
- Razor Pages представляє модель, у якій за обробку запиту відповідають спеціальні сутності - сторінки Razor Pages. Кожну окрему таку сутність можна порівнювати з окремою веб-сторінкою;
- ASP.NET Core Web API - представляє реалізацію патерну REST, у якому кожному типу http-запиту (GET, POST, PUT, DELETE) призначений окремий ресурс. Такі ресурси визначаються як методи контролера Web API;
- Blazor представляє фреймворк, який дозволяє створювати інтерактивні програми як на стороні сервера, так і на стороні клієнта та дозволяє задіяти на рівні браузера низькорівневий код WebAssembly.

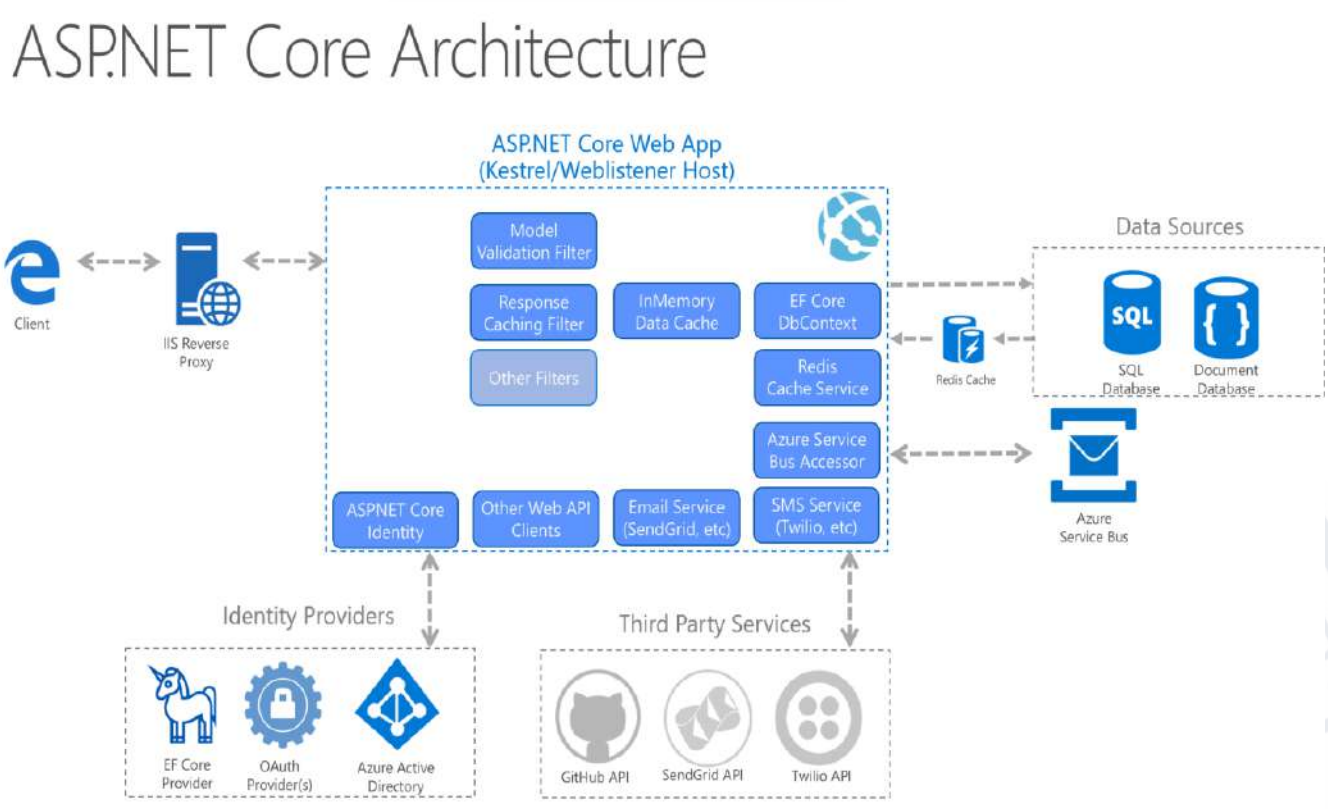


Рисунок 3.3 – Архітектура ASP.NET Core додатку [33]

На сьогоднішній день однією із найпопулярніших моделей створення веб-додатків за допомогою ASP.NET Core є ASP.NET Core Web API, її основні переваги:

- реалізація патерну REST;
- забезпечує достатню гнучкість у створенні веб-API за рахунок узгодження вмісту та забезпечує підтримку маршрутизації ASP.NET;
- архітектура веб-API дуже легка, що робить її ідеальною альтернативою для розробників, що хочуть створювати програми для пристроїв з обмеженою пропускною здатністю;
- використання клієнт-серверного підходу;
- підтримує шаблони URL-адрес і методи HTTP;
- підтримує прив'язку моделі та її валідацію;
- кросплатформеність: можливість розробки та розгортання додатків на Windows, Mac, Linux.

- Вбудована підтримка впровадження залежностей

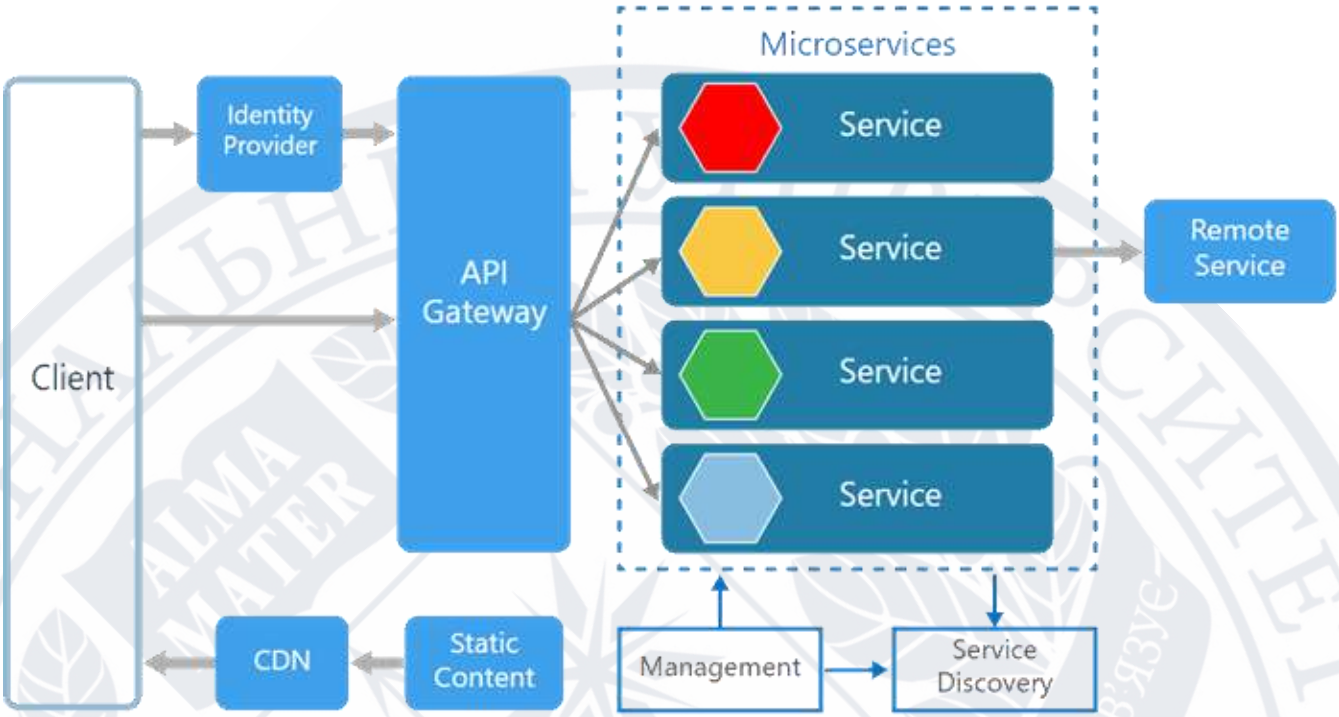


Рисунок 3.4 – Архітектура ASP.NET Core Web Api додатку [34]

Ключові переваги ASP.NET Core [35]:

- крос-платформеність. Однією з відмінних рис ASP.NET Core є його крос-платформеннісьб. На відміну від свого попередника, він може працювати на Windows, Linux та macOS, що робить додатки, розроблені з його допомогою, більш універсальними та адаптованими до різних серверних середовищ;
- продуктивність. ASP.NET Core відомий своєю підвищеною продуктивністю. Середовище було оптимізоване, щоб бути швидшим та ефективнішим, що є критично важливим для додатків, які обробляють великі обсяги даних або вимагають швидкого часу відгуку;
- модульність і гнучкістью Завдяки більш модульній структурі, ASP.NET Core дозволяє розробникам включати в свої додатки тільки необхідні компоненти, зменшуючи накладні витрати і підвищуючи продуктивність. Ця модульність також означає більшу гнучкість у виборі бібліотек та інструментів, що дозволяє застосовувати більш індивідуальний підхід до розробки;

- підтримка архітектури мікросервісів. ASP.NET Core добре підходить для створення додатків з використанням архітектури мікросервісів - підходу, який стає все більш популярним для створення масштабованих, складних додатків. Кожен сервіс можна розробляти, розгортати та масштабувати незалежно, що є перевагою для великих, багатогранних систем, таких як системи охорони здоров'я;
- покращені функції безпеки. Безпека є першочерговим завданням для додатків у сфері охорони здоров'я. ASP.NET Core забезпечує розширені функції безпеки та відповідність галузевим стандартам, гарантуючи захист конфіденційних даних пацієнтів;
- надійна екосистема та підтримка спільноти. Будучи частиною екосистеми .NET, ASP.NET Core користується перевагами великої спільноти розробників і широкого спектру доступних бібліотек та інструментів. Ця підтримка спільноти є безцінною для усунення несправностей та підвищення ефективності розробки.

Виклики та обмеження

- крива навчання. Для розробників, які звикли до старого фреймворку ASP.NET, перехід на ASP.NET Core може бути пов'язаний з певними труднощами через відмінності в структурі та роботі;
- швидкі зміни та оновлення. ASP.NET Core часто оновлюється новими функціями та покращеннями. Хоча це гарантує, що фреймворк залишається сучасним та ефективним, це також може створити проблеми з відстеженням останніх змін, особливо у великих проектах з довгими циклами розробки;
- обмежена підтримка сторонніх бібліотек. Незважаючи на те, що кількість сторонніх бібліотек та інструментів постійно зростає, деякі з них ще не були портовані або оптимізовані для ASP.NET Core. Це може обмежити можливості в певних сценаріях розробки.

Технологія MongoDB

MongoDB є однією з найпопулярніших систем керування базами даних (СКБД) типу NoSQL. Він використовує модель документа, де дані зберігаються у вигляді документів у форматі BSON (Binary JSON). MongoDB дозволяє зберігати та опрацьовувати великі обсяги даних швидко та ефективно [36].

Основні характеристики MongoDB [37]:

- гнучка схема даних: MongoDB не вимагає жорсткого визначення схеми даних перед збереженням. Ви можете зберігати різнорідні дані у вигляді документів з різною структурою, що дозволяє легко змінювати схему та додавати нові поля без необхідності перетворення всієї бази даних;
- горизонтальне масштабування: MongoDB дозволяє горизонтально масштабувати базу даних шляхом розподілу даних на різні сервери (шардування). Це дозволяє обробляти великі навантаження та забезпечує високу доступність даних;
- можливості запитів: MongoDB надає потужний механізм запитів, включаючи запити на основі об'єктів, запити з умовами, текстовий пошук та агрегаційні запити. Він також підтримує індексування для прискорення пошуку та оптимізації запитів;
- розширені можливості: MongoDB має багато додаткових можливостей, таких як реплікація для забезпечення надійності даних, механізм транзакцій для атомарних операцій, геопросторові запити для роботи з географічними даними та інші.

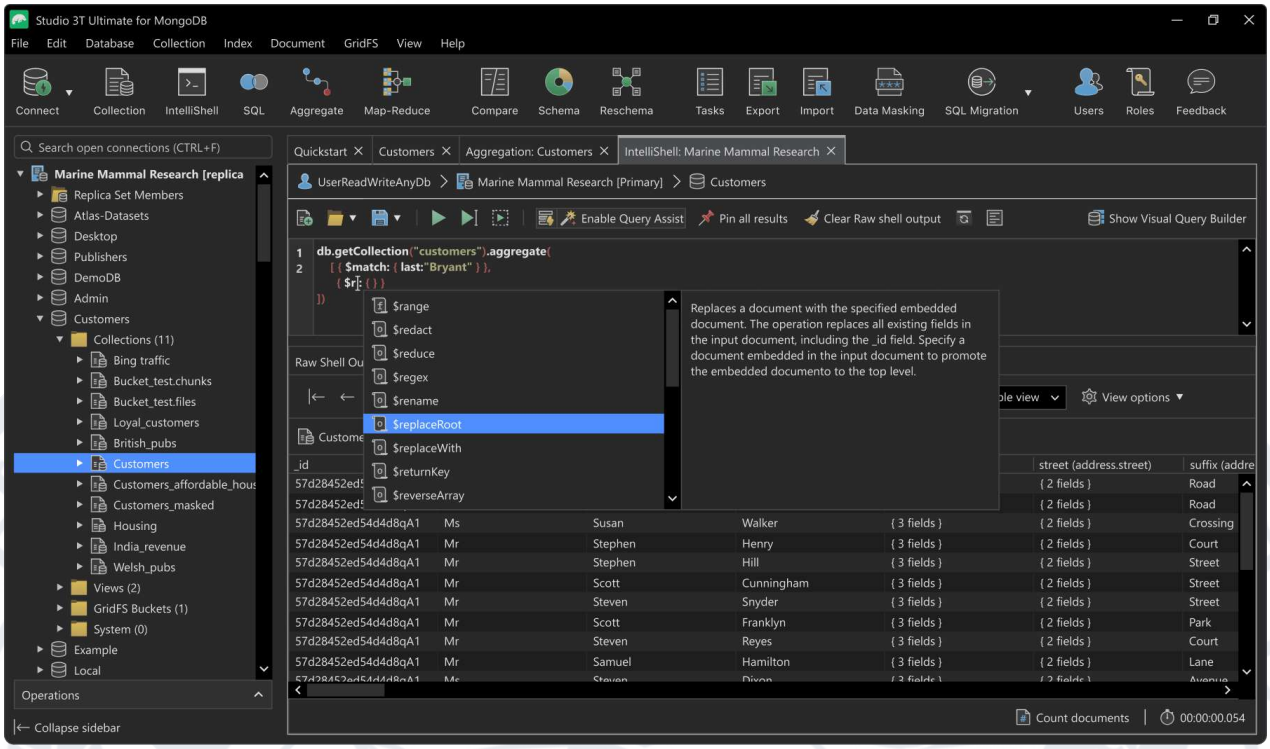


Рисунок 3.5 – IDE MongoDB

Сфери використання MongoDB [38]:

- веб-додатки: MongoDB є популярним вибором для зберігання даних веб-додатків, зокрема соціальних мереж, систем управління вмістом, електронної комерції та інших веб-орієнтованих додатків. Він забезпечує швидкий доступ до даних та можливість масштабування для високого навантаження;
- аналітика даних: MongoDB може використовуватися для зберігання та аналізу великих обсягів даних, включаючи дані журналів, логів, метрик та іншої неструктурованої інформації. Він дозволяє виконувати складні аналітичні запити та витягувати цінні інсайти з даних;
- інтернет речей (IoT): MongoDB використовується для зберігання та обробки даних, зібраних від підключених до Інтернету пристроїв. Він дозволяє легко зберігати та аналізувати великі обсяги даних, що надходять з сенсорів та пристроїв IoT;
- телекомунікації: MongoDB використовується в галузі телекомунікацій для зберігання абонентських даних, журналів викликів, статистики

мережі та іншої телекомунікаційної інформації. Він забезпечує швидкий доступ до даних та можливість реал-тайм аналітики.

MongoDB використовує формат, відомий як BSON (binary JSON), для зберігання даних. Це відрізняється від реляційних баз даних, оскільки MongoDB дозволяє зберігати різні об'єкти, незалежно від їх властивостей та структури. MongoDB є документо-орієнтованою базою даних, що означає, що вона зберігає дані у вигляді документів, а не рядків в таблицях, як це зазвичай відбувається. Ці документи зберігаються в колекціях, де кожен документ представляє собою сховище пар ключ-значення.

Дані можуть зберігатися в такому вигляді:


```
{
  "_id": "KU+rcKy4QiS4CE",
  "_t": [
    "FormBase",
    "DynamicForm"
  ],
  "IsDeleted": false,
  "name": "Starlor",
  "lastModified": "2023-05-16T08:53:54.799Z",
  "isValid": false,
  "type": 1,
  "isMultiEntry": false,
  "controlSets": [],
  "controls": [
    {
      "_t": [
        "Control",
        "TextBox"
      ],
      "_id": "Uy7bpRyrztwR+s2ANF2g",
      "IsDeleted": false,
      "type": "textbox",
      "label": "Text field",
      "validationRules": [
        {
          "type": 0,
          "value": null,
          "message": "This field is required."
        },
```

```
        {
          "type": 1,
          "value": null,
          "message": null
        }
      ],
      "value": "qwe"
    },
    {
      "_t": [
        "Control",
        "CheckBoxGroup"
      ],
      "_id": "9lY8lXjP3mDk",
      "IsDeleted": false,
      "type": "checkbox",
      "label": "Checkbox list",
      "validationRules": [],
      "options": [
        {
          "name": "qwe",
          "value": "qwe",
          "isChecked": false
        }
      ],
      "values": []
    }
  ]
}
```


Рис 3.6 - Приклад моделі даних

Microsoft SQL Server

Microsoft SQL Server(MSSQL Server), - це реляційна система керування базами даних (СКБД), розроблена компанією Microsoft. Як сервер баз даних, його основна функція полягає у зберіганні та отриманні даних на запит інших програмних додатків, які можуть працювати як на тому ж комп'ютері, так і на іншому комп'ютері в мережі. SQL Server, подібно до інших технологій СУБД, оснований на реляційній моделі, де таблиці зі строковою структурою використовуються для зв'язування взаємопов'язаних даних із різних таблиць. Це

дозволяє уникнути повторного зберігання однакової інформації у багатьох місцях. Реляційна модель також підтримує посилальну цілісність та інші види обмежень цілісності, які забезпечують точність даних [39]. Ці обмеження є частиною принципів атомарності, узгодженості, ізоляції та довговічності, відомих як властивості ACID, які забезпечують надійну обробку транзакцій.

Двигун бази даних SQL Server від Microsoft є ключовим елементом SQL Server. Він відповідає за збереження, обробку даних та їх безпеку. Цей компонент включає реляційний механізм, який виконує команди та запити, а також механізм зберігання, який управляє файлами баз даних, сторінками, таблицями, індексами, буферами даних і транзакціями.

Процес роботи SQL Database Engine, який є частиною Microsoft SQL Server, можна розділити на кілька основних етапів [40]:

- парсинг та Аналіз Запиту (Query Parsing and Analysis): Коли SQL Server отримує запит, він спочатку розбирає його, перевіряючи синтаксис і переконуючись, що всі назви об’єктів відомі та існують;
- оптимізація Запиту (Query Optimization): Після аналізу запиту двигун бази даних визначає найбільш ефективний спосіб виконання запиту. Це включає вибір найкращих індексів для використання, визначення порядку виконання операцій тощо;
- генерація Плану Виконання (Execution Plan Generation): На основі оптимізації формується план виконання, який є докладним описом кроків, необхідних для виконання запиту;
- виконання Запиту (Query Execution): Виконання запиту згідно з сформованим планом. Це включає зчитування даних з бази, виконання обчислень, об’єднання даних тощо;
- обробка Транзакцій (Transaction Processing): Управління транзакціями, яке забезпечує атомарність, узгодженість, ізоляцію та довговічність (властивості ACID);

- зберігання Даних (Data Storage Management): Включає управління файлами баз даних, сторінками, таблицями, індексами, буферами даних;
- забезпечення Безпеки (Security Enforcement): Забезпечення безпеки та контроль доступу до даних, а також здійснення політик аудиту та дотримання стандартів.

Кожен з цих етапів є критично важливим для ефективного та безпечного управління даними в SQL Server Database Engine.

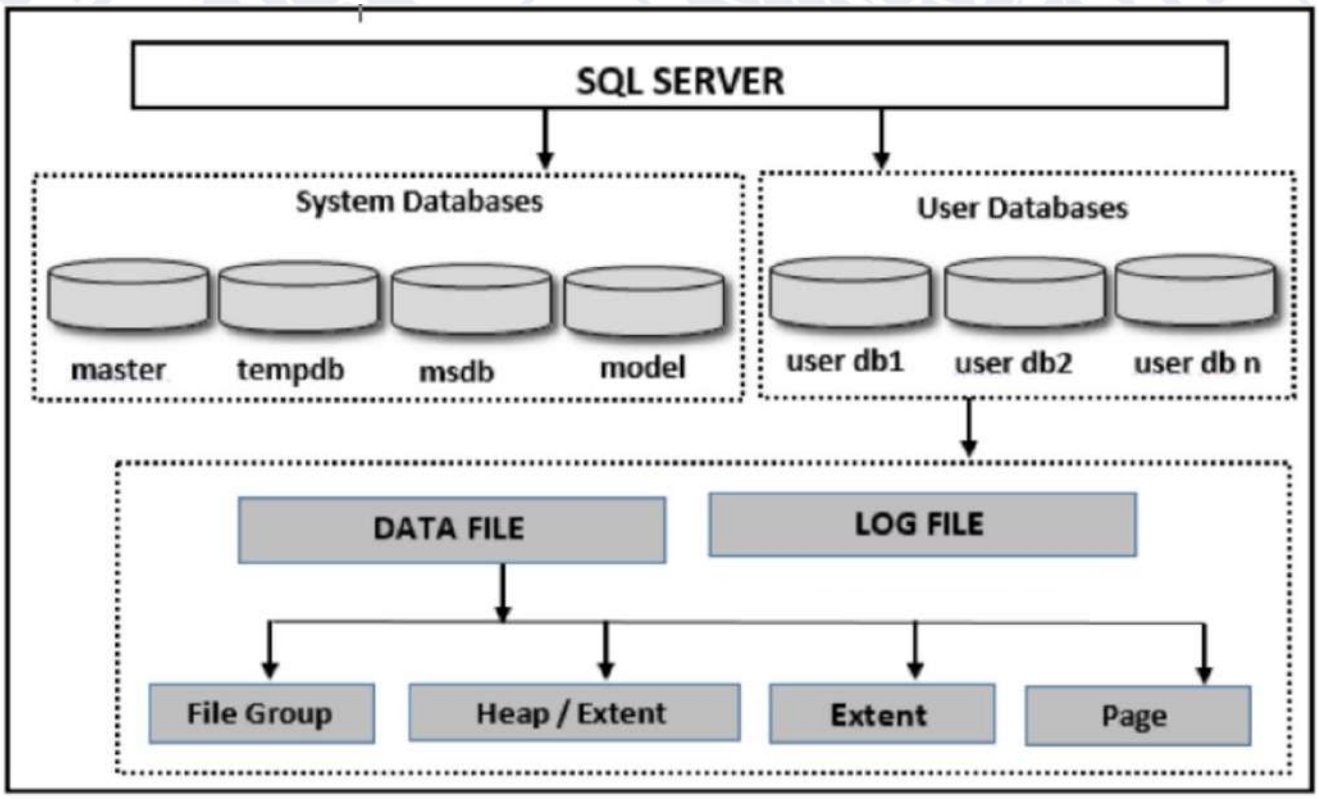


Рисунок 3.7 – Схема роботи MSSQL Engine

3.2 Архітектура Back-End

Multilayered Architecture (Багатошарова архітектура)

Багаторівнева архітектура додатків .NET - це шаблон проектування, який організовує додаток в окремі шари, кожен з яких має певні обов'язки та ролі. Ця архітектура спрямована на розділення завдань, тобто різні аспекти програми фізично відокремлені, але безперешкодно працюють разом [41]. У контексті

.NET ця архітектура зазвичай передбачає поділ додатку щонайменше на три рівні: Представлення, бізнес-логіка та доступ до даних.

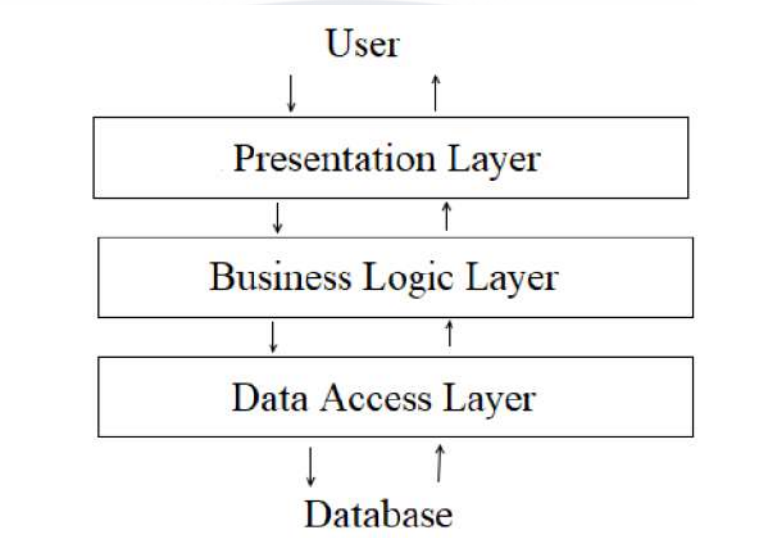


Рисунок 3.8 – Схема багаторівневої архітектури

Порівнева сегрегація дозволяє відповідно керувати та підтримувати кожен рівень. Теоретично це повинно значно спростити спосіб управління програмною інфраструктурою. Багаторівневий підхід особливо підходить для розробки веб-додатків та хмарних додатків. Це також полегшує оновлення будь-яких застарілих систем – коли архітектура розбита на кілька шарів, зміни, які потрібно внести, будуть простішими та менш масштабними, ніж вони могли б бути в іншому випадку. Багаторівнева архітектура також дозволяє, при потребі, розділити програмні компоненти на окремих серверах. Якщо розглянута система вимагає швидшого мережевого зв’язку, високої надійності та високої продуктивності, то n-tier має можливість забезпечити це, оскільки цей архітектурний шаблон призначений для зменшення накладних витрат, спричинених мережевим трафіком [42].

Архітектура додатку розділена на чотири логічні рівні згідно багаторівневої архітектури:

1. Domain Layer – рівень доступу до даних, де зберігаються моделі, описують використовувані сутності, а також розміщуються специфічні класи для роботи з різними технологіями доступу до даних, Цей рівень обробляє всі взаємодії з джерелом даних/базою даних. Його обов'язки включають пошук, вставку, оновлення та видалення даних. Часто

реалізується за допомогою Entity Framework for ORM (об'єктно-реляційне відображення) для взаємодії з SQL Server або іншими базами даних.

2. Core Layer – Цей рівень містить основну функціональність системи. Він обробляє вхідні дані користувача, застосовує бізнес-правила та приймає рішення. Не залежить від користувацького інтерфейсу, тобто одна і та ж бізнес-логіка може обслуговувати різні типи клієнтів (веб-, десктопні, мобільні)..
3. Core Infrastructure Layer – рівень спільних модулів, який включає в себе інтерфейси(контракти) основних репозиторіїв та сервісів, спільні моделі, налаштування. Допомагає зробити код слабозв’язаним та з легкістю підключати інші компоненти або взаємозаміняти їх. Рівень представлення не може безпосередньо отримувати дані з бази даних.В даному випадку BLL виступатиме в ролі посередника між двома рівнями.
4. Web Api Layer – фактичний інтерфейс, через який клієнти можуть працювати з API, реалізовано через ASP.NET Core. Конфігурації маршрутів визначаються атрибутами. Цей рівень побудовано за архітектурним принципом REST.

Нижче наведена схема архітектури додатку з її основними компонентами:

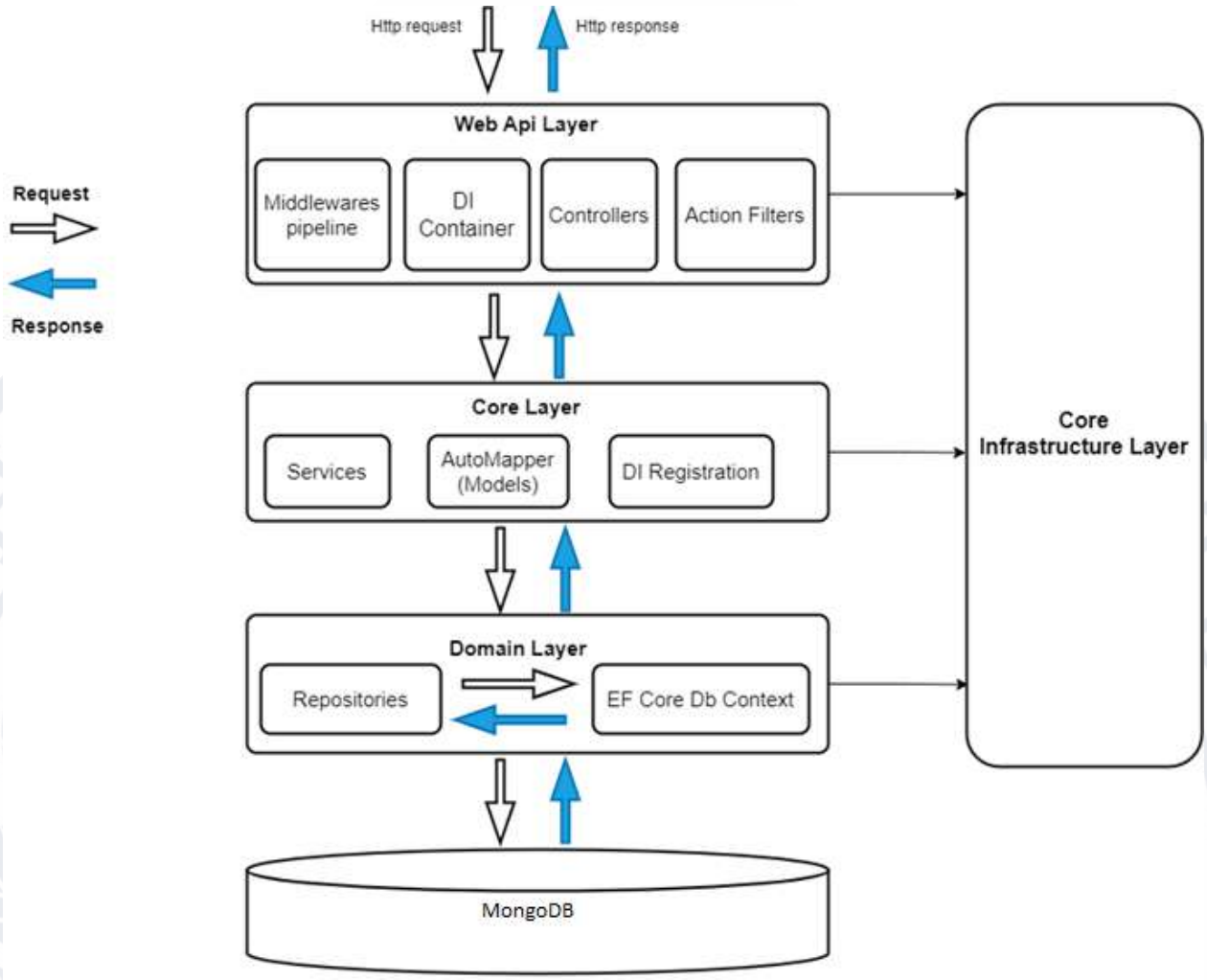


Рисунок 3.9 – Архітектура одного з компонентів додатку

Переваги багаторівневої архітектури:

- легкість супроводу. Зміни в одному шарі (наприклад, оновлення інтерфейсу користувача) зазвичай не впливають на інші шари, що полегшує підтримку програми;
- масштабованість. Кожен рівень можна масштабувати незалежно за потреби;
- повторне використання. Бізнес-логіку можна повторно використовувати на різних рівнях презентації або в різних додатках;
- тестування. Поділ завдань полегшує тестування окремих компонентів (наприклад, модульне тестування бізнес-логіки, не турбуючись про інтерфейс користувача);

Недоліки

- складність. Ця архітектура може вносити додаткову складність і може вимагати більш ретельного проектування та планування;
- накладні витрати на продуктивність. Додаткові шари можуть означати більше обчислень і передачі даних, що потенційно впливає на продуктивність.

Отже, багаторівнева архітектура додатків .NET - це підхід до проектування, який розділяє додаток на окремі шари, кожен з яких відповідає за певний аспект роботи програми. Такий поділ покращує зручність обслуговування, масштабованість і тестування, але може призвести до ускладнення та потенційного зниження продуктивності.

Аутентифікація та авторизація

Для реалізації механізмів аутентифікації/авторизації використовується інструмент ASP.NET Core Identity та JWT-токени. Стандартна база даних була розширена за допомогою засобів ASP.NET Core Identity[43]:

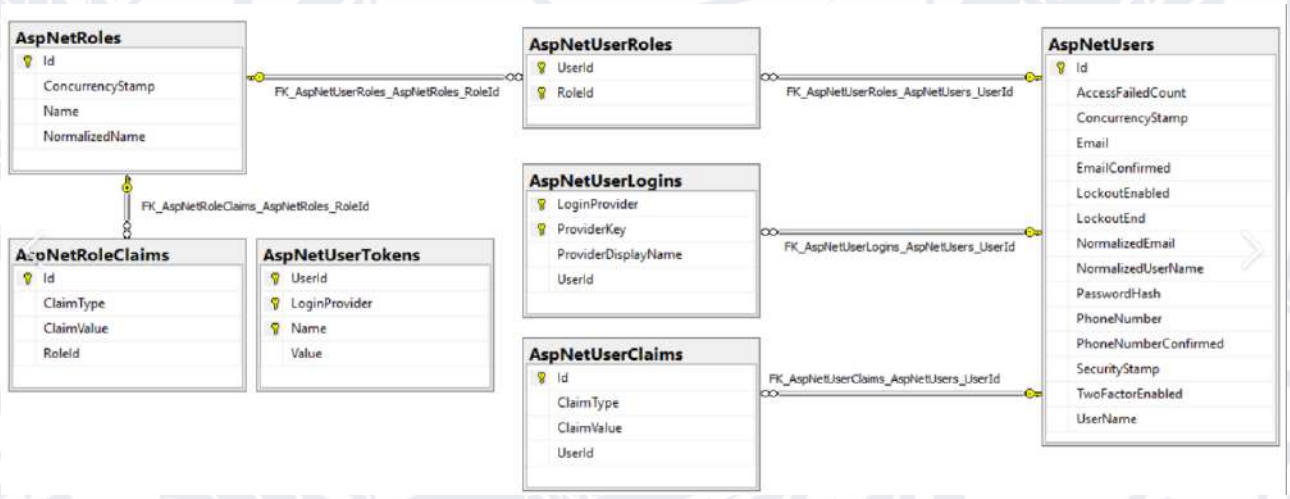


Рисунок 3.10 – Схема бази даних для аутентифікації/авторизації користувачів

У файлі appsettings.json описані рядки підключення до баз даних та налаштування для створення та видачі сервером JWT-токенів.

```
"ConnectionStrings": {
  "DefaultConnection": "Server=(localdb)\\mssqllocaldb;Database=WebForumDb;Trusted_Connection=True;MultipleActiveResultSets=true",
  "WebForumProd": "Server=tcp:webforumdb.database.windows.net,1433;Initial Catalog=webforumdb;Persist Security Info=False;User ID=webforumdb
},
"AllowedHosts": "*",
"JWTSettings": {
  "securityKey": "WebForumSuperSecretKey12",
  "validIssuer": "WebForumApi",
  "validAudience": "WebForumApi",
  "expiryInMinutes": 180
}
}
```

Рисунок 3.10 – налаштування в файлі appsettings.json

Такий підхід дає змогу тримати всі необхідні налаштування в одному місці та при потребі легко змінювати їх. Наприклад, секція «JWTSettings» містить рядки:

- securityKey – прихований ключ за допомогою якого сервер шифрує токени та при вхідних запитах перевіряє їх валідність
- validIssuer – вказує на того, хто видає токени
- validAudience – вказує на того хто приймає токени
- expiryInMinutes – задає час життя токену у хвилинах

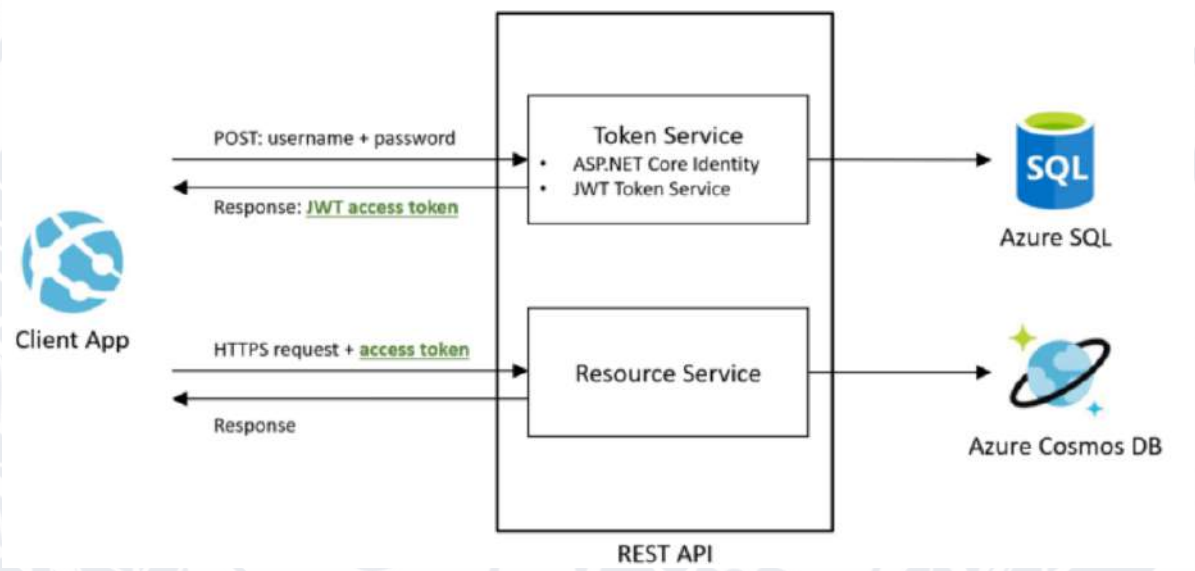


Рисунок 3.11 – Процес авторизації/аутентифікації за допомогою JWT

Принцип роботи сервісу аутентифікалії/авторизації [44]:

- програма або клієнт запитує авторизацію до сервера авторизації. Це виконується за допомогою одного з різних потоків авторизації. Наприклад, типова веб-програма, сумісна з OpenID Connect, буде проходити через кінцеву точку /oauth/authorize за допомогою потоку коду авторизації;
- коли авторизація надається, сервер авторизації повертає токен доступу до програми;

- програма використовує маркер доступу для доступу до захищеного ресурсу (наприклад, API).

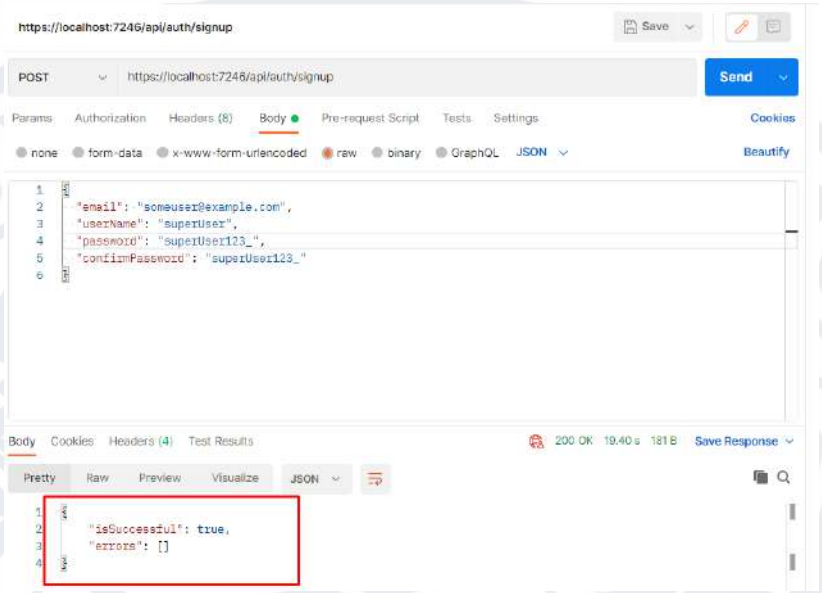


Рисунок 3.12– Приклад запиту на реєстрацію нового користувача

3.3 Демонстрація роботи додатку

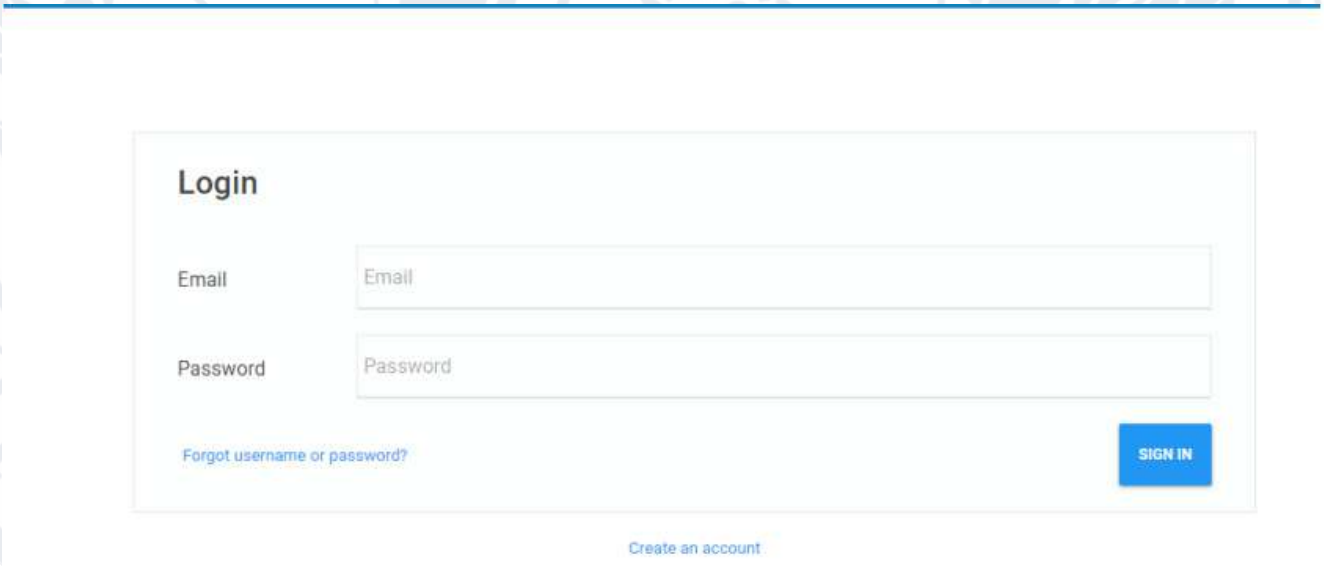


Рисунок 3.13 – сторінка входу

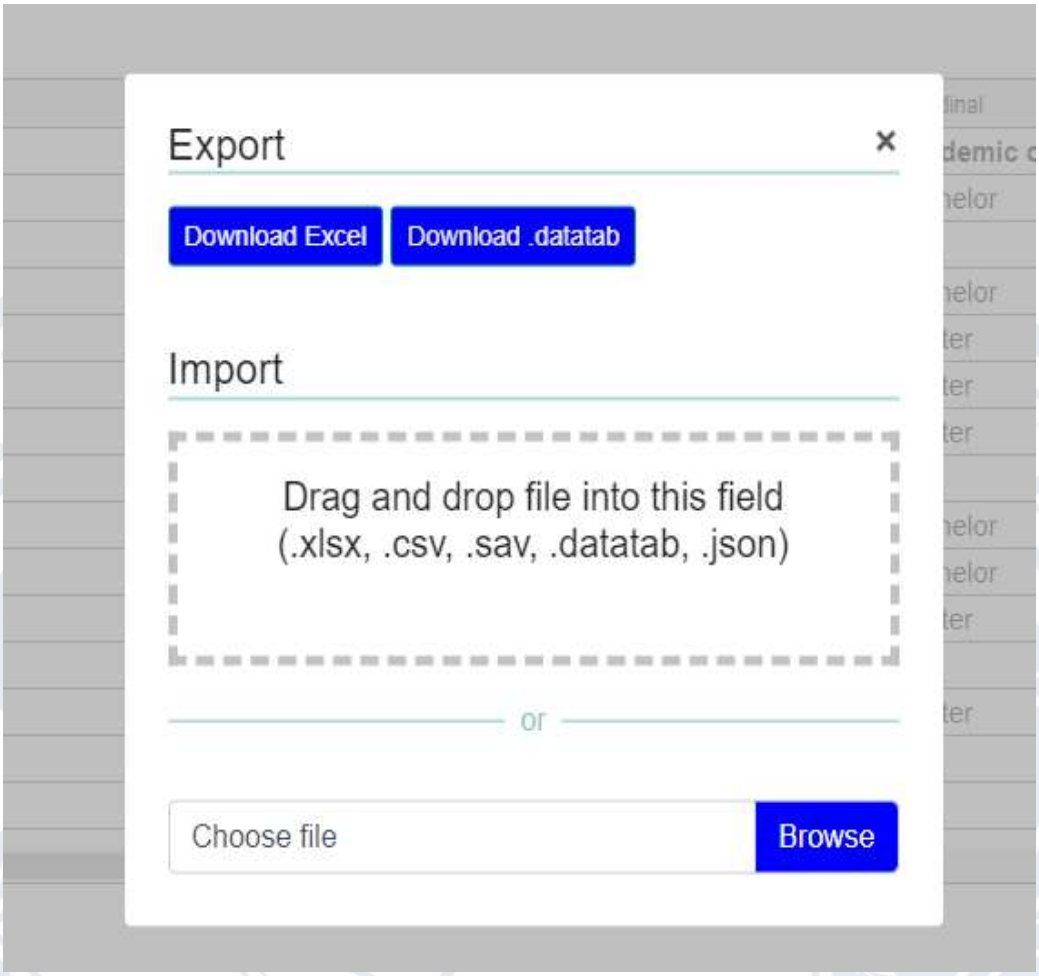


Рисунок 3.14 – Меню завантаження файлів

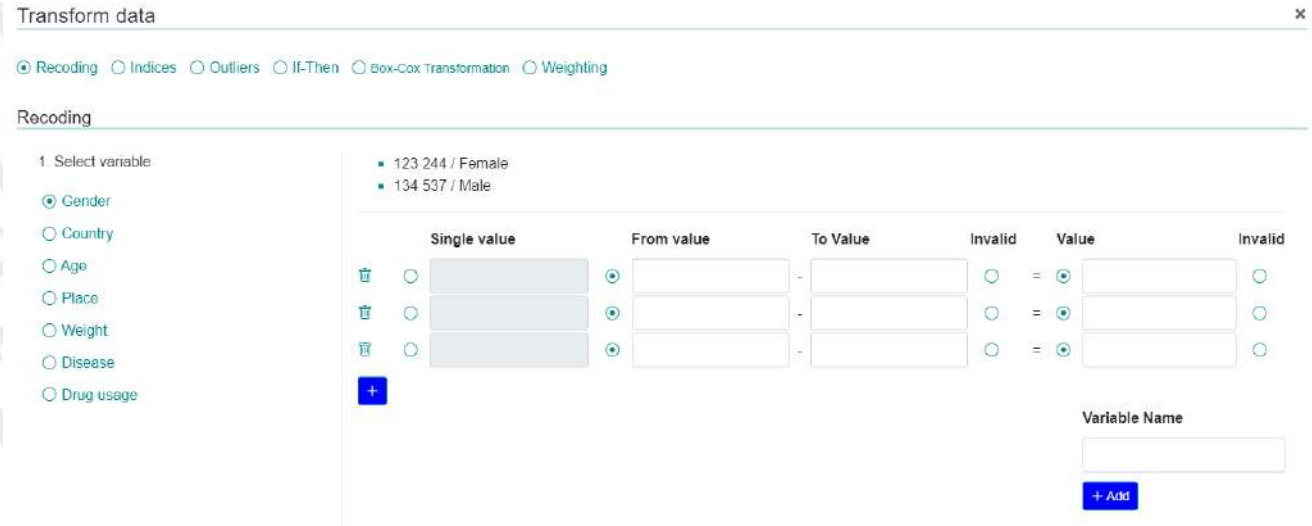
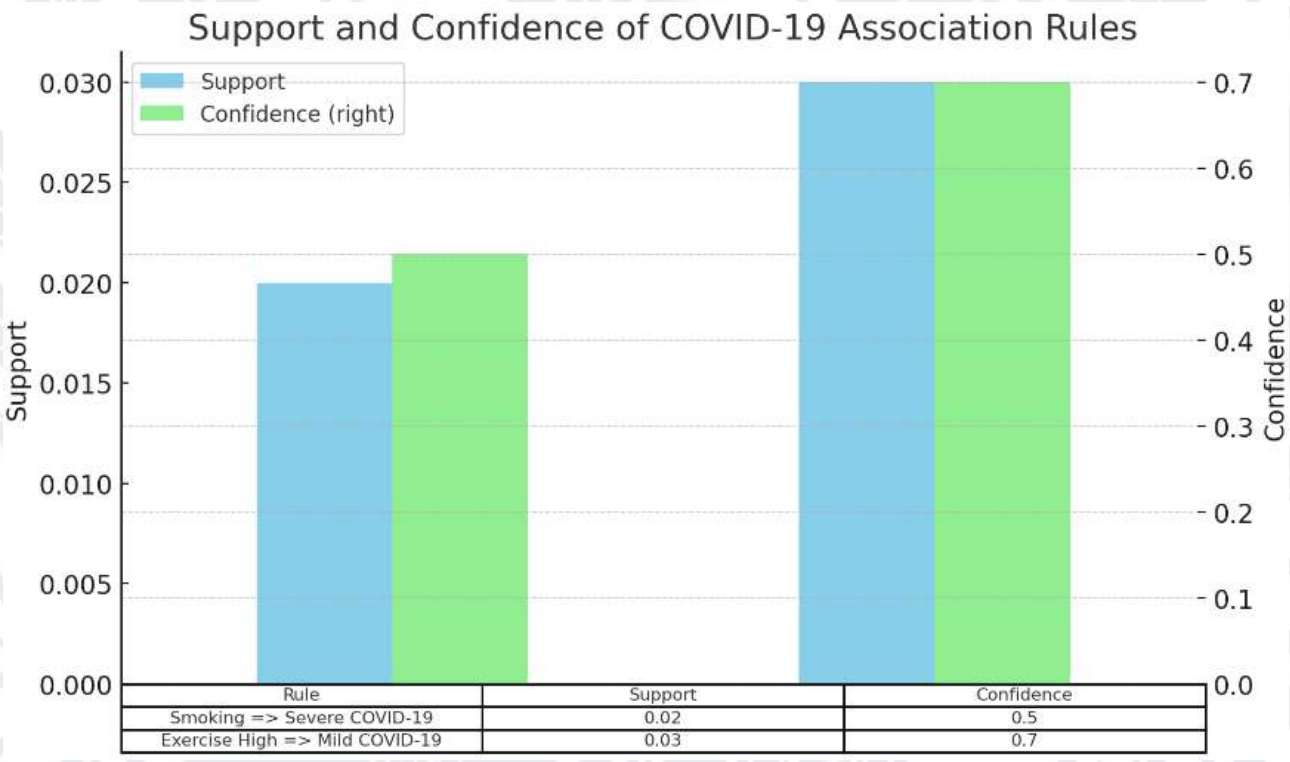


Рисунок 3.15 – Меню вибору параметрів

3.4 Результат роботи та аналізу системи

Проаналізуємо набір даних COVID-19 для пошуку закономірностей і кореляцій. Цей набір містить різноманітні дані, включаючи демографічні дані пацієнтів, фактори способу життя, статус інфікування COVID-19 та іншу важливу медичну інформацію. Точки даних: Вік пацієнта, стать, основні захворювання (діабет, гіпертонія, тощо), фактори способу життя (куріння, частота фізичних вправ), статус інфікування COVID-19 (позитивний/негативний), ступінь тяжкості інфекції (легка, середня, тяжка) та результат (одужав, госпіталізований, помер).

Мета аналізу: виявити зв’язок між факторами способу життя та тяжкістю перебігу і наслідками інфікування COVID-19.



Rule	Support	Confidence
Smoking => Severe COVID-19	0.02	0.5
Exercise High => Mild COVID-19	0.03	0.7

Рисунок 3.16 – Отримані результати

Набір даних складається з двох асоціативних правил - "Куріння => важка форма COVID-19" та "Висока фізична активність => легка форма COVID-19" - з відповідними значеннями підтримки та достовірності.

Підтримка показує відносну частоту правила серед усіх транзакцій. Правило "Куріння => важкий перебіг COVID-19" має підтримку 0,02, тобто воно з'являється у 2% усіх транзакцій.

Достовірність показує, як часто правило виявляється вірним. Достовірність 0,5 для "Куріння => Тяжкий перебіг COVID-19" означає, що в 50% випадків, коли фактором є куріння, настає тяжкий перебіг COVID-19.

Отримані результати аналізу мають потенціал для значного покращення якості медичних послуг, допомоги фахівцям у цій галузі та впливу на медичні дослідження. Це може бути наступним чином

- Покращення медичних послуг. Виявляючи кореляції, такі як підвищена тяжкість перебігу COVID-19 у курців, медичні працівники можуть більш ефективно адаптувати свої плани догляду за пацієнтами. Наприклад, вони можуть визначити пріоритетність певних профілактичних заходів або стратегій лікування для таких пацієнтів.
- Оцінка та управління ризиками. Розуміння факторів, пов'язаних з різними наслідками COVID-19, допомагає краще оцінювати ризики та керувати ними. Це може призвести до більш проактивної та превентивної медичної допомоги, зменшуючи навантаження на системи охорони здоров'я.
- Нові напрямки для досліджень. Виявлені зв'язки можуть привести до нових напрямків досліджень. Наприклад, зв'язок між факторами способу життя та тяжкістю перебігу COVID-19 може відкрити подальші дослідження того, як ці фактори впливають на інші захворювання.
- Інформування політики громадського здоров'я. Розуміння того, як фактори способу життя впливають на COVID-19, може стати основою для стратегій і кампаній у сфері громадського здоров'я, зокрема для інформування громадськості про ефективні заходи профілактики.

Отже, аналіз демонструє силу даних у формуванні сучасних медичних практик. Використовуючи можливості аналітики великих даних, системи охорони здоров'я можуть перейти до більш науково-обґрунтованих практик, покращити результати лікування пацієнтів і прокласти шлях до більш тонких і ефективних стратегій громадського здоров'я.

Висновки до розділу

В даному розділі були розглянуті технології та засоби використані при розробці системи обробки та аналізу даних в медичній сфері. Описані та проаналізовані сучасні архітектурні підходи для побудови таких систем. Наведені та проаналізовані результати роботи застосунку, що підкреслило актуальність даної теми.

ВИСНОВКИ

В рамках дипломної роботи були успішно виконані всі завдання. Проведено детальний аналіз предметної області, що підтвердив зростання попиту на технології, пов'язані з використанням Big Data в медичній сфері. Інформаційні технології стають невід'ємною частиною сучасного світу, і вони можуть вплинути на різні аспекти медичної практики та досліджень. Проаналізовано існуючі системи та сучасні методи, що застосовуються для вирішення подібних проблем. Цей аналіз дав цінну інформацію, висвітливши як потенціал, так і «підводні камені» різних підходів у сфері обробки та аналізу медичних даних. Проведено аналіз, щоб глибше вивчити предметну область і успішно використати архітектурні підходи, програмні засоби та інструменти для розробки системи аналізу даних з використанням технологій Big Data. Були проаналізовані різні моделі та підходи, що використовуються в даний час, висвітлено їхні сильні та слабкі сторони. Дослідження поширювалося на практичне застосування цих систем у реальних умовах, зокрема, зосереджуючись на тому, як вони керують, обробляють та аналізують великі обсяги медичних даних. У результаті роботи було розроблено систему для збору та аналізу медичних даних, яка повністю відповідає всім вимогам, які були поставлені. Усі функції додатку працюють як очікувалося та описано у специфікаціях.

Під час роботи над дипломним проектом набуто досвід у проектуванні архітектури серверних додатків, використовуючи багатошаровий підхід та принципи REST, набуті знання в сфері Big Data. Впроваджено та обґрунтовано сучасні підходи та концепції розробки програмного забезпечення, використовуючи бази даних і вивчаючи методологію тестування додатків. Висвітлено практичну цінність дослідження, вказані можливі переваги використання аналітики великих даних в медичній сфері.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Big Data in Healthcare [Електронний ресурс]. Режим доступу до ресурсу: https://straitsresearch.com/report/big-data-in-healthcare-market
2. Global Big Data in Healthcare Market - Industry Trends & Forecast Report 2028 [Електронний ресурс]. Режим доступу до ресурсу: https://www.blueweaveconsulting.com/report/big-data-in-healthcare-market
3. 24 Examples Of Big Data Analytics In Healthcare That Can Save People [Електронний ресурс]. Режим доступу до ресурсу: https://www.datapine.com/blog/big-data-examples-in-healthcare/
4. Big data in healthcare: management, analysis and future prospects [Електронний ресурс]. Режим доступу до ресурсу: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0217-0
5. Personalized Medicine [Електронний ресурс]. Режим доступу до ресурсу: https://www.genome.gov/genetics-glossary/Personalized-Medicine
6. How is Big Data Helping in the Development of Healthcare? [Електронний ресурс]. Режим доступу до ресурсу: https://www.analyticsvidhya.com/blog/2022/09/how-is-big-data-helping-in-the-development-of-healthcare/
7. Grand Challenges for Medtech Data Analytics [Електронний ресурс]. Режим доступу до ресурсу: https://www.frontiersin.org/articles/10.3389/fmedt.2019.00002/full
8. What is Big Data? [Електронний ресурс]. Режим доступу до ресурсу: https://www.oracle.com/big-data/what-is-big-data/
9. Characteristics of Big Data: Types & Examples [Електронний ресурс]. Режим доступу до ресурсу: https://bau.edu/blog/characteristics-of-big-data/
10. Defining Big Data Via the Three Vs [Електронний ресурс]. Режим доступу до ресурсу: https://csresearchers.wordpress.com/2015/02/18/defining-big-data-via-the-three-vs/

11. The Power of Big Data Analytics in Healthcare [Електронний ресурс]. Режим доступу до ресурсу: https://kms-healthcare.com/5-benefits-of-big-data-analytics-in-healthcare/
12. Top 10 Big Data Solutions in Healthcare Healthcare [Електронний ресурс]. Режим доступу до ресурсу: https://symphony-solutions.com/insights/top-10-big-data-solutions-in-healthcare
13. What is Data Processing? [Електронний ресурс]. Режим доступу до ресурсу: https://ua.talend.com/resources/what-is-data-processing/
14. The use of Big Data Analytics in healthcare [Електронний ресурс]. Режим доступу до ресурсу: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8733917/
15. How IBM's Watson Went From the Future of Health Care to Sold Off for Parts [Електронний ресурс]. Режим доступу до ресурсу: https://slate.com/technology/2022/01/ibm-watson-health-failure-artificial-intelligence.html
16. IBM Watson Health's Challenges Tell Us More About Healthcare Data Than It Does About AI [Електронний ресурс]. Режим доступу до ресурсу: https://www.forbes.com/sites/forbestechcouncil/2022/05/03/ibm-watson-healths-challenges-tell-us-more-about-healthcare-data-than-it-does-about-ai/?sh=189bba355b48
17. Apple Watch. Empowering your patients to live a healthier day [Електронний ресурс]. Режим доступу до ресурсу: https://www.apple.com/healthcare/apple-watch/
18. 8 best Apple Watch health features you didn't know about [Електронний ресурс]. Режим доступу до ресурсу: https://www.insider.com/guides/health/fitness/best-apple-watch-health-features
19. 23andMe [Електронний ресурс]. Режим доступу до ресурсу: https://www.23andme.com/about/
20. 23andMe Review [Електронний ресурс]. Режим доступу до ресурсу: https://www.forbes.com/health/medical-supplies/23andme-

review/#:~:text=What%20are%20the%20disadvantages%20of,of%20a%20disease%2C%20explains%20Klee.

21. What are association rules in data mining? [Електронний ресурс]. Режим доступу до ресурсу: https://www.techtarget.com/searchbusinessanalytics/definition/association-rules-in-data-mining
22. An Overview of Association Rule Mining & its Applications [Електронний ресурс]. Режим доступу до ресурсу: https://www.upgrad.com/blog/association-rule-mining-an-overview-and-its-applications/
23. Apriori Algorithm [Електронний ресурс]. Режим доступу до ресурсу: https://www.geeksforgeeks.org/apriori-algorithm/
24. What is Apriori Algorithm? [Електронний ресурс]. Режим доступу до ресурсу: https://www.javatpoint.com/apriori-algorithm
25. AIS Algortihm [Електронний ресурс]. Режим доступу до ресурсу: https://www.ijcsmc.com/docs/papers/June2015/V4I6201552.pdf
26. AIs, Algorithms & Machine Learning: Which is For You? [Електронний ресурс]. Режим доступу до ресурсу: https://www.valuer.ai/blog/ais-algorithms-machine-learning-which-is-for-you
27. SETM Algorithm [Електронний ресурс]. Режим доступу до ресурсу: https://www.saedsayad.com/association_rules.htm
28. An Implementation of SETM Algorithm Using Super Market Dataset [Електронний ресурс]. Режим доступу до ресурсу: https://indjst.org/articles/mining-frequent-item-sets-for-association-rule-mining-in-relational-databases-an-implementation-of-setm-algorithm-using-super-market-dataset
29. Comparison and Analysis of Algorithms for Association Rules [Електронний ресурс]. Режим доступу до ресурсу: https://www.researchgate.net/publication/220698670_Comparison_and_Analysis_of_Algorithms_for_Association_Rules

30. What is .NET Platform? [Електронний ресурс]. Режим доступу до ресурсу: https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet
31. "Pro C# 9 with .NET 5" by Andrew Troelsen and Philip Japikse [Текст]
32. Overview of ASP.NET Core [Електронний ресурс]. Режим доступу до ресурсу: https://learn.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-8.0
33. ASP.NET Core Application Architecture [Електронний ресурс]. Режим доступу до ресурсу: https://dotnet.microsoft.com/en-us/learn/aspnet/architecture
34. ASP.NET Core 8 Pros and Cons [Електронний ресурс]. Режим доступу до ресурсу: https://ukad-group.com/blog/aspnet-core-8-pros-and-cons/
35. What is MongoDB [Електронний ресурс]. Режим доступу до ресурсу: https://www.mongodb.com/what-is-mongodb
36. MongoDB Features [Електронний ресурс]. Режим доступу до ресурсу: https://www.mongodb.com/what-is-mongodb/features
37. Why Use MongoDB and When to Use It? [Електронний ресурс]. Режим доступу до ресурсу: https://www.mongodb.com/why-use-mongodb
38. Microsoft SQL Server [Електронний ресурс]. Режим доступу до ресурсу: https://www.microsoft.com/en-us/sql-server/sql-server-2022
39. SQL Engine Definition [Електронний ресурс]. Режим доступу до ресурсу: https://www.heavy.ai/technical-glossary/sql-engine
40. Understanding Multilayered Architecture in .NET [Електронний ресурс]. Режим доступу до ресурсу: https://www.c-sharpcorner.com/UploadFile/1492b1/understanding-multilayered-architecture-in-net/
41. Multi-Layered Architecture for ASP.NET Core [Електронний ресурс]. Режим доступу до ресурсу: https://copyprogramming.com/howto/asp-net-core-multi-layer-architecture
42. ASP.NET Core Identity DB Scheme [Електронний ресурс]. Режим доступу до ресурсу: https://deblokt.com/2020/01/24/04-part-2-identityserver4-asp-net-core-identity-net-core-3-1/

43. JWT Introduction [Електронний ресурс]. Режим доступу до ресурсу: https://jwt.io/introduction

Додаток 2 до наказу
від «31» березня 2023 року
№119/05

ДЕКЛАРАЦІЯ

про дотримання академічної доброчесності

Я, _____
_____

Повністю вказується ПІБ та статус (посада для працівників, освітня (освітньо-наукова) програма – для здобувачів вищої освіти)

що нижче підписалась/підписався, розуміючи та підтримуючи загальновизнані засади справедливості, доброчесності та законності,

ЗОБОВ'ЯЗУЮСЬ:

дотримуватися принципів та правил академічної доброчесності, що визначені законодавством України, локальними нормативними актами Донецького національного університету імені Василя Стуса, положеннями, правилами, умовами, визначеними іншими суб'єктами, та не допускати їх порушення.

ПІДТВЕРДЖУЮ:

що мені відомі положення статті 42 Закону України «Про освіту»;

що у даній роботі не представляла/представляв чиїсь роботи повністю або частково як свої власні. Там, де я скористалася/скористався працею інших, я зробила/зробив відповідні посилання на джерела інформації;

що дана робота не передавалась іншим особам і подається вперше, не порушує авторських та суміжних прав закріплених статтями 21-25 Закону України «Про авторське право та суміжні права», а дані та інформація не отримувались в недозволений спосіб.

УСВІДОМЛЮЮ:

що ця робота може бути перевірена університетом на плагіат або інші порушення академічної доброчесності, в тому числі з використанням спеціалізованих сервісів;

що у разі порушення академічної доброчесності, до мене можуть бути застосовані процедури, передбачені законодавством України та Кодексом академічної доброчесності та корпоративної етики Донецького національного університету імені Василя Стуса, іншими локальними нормативними актами університету, та я можу бути притягнута/притягнутий до академічної відповідальності.

_____ _____

(дата) (підпис)