

ЗАТВЕРДЖУЮ
Директор ЗПО «ДІТК»
Олександр СТРИЛЕЦЬ
31.08.2026 р.
ПОГОДЖЕНО
Заступник директора з навчально-методичної
роботи ЗПО «ДІТК»
Сергій ГОРДІЄНКО
31.08.2026 р.

КОНСПЕКТ ЛЕКЦІЙ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «АНАЛІТИКА ДАНИХ ТА ЦИФРОВІ ІНСТРУМЕНТИ В СУЧАСНОМУ МАШИНОБУДУВАННІ»

ВСТУП

Лекційний курс з дисципліни «Аналітика даних та цифрові інструменти в сучасному машинобудуванні» покликаний сформулювати у вас цілісне розуміння того, як дані перетворюються на знання, а знання — на обґрунтовані інженерні рішення. У сучасному світі машинобудування вже неможливо уявити без цифрових технологій: кожен верстат, кожен зварювальний апарат, кожен датчик генерує інформацію, яка потребує обробки та аналізу.

Аналітика даних — це не просто набір інструментів. Це спосіб мислення, який дозволяє інженеру бачити закономірності там, де інші бачать хаос, прогнозувати проблеми до того, як вони стануть критичними, та приймати рішення на основі фактів, а не інтуїції. Саме тому цей курс є важливою складовою підготовки фахового молодшого бакалавра з прикладної механіки.

Лекційний матеріал побудований за принципом «від простого до складного». Ви починаєте з базових понять аналітики даних, знайомитеся з життєвим циклом даних, типами даних та джерелами інформації у машинобудуванні. Далі ви опановуєте статистичні методи, вчитеся візуалізувати дані та будувати регресійні моделі. Завершується курс вивченням основ машинного навчання, інтеграції штучного інтелекту у виробництво та етичних аспектів роботи з даними.

Кожна лекція супроводжується практичними прикладами з галузі зварювального виробництва. Ви побачите, як аналітика даних допомагає виявляти дефекти зварних швів, прогнозувати знос обладнання, оптимізувати технологічні процеси та підвищувати якість продукції.

РОЗДІЛ І.

ЖИТТЄВИЙ ЦИКЛ АНАЛІТИКИ ВЕЛИКИХ ДАНИХ

Тема 1. Класифікація та типи обчислень великих даних

У сучасному світі великі дані відіграють важливу роль, оскільки обсяг та різноманітність даних постійно зростають. Ці дані можуть бути використані для отримання нових знань, прийняття рішень та вдосконалення різноманітних процесів. Підключені пристрої Інтернету речей спричиняють збільшення обсягу та різноманітності даних. Вони надають нові можливості для збору та використання інформації з реального світу. Дані можуть бути структурованими (такими як таблиці баз даних) або неструктурованими (такими як текст, зображення, аудіо, відео). Технологія хмарних обчислень дозволяє отримувати доступ до обчислювальних ресурсів через мережу Інтернет та використовується для забезпечення обробки даних, зберігання та спільної роботи. Технологія туманних обчислень покликана надати обчислювальні ресурси та обробку даних на рівні "туману", ближче до джерела даних, забезпечуючи швидку обробку та низьку затримку.

1.1. Великі дані. Зростання даних в учасному світі

Поняття "великі дані" (Big Data) не має жорстко визначеного числового значення, оскільки обсяг даних, який може вважатися "великим", залежить від контексту, доступних ресурсів та технологічних можливостей на певний момент часу. Наприклад, Пол Зікопаулос з IBM заявив, що для віднесення до великих даних необхідно мати обсяг від 200 до 600 терабайт (1 терабайт (TB) – це одиниця вимірювання обсягу даних, яка еквівалентна 1 трильйону (10^{12}) байтів). Міжнародна корпорація даних (International Data Corporation, IDC) вважає, що набір даних об'ємом 100 терабайт можна класифікувати як Big Data. Якщо ж дані є потоковими, їхній обсяг може бути меншим за 100 терабайт, але все одно вважатися Big Data, за умови, що обсяг цих даних збільшується більш ніж на 60% щороку. Згідно з Національним інститутом стандартів і технологій (National Institute of Standards and Technology, NIST): "Концепція великих даних передбачає розподіл систем даних на незалежні ресурси, пов'язані горизонтально, що дозволяє досягти масштабованості, необхідної для ефективної обробки

великих обсягів інформації". Великі дані можуть вважатися тоді, коли їх обсяг настільки великий, що стандартні методи обробки та аналізу стають недостатніми або непридатними. Це може бути, наприклад, коли обсяг даних вимірюється в **петабайтах (1 петабайт (PB) – це одиниця виміру обсягу даних, яка еквівалентна 1 квадрильйону (10^{15}) байтів або 1024 екзабайтам)** або коли обробка даних вимагає використання розподілених систем обчислень.

Розглянемо кілька прикладів генерування великих даних у реальному світі. Наприклад, двигун Airbus A380 під час польоту з Лондона до Сінгапура генерує 1 петабайт даних. Великий адронний колайдер (Large Hadron Collider, LHC) створює 1 гігабайт даних щосекунди (рис.1.1). LHC складається з понад 170 обчислювальних центрів у 42 країнах світу та здатний обробляти понад 30 петабайтів на рік даних, які утворюються внаслідок фізичних експериментів на LHC.

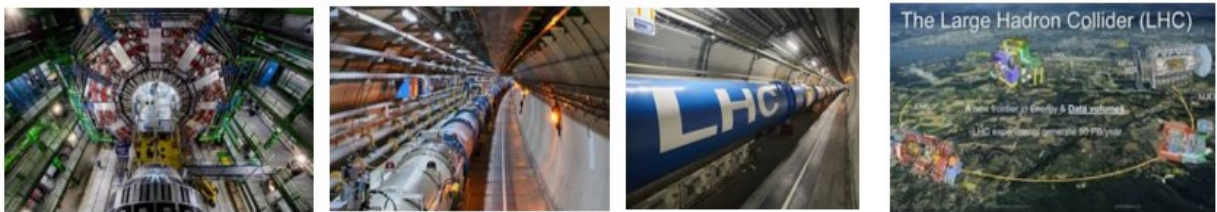


Рис. 1.1. Великий адронний колайдер (Large Hadron Collider, LHC)

Найбільший радіотелескоп у світі – квадратний кілометровий масив (Square Kilometre Array, SKA), який продукує 20 екзабайтів даних щодня, що дорівнює 20 мільярдам гігабайт на день.

Інтернет речей збирає дані за допомогою різних датчиків. Ці дані можуть надходити, наприклад, від датчиків температури та вологості, які використовуються в сільському господарстві. Сьогодні датчики встановлюються практично всюди: у смартфонах, автомобілях, реактивних двигунах і побутовій техніці. Кількість пристроїв з датчиками щороку зростає, що сприяє швидкому збільшенню обсягів великих даних.

Для приблизного розуміння, великі дані зазвичай характеризуються терміном **"3V"**, що вказує на три основні властивості або аспекти таких даних: об'єм (**Volume**), різноманітність (**Variety**), швидкість (**Velocity**). Великі дані означають великі обсяги інформації,

які перевищують можливості традиційних баз даних та інструментів обробки даних. В цьому контексті великими можуть бути обсяги даних, що досягають кількох терабайт, петабайт чи навіть більше. Великі дані можуть включати різноманітні типи даних, такі як текстові, числові, відео, аудіо, графічні файли, соціальні медіа-дописи, сенсорні дані, геопросторові дані тощо, це визначає їх різноманітність.

Швидкість потоку даних також є важливою характеристикою великих даних. Дані можуть надходити з великою швидкістю, наприклад, це можуть бути дані, які генеруються в реальному часі (real-time data), які потрібно обробляти та аналізувати без затримок.

Крім "3V", деякі експерти додають ще "V" для властивості "валідність" (**Validity**), що означає правильність та точність даних, а також "V" для властивості "волатильність" (**Volatility**), що стосується зміни обсягів та характеру даних з часом.

На сьогоднішній день, із зростанням обчислювальних ресурсів та технологічних можливостей, деякі великі компанії можуть опрацьовувати обсяги даних, що декілька років тому вважалися великими, за допомогою ефективних інструментів та розподілених систем. Зростання обсягу даних відбувається експоненційно і називається "**експлозією даних**". Це вимагає розвитку нових технологій для збору, зберігання, обробки та аналізу даних. Обробка та аналіз великих обсягів даних може допомогти отримати нові знання для різних галузей, від бізнесу до медицини, транспорту та науки.

Зберігання та аналітика великих даних приносять із собою ряд проблем, пов'язаних з їхнім обсягом, різноманітністю та швидкістю надходження. Зокрема, обробка та зберігання великих обсягів даних вимагають потужних обчислювальних ресурсів та інфраструктури, яка може бути дороговартісною. Зберігання великих даних може призвести до значних витрат на серверне обладнання, диски та системи зберігання даних. Неструктуровані дані (такі як текст, зображення, відео) потребують спеціалізованих методів обробки та аналізу, що може бути складним завданням. Інтеграція даних з різних джерел може стати проблемою через різноманітність форматів та структур даних.

Деякі застосунки та складні програмні системи потребують аналізу даних у реальному часі, що вимагає швидкості та низької латентності в обробці. Обробка та зберігання великих даних може

викликати питання щодо захисту даних від несанкціонованого доступу та кібератак. Приклади небезпек зберігання приватних даних включають несанкціонований доступ до персональної інформації, ідентифікаційних даних, фінансових реквізитів та медичної історії. Це може призвести до крадіжки особистої інформації та використання її для шахрайства, кібератак, або навіть для шантажу. Крім того, недостатня захищеність даних може призвести до порушення конфіденційності, що може викликати негативний вплив на репутацію особи або організації.

Збільшення обсягів великих даних створює такі загрози, як потенційні проблеми з конфіденційністю та безпекою даних, складність управління та аналізу великих обсягів інформації, можливість виникнення помилкових або неточних висновків через надмірну кількість даних, а також потребу в інвестиціях у високотехнологічні інфраструктури та персонал. Великі обсяги даних можуть створити проблеми з їх ефективним зберіганням, резервним копіюванням та архівацією, також можуть вимагати вдосконалення методів обробки та аналізу даних для отримання цінної інформації.

Обробка та аналіз великих даних може вимагати використання спеціалізованих алгоритмів та програмних інструментів, які можуть бути важкодоступними або вимагати спеціалізованого навчання.

1.2. Зростання пристроїв Інтернету речей

Зростання великих даних пов'язане зі збільшенням кількості пристроїв **Інтернету речей (Internet of Things, IoT)**, а також пристроїв, обладнаних сенсорами та здатними генерувати дані. IoT описує мережу фізичних об'єктів (приладів, сенсорів, пристроїв), які обладнані технологією збору та обміну даними через мережу Інтернет без прямої взаємодії з людиною. Ці об'єкти можуть збирати, передавати та аналізувати інформацію, що може включати різні параметри та дії (сенсори в автомобілях, побутових пристроях, медичних пристроях тощо). Кожен такий пристрій може генерувати великі обсяги даних навіть за короткий проміжок часу. Наприклад, це можуть бути смартфони, медичні пристрої, смарт-прилади для дому, автомобілі з IoT-забезпеченням тощо. Інтернет речей, створений у 1999 році Кевіном Ештоном, з часом привертає все більше уваги через збільшення

кількості підключених пристроїв і, як наслідок, збільшення обсягу даних. В епоху великих даних запис даних із кількох середовищ і від різних користувачів надзвичайно цінний як з точки зору статистики, так і з точки зору бізнесу та економіки. У наш час майже кожен пристрій, присутній у повсякденному житті, містить вбудовану електроніку, яка перетворює його на потенційний вузол IoT.

Вузли IoT здатні сприймати інформацію та передавати її завдяки комунікаційному інтерфейсу. Досі парадигма IoT мала величезний потенціал, вплив як на життя споживачів, так і на бізнес-моделі через зниження вартості впровадження цих пристроїв і зростання попиту. Очікується, що ця тенденція буде швидко зростати.

Вузли IoT повинні сприймати та збирати дані щодо їх конкретного завдання. Завдань може бути багато, починаючи від «розумного» побутового приладу, моніторингу сну, відстеження фізичної активності людини тощо. Така інформація, зібрана від тисячі осіб, має гарантувати абсолютну конфіденційність для кінцевих користувачів.

Згідно парадигми **кордонних обчислень (Edge Computing)**, частина робочого навантаження децентралізована та розподілена між вузлами IoT, перетворюючи їх з простих датчиків на потужніші та інтелектуальні вбудовані системи. Завдяки цьому підходу зібрані дані можуть бути додатково проаналізовані безпосередньо в самій системі, дозволяючи більш швидшу операцію пост-обробки після передачі даних. Багато хто вважає парадигму периферійних обчислень екологічнішою альтернативою хмарним обчисленням завдяки її здатності обмежувати обсяг даних, що переміщуються, а отже, скорочувати споживання енергії. Зазвичай дані, які потрібно зберегти в пам'яті, надходять від периферійних пристроїв, які представлені датчиками, модулями прийому передавача для зв'язку або роз'ємами. Етапи, які виконуються у вбудованій програмі, можна представити отриманням даних із датчиків, опрацюванням їх через мікроконтролер та передачею результатів за допомогою бездротового зв'язку (Bluetooth, мобільної мережі, Wi-Fi, Zigbee, Z-Wave, LoRaWAN, спеціальних протоколів передачі даних тощо).

MCU (Microcontroller Unit) – це компактний інтегрований пристрій (мікросхема), що об'єднує в собі центральний процесор **CPU (Central Processing Unit)**, пам'ять і периферійні пристрої для управління

електронними системами. Такі пристрої зазвичай використовуються для вбудованих систем, таких як побутові пристрої, автомобільні системи, медичні прилади, роботи, промислове устаткування та інші додатки, де потрібно виконувати контроль, обробку даних або керування різними функціями. MCU має невеликі розміри, низьку вартість та низьке енергоспоживання, що робить їх ідеальним вибором для багатьох вбудованих систем (рис.1.2).

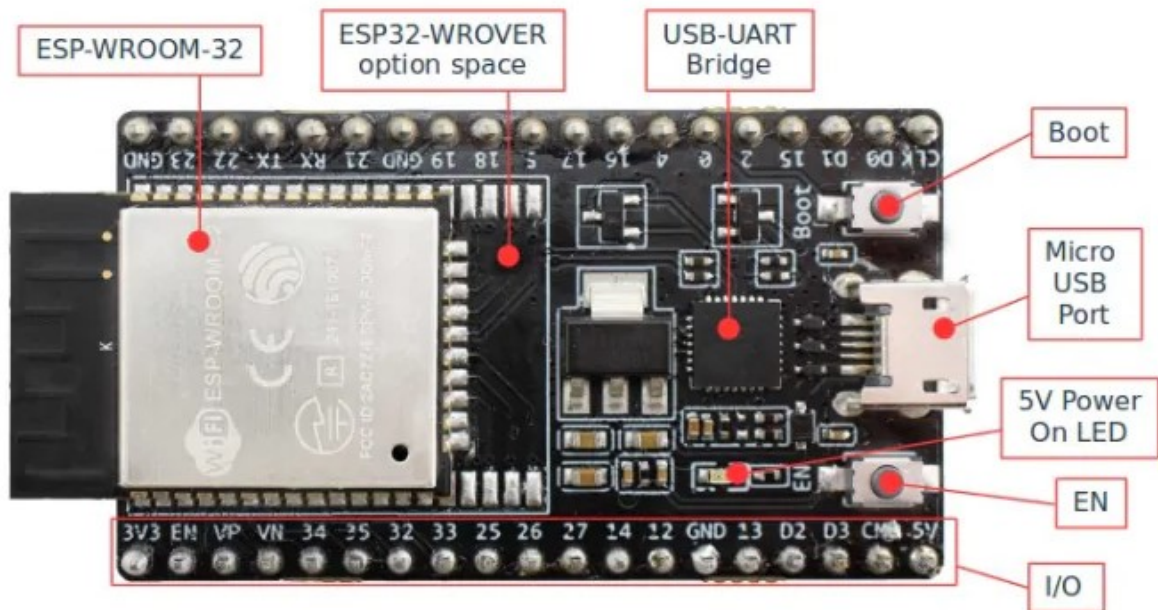


Рис. 1.2. Мікроконтролер ESP32 від Espressif Systems

Серія ESP32 від Espressif Systems поєднує Wi-Fi і Bluetooth Low Energy (BLE) в одному SoC (System on a Chip), який широко використовується в додатках IoT завдяки своїй енергоефективності та універсальності. Під час фази пробудження, яка ініціюється зовнішньою подією, стан MCU відновлюється разом із джерелом живлення та сигналом годинника. Після виконання завдання MCU зберігає свій поточний стан перед повторним переходом у стан низького енергоспоживання. Така поведінка є передбачуваною, тому головна система може прокидатися та переводити в режим сну різних учасників, включаючи себе, вдаючись до чітко визначеного керування живленням.

ESP32 від Espressif Systems – це серія мікроконтролерів, основними характеристиками яких є інтеграція бездротових інтерфейсів, можливість енергоефективної роботи, а також велика кількість периферійних модулів. ESP32 оснащено двоядерним

процесором Xtensa LX6 від Tensilica, який може працювати на тактових частотах до 240 МГц. Є також версії з одноядерним процесором для зниження енергоспоживання. Процесор має 32-розрядну архітектуру з підтримкою як основного ядра для обчислень, так і додаткового ядра для виконання фонових завдань.

В ESP32 вбудовано 520 КБ SRAM та підтримка зовнішньої оперативної пам'яті (PSRAM) до 8 МБ, що дозволяє працювати з великими обчислювальними задачами або даними. ESP32 має вбудовану флеш-пам'ять, зазвичай об'ємом від 4 МБ до 16 МБ, яка використовується для зберігання прошивок і даних користувача.

ESP32 підтримує стандарти безпроводового зв'язку 802.11 b/g/n з частотою 2,4 ГГц. Може працювати як в режимі точки доступу (AP), так і в режимі клієнта (STA), підтримує як Bluetooth 4.2, так і Bluetooth Low Energy (BLE), що робить його придатним для енергоефективних рішень в IoT-пристроях.

В ESP32 наявні такі периферійні інтерфейси:

- GPIO (General-Purpose Input/Output), що містить до 34 багатофункціональних пінів для роботи із зовнішніми компонентами, які підтримують такі протоколи, як SPI, I2C, I2S, UART;
- вбудовані аналого-цифрові перетворювачі (ADC, Analog-to-Digital Converter) та цифро-аналогові перетворювачі (DAC, Digital-to-analog converter) для вимірювання та виведення аналогових сигналів;
- мікроконтролер підтримує широтно-імпульсну модуляцію (PWM, Pulse-Width Modulation), що дозволяє регулювати потужність або керувати двигунами;
- вбудовані криптографічні прискорювачі для безпечного обміну даними через мережі, підтримується апаратне шифрування AES, SHA та RSA. Також є функція Secure Boot, яка запобігає запуску несанкціонованого програмного забезпечення.

Завдяки вбудованим бездротовим модулям, ESP32 може одночасно виконувати кілька завдань — наприклад, керувати підключенням до Wi-Fi-мережі та обмінюватися даними через Bluetooth, що робить даний мікроконтролер універсальним для створення IoT-пристроїв, які потребують одночасного використання кількох типів зв'язку.

ESP32 підтримує кілька режимів низького енергоспоживання:

- Deep Sleep – споживання лише кількох мікроамперів, дозволяючи мікроконтролеру залишатися в стані очікування з мінімальним використанням енергії;

- Light Sleep – використовується для виконання періодичних задач, при цьому енергоспоживання залишається низьким.

Ці режими роблять ESP32 ідеальним для батарейних рішень або пристроїв, які повинні працювати тривалий час без підзарядки.

ESP32 може одночасно виконувати кілька завдань завдяки FreeRTOS – операційній системі реального часу, яка інтегрована в прошивку. Це дозволяє створювати більш складні програми з чітким управлінням задачами.

Програмувати ESP32 можна кількома мовами.

- C/C++ через Arduino IDE – широко використовується через простоту та наявність великої кількості бібліотек.

- MicroPython – інтерпретована мова програмування, що дозволяє швидко прототипувати та виконувати скрипти без потреби в компіляції.

- ESP-IDF (Espressif IoT Development Framework) – офіційний SDK, який дозволяє використовувати всі можливості ESP32 для розробки додатків з високою продуктивністю.

Завдяки інтегрованому Wi-Fi та Bluetooth, ESP32 є ідеальним вибором для побудови «розумних» пристроїв, таких як датчики, «розумні» світильники та системи керування доступом. Підтримка широкого набору інтерфейсів дозволяє використовувати ESP32 для домашньої або промислової автоматизації. Завдяки енергоефективним режимам, ESP32 використовується в проєктах, що потребують тривалої автономної роботи. З підтримкою протоколу I2S, мікроконтролер може обробляти аудіо та використовуватися для створення аудіопристроїв або потокової передачі звуку. Перевагами ESP32 є висока продуктивність при низькому споживанні енергії, підтримка Wi-Fi та Bluetooth на одному чипі, багатофункціональні периферійні інтерфейси, гнучкість у програмуванні та підтримка різних мов. ESP32 є одним із найпопулярніших мікроконтролерів завдяки його можливостям та доступності для широкого кола розробників.

Разом з апаратними системами обробка даних залучає центральний процесор CPU, оскільки він має отримати дані з пам'яті чи

датчиків, обробити їх (зазвичай за допомогою спеціального алгоритму програми) та зберегти у пам'яті. З точки зору обробки даних, економія енергії може бути досягнута шляхом збільшення паралелізму, а саме обсягу даних за цикл, який може обробити центральний процесор або оптимізації архітектури щодо енергоспоживання за цикл. Мікроконтролери перейшли від початкових 8-розрядних до 32-розрядних сучасних пристроїв.

Кількість підключених пристроїв до Інтернету речей зростає з кожним роком. Від смартфонів і побутових пристроїв до індустріальних сенсорів та транспортних систем, все більше пристроїв стають частинами мережі IoT і використовуються в медицині, автомобільній промисловості, сільському господарстві, логістиці, містобудуванні та багатьох інших галузях.

Процесори стають більш потужними, а алгоритми – складнішими. Це дозволяє пристроям IoT аналізувати та розуміти дані на місці їх збору, забезпечуючи більш ефективний та швидкий відгук та запуск виконавчих механізмів. Зі зростанням підключених пристроїв також зростає ризик кібератак та порушення конфіденційності даних.

Застосування штучного інтелекту допомагає пристроям IoT більш ефективно аналізувати дані, виявляти закономірності та виходити на нові рівні автоматизації та оптимізації.

Штучний інтелект (Artificial Intelligence, AI) – це галузь комп'ютерних наук, що ставить перед собою завдання створення систем, які можуть самостійно виконувати завдання, які зазвичай потребують інтелекту людини. Розробка спільних стандартів та протоколів забезпечує сумісність та взаємодію між різними пристроями та системами, що робить реалізацію IoT більш ефективною. Застосування IoT у геопросторових інформаційних системах допомагає в зборі та аналізі даних про місцезположення, що є корисним для містобудування, транспорту та інших галузей. Розробка більш ефективних та довготривалих джерел живлення дозволяє пристроям IoT працювати безперервно і довше. Розвиток IoT продовжує динамічно зростати і відкриває багато можливостей для технологічних інновацій у різних сферах. Ринок IoT демонструє стійкість, всупереч загальному економічному спаду. Наприклад, у 2022 році витрати підприємств США на IoT зросли на 21,5% до 201 мільярда доларів. Ринок IoT в Україні

також активно розвивається, спостерігається зростання інтересу до впровадження таких технологій у промисловість, сільське господарство, медицину та містобудування. В Україні розробляються та використовуються IoT-рішення для вдосконалення виробничих процесів, збільшення продуктивності, зниження витрат, спостерігається збільшення інвестицій у стартапи, що спеціалізуються на IoT-розробках. У сільському господарстві впроваджуються IoT-рішення для всебічного контролю, управління посівами, відстеження стану худоби та ефективного використання ресурсів, що підвищує продуктивність та оптимізує господарські витрати (рис. 1.3).

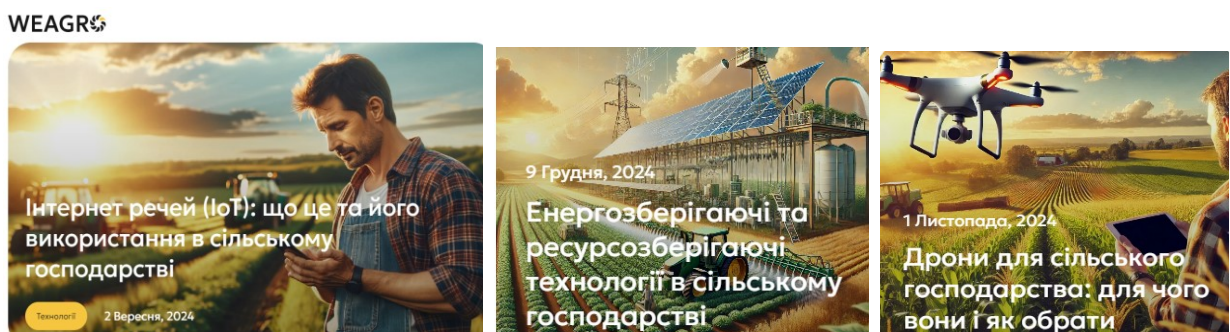


Рис. 1.3. Публікації сайту <https://weagro.com.ua>

У містобудуванні концепція «Smart City» набуває популярності, спрямовуючи зусилля на інтеграцію сучасних технологій для покращення управління міськими ресурсами та інфраструктурою. Це включає впровадження інтелектуальних систем управління транспортом, енергоспоживанням та комунальними послугами, що підвищує якість життя мешканців міст.

Інвестиції в українські IoT-стартапи також зростають. Наприклад, у 2016 році проєкт «розумної» гантелі Helko отримав 40 тисяч доларів від IoT Hub Kiev та 3 тисячі доларів від коворкінгу «Платформа» за 11% частки в стартапі. Українські стартапи у 2023 році залучили 209 мільйонів доларів венчурних інвестицій, що свідчить про сповільнення падіння обсягу інвестицій та появу ознак відновлення ринку.

У 2024 році український стартап Vidar Systems, що спеціалізується на розробці IoT-рішень, залучив 450 тисяч доларів інвестицій від чеського венчурного фонду Presto Ventures та групи інвесторів, що свідчить про зростаючий інтерес до українських IoT-розробок.

1.3. Класифікація великих даних

Великі дані можна класифікувати за різними аспектами відповідно до їх характеристик, технологій обробки та використання. Розглянемо основні класифікаційні категорії великих даних. За **обсягом (Volume)** дані можна розділити на **малі** та **великі**.

Малі дані (Small Data) – це невеликі обсяги даних, які можуть бути оброблені традиційними методами, зазвичай Small Data може охоплювати дані, які вміщуються в масштабі від 1 до десятків гігабайт (**1 гігабайт (ГБ)** становить 1024 мегабайти (МБ) або приблизно 1 мільярд байтів, це величина об'єму пам'яті, яка використовується для зберігання даних, таких як тексти, зображення, відео, аудіо тощо). Наприклад, це можуть бути дані з окремого датасету, текстові файли, дані з вимірювальних приладів, які можуть бути оброблені на локальних комп'ютерах або серверах без залучення великої обчислювальної інфраструктури.

Великі дані (Big Data) – це великі обсяги даних, які перевищують можливості традиційних методів та вимагають спеціалізованих підходів, зазвичай, Big Data охоплює обсяги даних, що вимірюються в **петабайтах (ПБ)**, **екзабайтах (ЕБ)** та більше, наприклад, великі дані можуть включати в себе масиви геномних послідовностей, дані сенсорів IoT, дані відеоспостереження та інші об'ємні набори даних.

1 петабайт (ПБ) становить 1024 терабайти (ТБ) або 1 квадрильйон (10^{15}) байтів. Такий обсяг може включати в себе великі масиви даних, великі бази даних, відеоархіви та інші об'єкти з великою кількістю інформації.

1 екзабайт (ЕБ) становить 1024 петабайти (ПБ) або 1 квінтильйон (10^{18}) байтів. Екзабайт може включати величезні обсяги інформації, такі як довготривалі архіви даних, великі наукові дослідження, глобальні масиви даних та інші великі ресурси зберігання.

Приватні великі дані – це дані, які належать конкретним організаціям або особам та зберігаються в закритих системах або базах даних. Вони можуть містити конфіденційну або чутливу інформацію, таку як особисті дані, фінансові дані або комерційні відомості.

Публічні великі дані – це дані, які доступні для загального використання та розповсюдження. Вони можуть бути зібрані з різних джерел, таких як урядові організації, дослідницькі установи, соціальні

мережі тощо, і можуть містити різноманітну інформацію, що відображає соціальні, економічні та інші аспекти суспільства.

Відмінності між приватними і публічними даними полягають у рівні доступності та конфіденційності. Приватні дані обмежені доступом і мають обмеження на використання, тоді як публічні дані доступні для широкого кола користувачів і можуть використовуватися для різних цілей без обмежень. Приватні дані регулюються законодавством певної країни чи уряду (рис. 1.4).

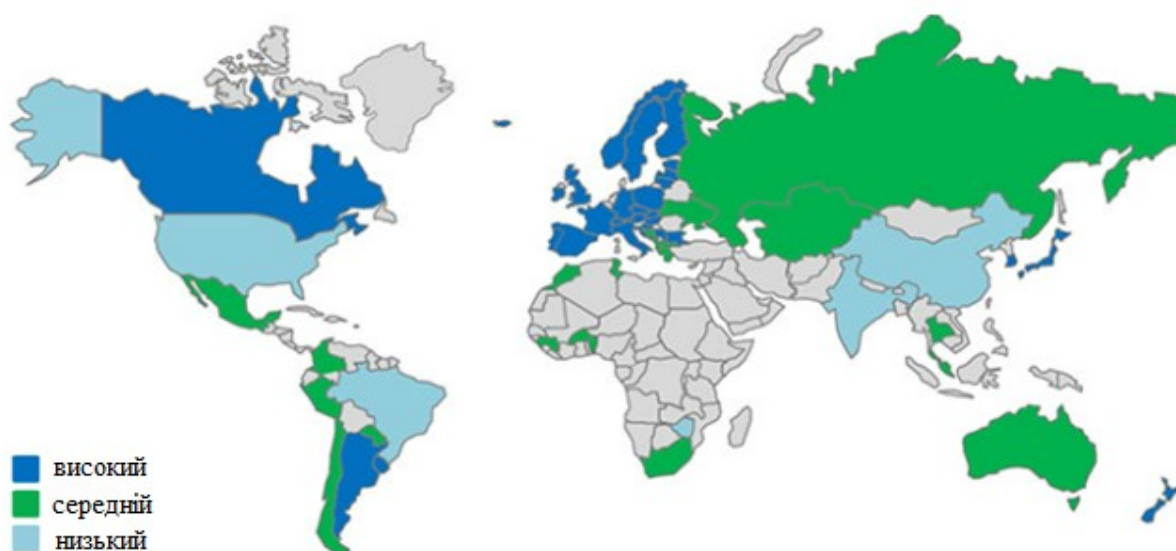


Рис. 1.4. Рівень законодавчого регулювання використання приватних даних у різних країнах (дані Всесвітнього економічного форуму-2014)

Законодавче регулювання приватних даних відрізняється між країнами, залежно від підходу до захисту конфіденційності громадян та бізнесу. Розглянемо приклади регулювання використання приватних даних у кількох країнах.

У Сполучених Штатах Америки немає єдиного всеосяжного закону про захист персональних даних. Натомість застосовується кілька федеральних і штатних законів, що регулюють певні галузі або аспекти.

- Health Insurance Portability and Accountability Act (HIPAA) – регулює захист медичних даних.
- Children's Online Privacy Protection Act (COPPA) – захищає конфіденційність дітей віком до 13 років в мережі Інтернет.
- California Consumer Privacy Act (CCPA) – один із

найсучасніших і жорстких законів на рівні штату, який забезпечує жителям Каліфорнії право контролювати, як їхні дані збираються, зберігаються та використовуються компаніями.

- Gramm-Leach-Bliley Act (GLBA) – стосується фінансових установ і вимагає, щоб вони захищали конфіденційність даних клієнтів.

У Європейському Союзі в 2018 році набув чинності закон General Data Protection Regulation (GDPR), основними вимогами якого є наступні.

- Права громадян на доступ до своїх даних, виправлення, видалення та обмеження їх обробки.

- Суворі правила щодо збору згоди на обробку даних.

- Високі штрафи за порушення (до 20 мільйонів євро або 4% річного обороту компанії).

- Обов'язкове повідомлення про порушення захисту даних (data breaches) протягом 72 годин.

GDPR є моделлю для законодавства про конфіденційність у багатьох країнах за межами ЄС.

Законодавство України щодо захисту персональних даних регулюється наступними актами.

1. Закон України "Про захист персональних даних" (2010 р.) – основний закон, який визначає, що персональні дані можуть оброблятися лише за згодою суб'єкта даних або на підставах, визначених законодавством. Передбачає для організацій обов'язок повідомляти про обробку персональних даних. Вимагає забезпечення конфіденційності та захисту даних. Зокрема, організації мають отримати чітку згоду громадян на обробку їхніх персональних даних.

2. Національна комісія з питань захисту персональних даних – державний орган, що відповідає за контроль дотримання законодавства про захист даних.

Україна рухається в бік гармонізації свого законодавства з європейськими стандартами, зокрема з GDPR, у рамках євроінтеграційних процесів.

В Канаді Personal Information Protection and Electronic Documents Act (PIPEDA) застосовується до комерційної діяльності і забезпечує права громадян на захист їхніх персональних даних.

У Бразилії Lei Geral de Proteção de Dados (LGPD) – це аналог

закону GDPR, який набрав чинності у 2020 році та встановлює жорсткі правила щодо збору, зберігання та використання приватних даних.

Кожна країна або регіон може мати свої особливості законодавства, але спільною тенденцією є посилення контролю за використанням персональних даних та підвищення відповідальності організацій щодо захисту приватності даних громадян.

За швидкістю (*Velocity*) оброблення великих даних виділяють методи пакетної обробки та обробки даних в реальному часі.

Пакетна обробка (*Batch Processing*) – це обробка даних в пакетах, зазвичай під час низького навантаження на систему. Це метод обробки даних або завдань, при якому група вхідних даних або завдань обробляється одночасно, в пакеті або партії. Дані обробляються в заздалегідь визначених пакетах, а не негайно по одному.

Пакетна обробка має свої переваги, коли потрібно обробляти великі обсяги даних або виконувати завдання, які можуть бути згруповані для оптимізації ресурсів та часу. Наприклад, при обробці великих обсягів даних, таких як фінансові транзакції, логи серверів або дослідницькі дані, важливо групувати їх у пакети для ефективної обробки та аналізу. У графічних програмах часто використовується пакетна обробка для застосування однакових фільтрів або змін до групи фотографій одночасно. Пакетна обробка також може бути використана для одночасного виконання пакету SQL-запитів до бази даних, таких як вставка, оновлення або видалення записів.

У фінансових системах пакетна обробка може використовуватись для одночасного оброблення групи фінансових транзакцій або платежів. У обробці природної мови чи обробці тексту можна застосовувати пакетну обробку для аналізу текстів у великих корпусах даних.

Пакетна обробка дозволяє ефективно використовувати ресурси, такі як обчислювальну потужність та час, оскільки декілька завдань обробляються одночасно. Однак цей підхід також може призвести до затримок у випадку, коли одне завдання в пакеті вимагає більше часу, ніж інші.

Обробка в реальному часі (*Real-Time Processing*) – це обробка даних негайно під час їхнього надходження, вимагає високої продуктивності та швидкості. Наприклад, обробка фінансових транзакцій в реальному часі дозволяє миттєво відслідковувати та

аналізувати грошові потоки.

У медичних системах обробка даних в реальному часі дозволяє відстежувати стан хворого, аналізувати показники та вчасно реагувати на зміни. Обробка даних в реальному часі в телекомунікаційних системах дозволяє миттєво передавати, обробляти та аналізувати дані великої кількості користувачів.

Обробка даних з сенсорів та пристроїв IoT в реальному часі дозволяє відстежувати стан об'єктів та взаємодіяти з ними негайно. Обробка даних з різних датчиків у виробничих процесах дозволяє негайно реагувати на проблеми та оптимізувати роботу обладнання. Технологія обробки в реальному часі вимагає високопродуктивного апаратного обладнання та програмного забезпечення, а також алгоритмів, які можуть виконувати операції швидко та ефективно.

У порівнянні з пакетною обробкою, де дані зазвичай групуються і обробляються разом, у реальному часі обробка відбувається індивідуально для кожної одиниці даних або події, дозволяючи миттєво реагувати на зміни та події.

За джерелом великих даних виділяють **дані соціальних мереж (Social Media Data)** – дані з соціальних платформ, таких як Twitter, Facebook, Instagram тощо; **сенсорні дані (Sensor Data)** – дані від різних сенсорів, таких як GPS, акселерометри, термометри тощо; **дані від IoT пристроїв (IoT Device Data)** – дані від підключених пристроїв IoT.

За значущістю (Value) виділяють «золоті дані» та дані незначної цінності.

«Золоті дані» (Golden Data) – це дані, які мають велику цінність для організації та допомагають приймати стратегічні рішення.

Дані незначної цінності (Low-Value Data) – це дані, які мають меншу стратегічну значущість та використовуються менше.

1.4. Структуровані та неструктуровані дані

За **різноманітністю (Variety)** виділяють **структуровані, неструктуровані та напівструктуровані дані** (рис. 1.5). Структуровані та неструктуровані дані – це дві основні категорії даних, які використовуються для опису інформації. Вони відрізняються за способом організації та форматом збереження.

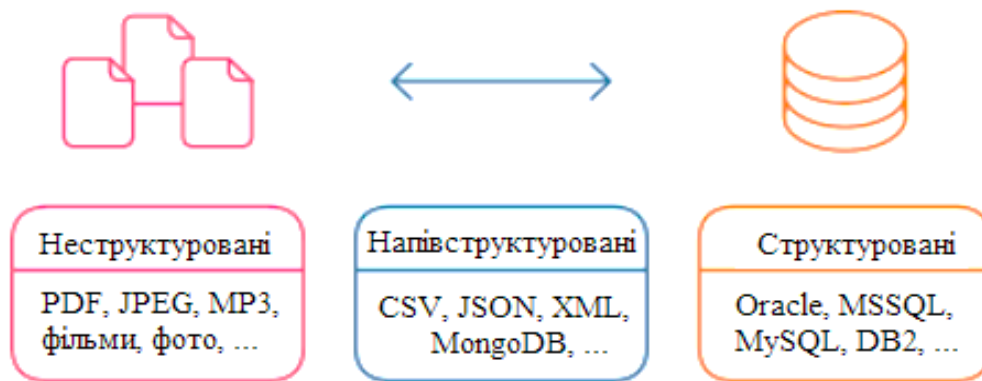


Рис. 1.5. Приклади структурованих, неструктурованих та напівструктурованих даних

Структуровані дані (Structured Data) – це дані, які організовані у чітку, систематичну структуру, яка ідентифікується та інтерпретується за допомогою стандартних правил. Вони зазвичай зберігаються в реляційних базах даних та таблицях, де кожен рядок та стовпець має чітку семантику. Розглянемо приклади структурованих даних.

Приклад 1. У таблиці бази даних з інформацією про товари кожен рядок представляє окремий товар, а стовпці містять інформацію, таку як ім'я, прізвище, адреса, номер телефону продавця тощо (рис.1.6).

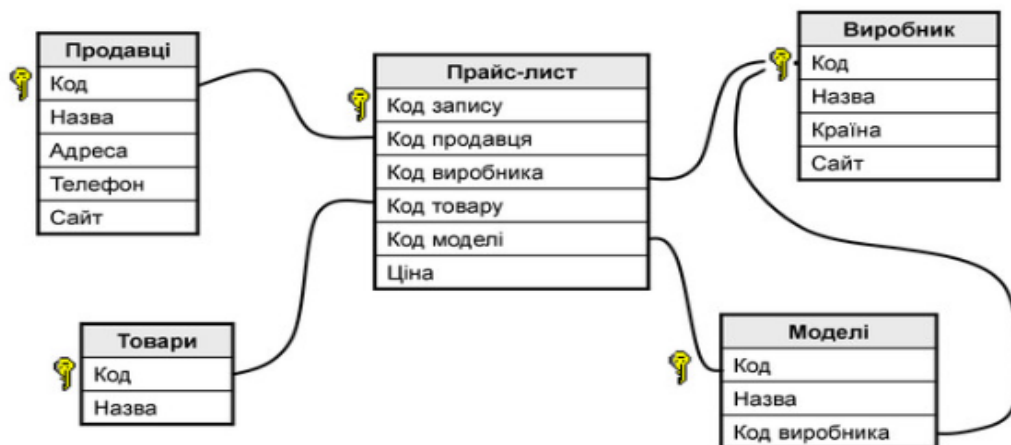


Рис. 1.6. Приклад структурованих даних реляційної бази даних

Приклад 2. Транзакційний журнал банку, де транзакції зберігаються у вигляді записів, кожен з яких має поля для дати, суми, типу операції тощо.

Приклад 3. Електронна таблиця з фінансовими даними, яка містить стовпці для дат, доходів, витрат та балансу.

Неструктуровані дані (Unstructured Data) – це дані без чіткої структури, такі як текстові файли, зображення, відео, аудіо тощо. Їх важко або навіть неможливо інтерпретувати за допомогою стандартних методів обробки, оскільки вони можуть містити текст, графіку, зображення, відео тощо. Розглянемо приклади неструктурованих даних.

Приклад 1. Текстові документи, наприклад, статті, електронні листи. Текст може містити інформацію різного типу – від простого тексту до складних аналітичних матеріалів.

Приклад 2. Зображення та фотографії, наприклад, фото з камери, картинки, графіки. Вони можуть містити велику кількість інформації, але ця інформація не завжди може бути автоматично розпізнана.

Приклад 3. Аудіо- та відео записи (музика, відеоролики, подкасти тощо). Основною інформацією є звук чи візуальне зображення, але їх інтерпретація вимагає спеціалізованого аналізу.

Приклад 4. Коментарі, дописи, повідомлення з соціальних мереж (Facebook, Instagram, Twitter, LinkedIn) – ці дані можуть містити текст, фотографії, посилання та інші вкладені об'єкти.

Приклад 5. Скановані документи, паперові документи, які були перетворені у цифровий вигляд, такі як рукописні нотатки або офіційні листи.

У сучасному світі дані можуть бути різноманітними, поєднуючи структуровані та неструктуровані елементи.

Напівструктуровані дані (Semi-Structured Data) – це дані, які мають часткову структуру, наприклад, JSON- або XML-файли. Особливості напівструктурованих даних включають можливість вміщувати різноманітні типи інформації в одному наборі, варіативність у форматі та структурі даних, а також можливість додавання нових полів або змін до існуючих даних без необхідності зміни всієї бази даних. Це робить напівструктуровані дані корисними для ситуацій, де потрібна гнучкість у зберіганні та обробці інформації, наприклад, в аналізі веблогів або обробці даних з різних джерел.

Напівструктуровані дані в аналізі веблогів можуть включати інформацію про запити користувачів, час відвідувань, IP-адреси, типи запитів та інші параметри, які дозволяють розуміти поведінку користувачів на вебсайті та оптимізувати його функціонування.

Реляційні бази даних задовольняють потреби аналітиків у

більшості застосувань. Однак через зростання обсягів інформації, таких як вебтранзакції та дані, створені машинами, великомасштабною аналітикою стає дедалі важче керувати лише за допомогою системи керування реляційними базами даних (Relational Database Management Systems, RDBMS). Реляційні бази даних зазвичай мають три компоненти: програмне забезпечення для керування базою даних, фізичні сервери, на яких встановлене це програмне забезпечення, і диски, на яких зберігаються дані.

У міру зростання баз даних потрібні більш потужні сервери для обробки та зберігання більших обсягів даних. З традиційною інфраструктурою RDBMS і кількістю даних, що використовують нові застосунки, легко досягти межі, коли навіть дуже потужний сервер і велика кількість сховища не можуть впоратися з обробкою.

Альтернативним рішенням є розподілена обробка в хмарі, яка розбиває великі обсяги даних на менші частини. Ці менші обсяги даних розподіляються між кількома місцями й обробляються багатьма комп'ютерами. Кожен комп'ютер у такій розподіленій архітектурі аналізує лише свою частину даних, розподіляючи навантаження між кількома машинами.

Деякі реляційні бази даних дозволяють такий тип розподілу, але іноді для максимальної ефективності цих систем і/або для керування гнучкими та неструктурованими даними потрібна інша база даних.

NoSQL – це база даних, яка зберігає й отримує дані по-іншому, ніж реляційні бази. NoSQL часто називають "нереляційною", оскільки вона не організовує дані у вигляді таблиць із рядками та стовпцями, що відповідають певній схемі. Натомість дані зберігаються у неструктурованому або напівструктурованому вигляді, що спрощує проектування баз даних.

Є чотири основні типи NoSQL баз даних. Розглянемо по одному прикладу кожного типу, щоб зрозуміти, чим вони відрізняються від реляційних баз даних.

Тип 1. Документні бази даних зберігають документи у форматах, які можуть бути прочитані машиною, таких як JSON, BSON або XML.

Хоча ці бази даних менш ефективні, ніж реляційні, для керування зв'язками між документами, вони набагато краще підходять для читання та пошуку документів. Крім того, структура одного документа не

обов'язково повинна збігатися зі структурою іншого документа в базі даних. Це дозволяє додавати нові елементи в документи без необхідності змінювати таблиці чи схеми (рис. 1.7).

Users		
Document 1	Document 2	Document 3
<pre>{ "id": "1", "name": "Florian Bouron", "isActive": true }</pre>	<pre>{ "id": "2", "fullName": "John Smith", "isActive": false }</pre>	<pre>{ "id": "3", "fullName": { "first": "Sarah", "last": "Stark" }, "isActive": true }</pre>

Рис. 1.7. Приклад фрагменту документної бази даних

Тип 2. Графові бази даних зберігають "вузли", які з'єднані з іншими "вузлами" у мережі. Вузлом може бути, наприклад, користувач у соціальній мережі, що має зв'язки з усіма своїми друзями.

Інший приклад – це точки на карті, які з'єднані в реальному житті, що дозволяє легко знаходити маршрути між ними.

Графові бази даних оптимізовані для швидкого виконання запитів та навігації через ці зв'язки набагато ефективніше, ніж реляційні бази даних, які з'єднують таблиці для пошуку взаємозв'язків. Вони спеціально створені для зберігання великих обсягів взаємопов'язаних вузлів.

На рис. 1.8 зображено фрагмент графової бази даних про фільми, де зображено вузли фільмів (фіолетові), вузли осіб (помаранчеві) та зв'язки між ними (стрілки), що відображають їхню роль у фільмі (наприклад, "ACTED_IN", "DIRECTED", "PRODUCED").

Така структура дозволяє моделювати складні взаємозв'язки, які можуть існувати між багатьма сутностями, що робить її ідеальною для роботи з пов'язаними даними, такими як кіноіндустрія. Перевагою графової бази даних є швидкий доступ до зв'язків між об'єктами та розширення моделі за рахунок додавання нових вузлів чи зв'язків, що ускладнює традиційні реляційні бази даних.

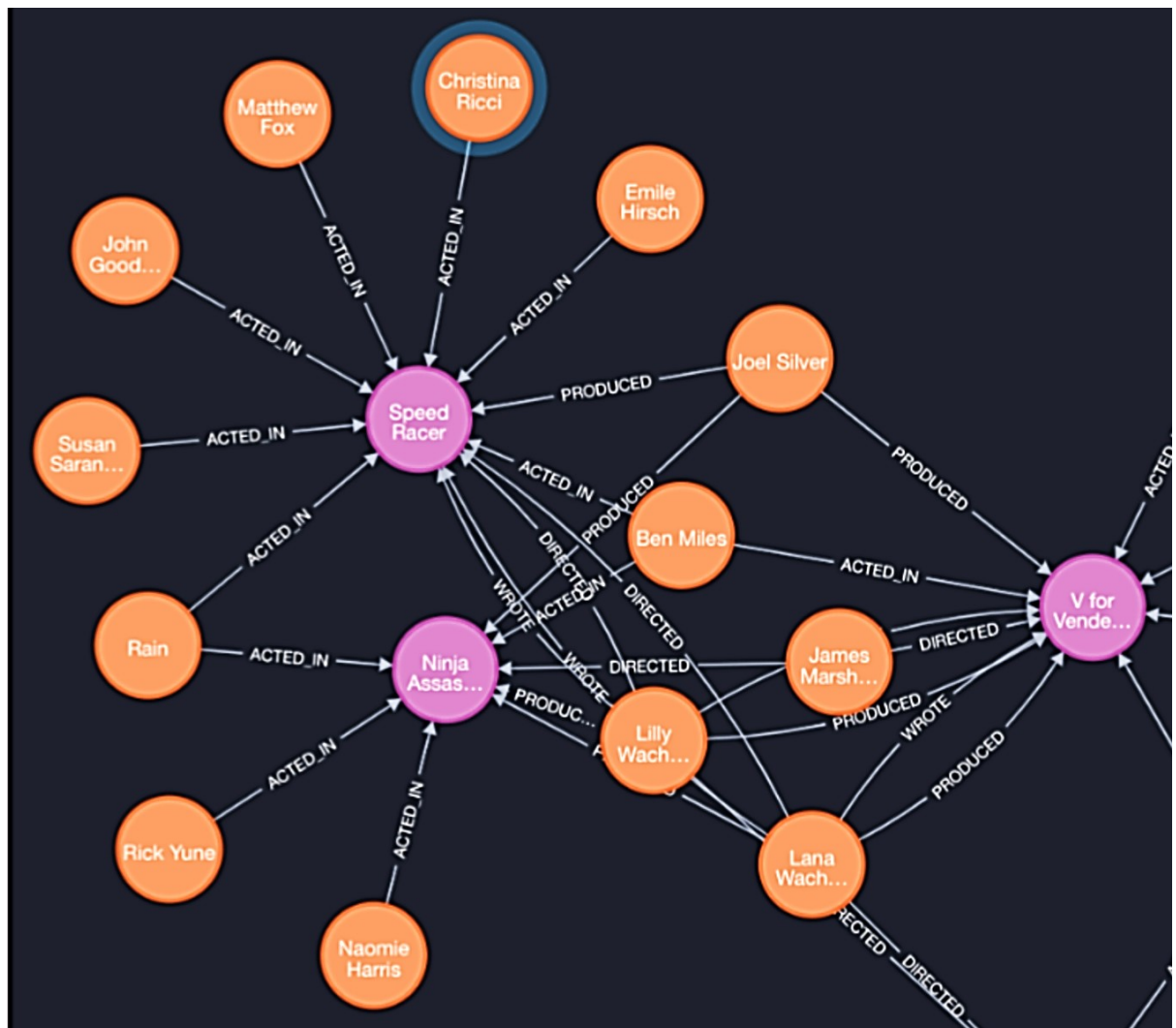


Рис. 1.8. Приклад фрагменту графової бази даних про фільми

Тип 3. Бази даних типу "ключ-значення" зберігають дані у форматі пар: ключ і значення (англ. key-value database або key-value store). Ключ діє як унікальний ідентифікатор, за яким можна швидко знаходити відповідне значення (рис. 1.9). Цей тип баз даних відрізняється простотою та високою швидкістю доступу до даних, оскільки вони не потребують складних операцій для пошуку.

Бази даних типу «ключ-значення» підходять для випадків, коли потрібно зберігати велику кількість простих даних, таких як налаштування конфігурацій або сесійні дані користувачів. Однак вони менш підходять для роботи зі складними запитами або коли дані мають складні взаємозв'язки.