Лекція 2.

Розроблення програмного забезпечення для аналізу веб-сайтів, які надають відкриті дані за допомогою Python Pandas. Відкриті дані, їх формати та засоби оброблення

План лекції

2.1. Можливості інструментів аналізу даних.
2.2. Роль Python в аналізі даних.
2.3. Традиційна аналітика великих даних та аналітика нового покоління.
2.4. Життєвий цикл аналізу даних.
2.5. Відкриті дані, їх формати та засоби обробки.
2.6. Веб-скрепінг.
2.7. Витягування, перетворення та завантаження даних.

2.1. Можливості інструментів аналізу даних

Інструмент, який потрібно використовувати, залежить від типу аналізу, який потрібно виконати. Деякі інструменти призначені для обробки маніпуляцій та візуалізації великих наборів даних. Інші інструменти розроблені з можливостями математичного моделювання для прогнозування. Незалежно від того, які інструменти використовуються, вони повинні відповідати наступним вимогам.

- **Простота використання** – інструмент, який легко вивчити та використовувати, часто є більш ефективним, ніж складний у використанні інструмент. Крім того, простий у використанні інструмент вимагає меншої кількості навчання та підтримки.
- **Маніпулювання даними** – програмне забезпечення повинно дозволяти користувачам очищати та змінювати дані, щоб зробити їх більш корисними. Це призводить до того, що дані є більш надійними, оскільки аномалії можна виявити, відкоригувати або видалити.
- **Спільний доступ** – дослідники повинні використовувати однакові набори даних, щоб мати змогу ефективно співпрацювати та інтерпретувати дані однаково.

- **Інтерактивна візуалізація** – щоб повністю зрозуміти, як змінюються дані з часом, важливо візуалізувати тенденції. Основні діаграми та графіки не можуть повною мірою представляти розвиток інформації так, як може виглядати теплова карта або перегляд руху часу.

2.2. Роль Python в аналізі даних

Існують різноманітні програми, які використовуються для форматування даних, їх очищення, аналізу та візуалізації. Багато компаній та організацій звертаються до інструментів з відкритим кодом для обробки та узагальнення даних. Мова програмування Python стала загальноприйнятим інструментом для оброблення даних.

Мова Python була створена у 1991 році як легка у вивченні мова програмування з багатьма бібліотеками, які використовуються для маніпулювання даними, машинного навчання та візуалізації даних. Завдяки використанню цих бібліотек програмістам не доводиться вивчати декілька мов програмування або витрачати час, вивчаючи, як використовувати різні програми для виконання функцій цих бібліотек. Python – це гнучка мова, яка зростає та стає все більш інтегральною для наукових досліджень завдяки гнучкості та простоті її вивчення.

Розглянемо основні бібліотеки Python для аналізу даних.

- **NumPy** – ця бібліотека додає підтримку роботи з масивами та матрицями. Має багато вбудованих математичних функцій для використання в наборах даних.
- **Pandas** – ця бібліотека додає підтримку таблиць та часових рядів. Використовується для маніпулювання та очищення даних.
- **Matplotlib** – ця бібліотека додає підтримку візуалізації даних. Це бібліотека для простих та складних 3D та контурних графіків.

2.3. Традиційна аналітика великих даних та аналітика нового покоління

До епохи великих даних роль часу в аналітиці даних обмежувалася тим, скільки часу знадобиться для складання набору даних з різних джерел або скільки часу потрібно для запуску набору даних за допомогою певного обчислення. З Big Data час стає важливим і в інших напрямках, оскільки значна частина даних отримується завдяки створенню можливостей негайно вжити заходів.

Дані постійно генеруються датчиками, споживачами товарів і послуг, користувачами соціальних мереж, реактивними двигунами, фондовим ринком і майже всім, що підключено до мережі. Ці дані не просто зростають у кількості, вони змінюються в режимі реального часу, що також вимагає аналізу даних в режимі реального часу під час збору даних.

Під час обговорення Big Data та прийняття рішень бізнесу на основі аналітики може покращити рентабельність інвестицій для бізнесу як функцію часу. Рішення, керовані даними, мають такі переваги:

- збільшено час на дослідження та розробку товарів та послуг;
- підвищення ефективності та швидше виготовлення та вихід на ринок;
- більш ефективний маркетинг та реклама.

Раніше, коли більшість наборів даних були відносно невеликими та керованими, аналітики могли використовувати традиційні засоби, такі як Excel або статистичну програму, таку як SPSS, для аналізу значущої інформації з цих даних. Зазвичай набір даних містив історичні дані, і обробка цих даних не завжди залежала від часу. Дані, якщо вони не надто великі, можна було очистити, відфільтрувати, обробити, узагальнити та візуалізувати за допомогою діаграм, графіків та інформаційних панелей. Зі збільшенням обсягів, швидкості та різноманітності наборів даних складність зберігання, обробки та агрегації даних стає проблемою для традиційних аналітичних інструментів. Великі набори даних можуть поширюватися та оброблятися на декількох географічно розсіяних фізичних пристроях, а також у хмарі. Для цих великих наборів даних потрібні

великі інструменти даних, такі як Hadoop та Apache Spark, щоб забезпечити аналіз у режимі реального часу та прогнозування моделювання.

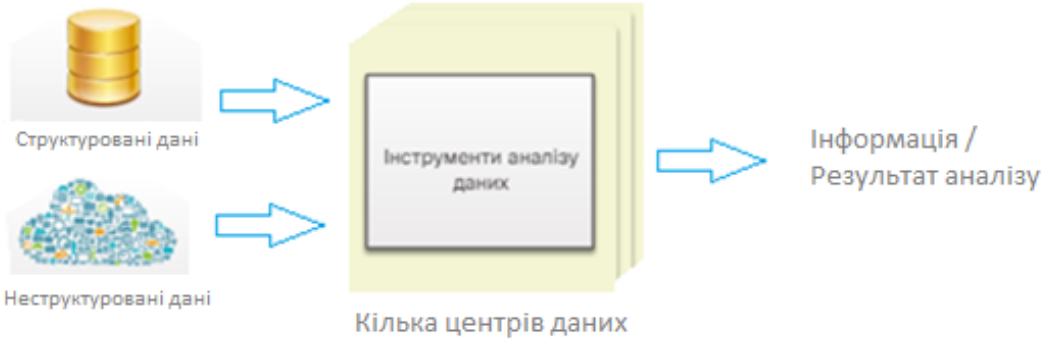

Рис. 2.1. Аналіз великих даних [1]

Для того, щоб підприємства приймали оптимальні рішення, вже недостатньо збирати дані попереднього фінансового року та виконувати описовий аналіз типів запитів. Все частіше необхідно використовувати інструменти прогнозного аналізу, щоб залишатися конкурентоспроможними у світі, у якому швидкість змін прискорюється. Аналітика наступного покоління не повинна покладатися виключно на здійснення статистичної аналітики для всього набору даних, як це робилося з традиційними інструментами. Через величезну кількість точок даних та атрибутів, новітні поведінки та уявлення можна отримати за допомогою вдосконаленого аналізу, що покращує точність прогнозування. Наприклад, можна відповісти на наступні запитання, щоб вносити корективи в реальний час:

• Які акції, швидше за все, матимуть найвищий щоденний приріст на основі торгів за останню годину?

• Який найкращий спосіб перевезти вантажні автомобілі на сьогодні в другій половині дня, виходячи з ранкових продажів, наявних запасів та поточних звітів про рух?

• Яке технічне обслуговування для літака базується на даних про продуктивність, отриманих під час останнього польоту?

Обробка даних, згенерованих машиною, у поєднанні з географічним розмахом дуже масштабних систем, кількістю пристроїв, що генерують дані, різноманітністю виробників пристроїв, частотою генерування даних та загальним

обсягом даних вимагає нового інфраструктурного програмного забезпечення. Це інфраструктурне програмне забезпечення повинне бути здатним розподіляти обчислення та зберігання даних між краєм, туманом та хмарою там, де це краще відповідає потребам бізнесу.

Рис. 2.2. Аналітика нового покоління [1]

Аналітика даних нового покоління дозволяє бізнесу краще розуміти вплив своїх продуктів і послуг, коригувати їх методи та цілі та швидше забезпечувати своїх клієнтів кращими продуктами. Можливість отримувати нові дані зі своїх даних приносить ділову цінність.

2.4. Життєвий цикл аналізу даних

Існує багато методологій проведення аналізу даних, включаючи популярний міжгалузевий стандартний процес обробки даних (CRISP-DM), який використовується більш ніж 40% аналітиків даних. Близько 27% аналітиків даних використовують власну методологію. Решта використовують інші методики. Максимально схожий на науковий метод, життєвий цикл аналізу даних розроблений для використання в бізнес-середовищі. Стрілки спрямовані в обидва боки між деякими кроками. Це підкреслює той факт, що життєвий цикл може зажадати багатьох ітерацій, перш ніж ті, хто приймає рішення, будуть досить впевнені, щоб рухатися вперед.

Рис. 2.3. Життєвий цикл аналізу даних [1]

Як і в науковому методі, життєвий цикл аналізу даних починається з питання. Наприклад, ми можемо задати питання: «Який злочин був найбільш поширеним у Києві, 20 серпня 2015 року?» Кожен крок життєвого циклу аналізу даних включає багато завдань, які необхідно виконати, перш ніж перейти до наступного кроку. Нижче наведено короткий опис кожного кроку.

- **Збір даних** – процес пошуку даних, визначення, чи є достатньо даних для завершення аналізу. Наприклад, ми шукаємо відкритий набір даних про злочинність для Києва протягом серпня 2015 року.
- **Підготовка даних** – цей крок може включати багато задач з перетворення даних у формат, відповідний інструменту, який буде використовуватися. Набір даних про злочини вже може бути підготовлений до аналізу. Однак зазвичай можна внести деякі коригування, які допоможуть відповісти на питання.
- **Вибір моделі** – цей крок включає вибір методики аналізу, яка найкраще відповість на питання із наявними даними. Після вибору моделі вибирається інструмент (або інструменти) для аналізу даних.
- **Аналіз даних** – процес тестування моделі на даних та визначення надійності моделі та аналізованих даних.

- **Представлення результатів** – це, як правило, останній крок для аналітиків даних. Це процес донесення результатів до осіб, які приймають рішення. Іноді аналітику даних просять рекомендувати дії. За даними про злочини 20 серпня, гістограма, кругова діаграма або якесь інше представлення можуть бути використані для повідомлення про те, який злочин найбільш поширений. Аналітик може запропонувати посилити присутність поліції у певних районах, щоб стримувати злочинність у конкретний день, наприклад, 20 серпня.
- **Прийняття рішень** – заключний крок у життєвому циклі аналізу даних. Організаційні лідери включають нові знання як частину загальної стратегії. Процес починається заново зі збору даних.

2.5. Відкриті дані, їх формати та засоби обробки

Є багато різних джерел даних. Велику кількість даних можна знайти у таких файлах, як документи MS Word, електронні листи, електронні таблиці, MS PowerPoints, PDF-файли, HTML та файли простого тексту. Це лише декілька типів файлів, які містять дані. Великі дані також можна знайти в державних та приватних архівах. Скановані паперові архіви, що містять історичні дані з різних джерел, безумовно, є Big Data. Наприклад, існує величезна кількість даних у формах медичного страхування та рахунках-фактурах, ділових виписках та взаємодії з клієнтами, а також у податкових документах. Цей список є лише невеликою частиною архівованих даних.

Внутрішні вихідні дані для організацій створюються за допомогою систем управління взаємовідносинами з клієнтами, систем управління навчанням, систем і записів людських ресурсів, інтрамереж та інших процесів.

Різні програми створюють файли в різних форматах, не обов'язково сумісних один з одним. З цієї причини потрібен універсальний формат файлу. Файли значень, розділених комами (comma-separated values, CSV) – це тип простого тексту, викладений у стандарті RFC 4180.

CSV-файли використовують коми для розділення стовпців таблиці даних, а символ нового рядка – для розділення рядків. Кожен рядок – це запис. Хоча вони

зазвичай використовуються для імпорту та експорту в традиційних базах даних та електронних таблицях, конкретного стандарту немає.

JSON і XML – це також типи файлів у простому тексті, які використовують стандартний спосіб подання записів даних. Ці формати файлів сумісні з широким колом застосувань. Перетворення даних у загальний формат є цінним способом поєднання даних із різних джерел.

Рис. 2.4. Формати даних [1]

Інтернет – хороше місце для пошуку великих даних. Там можна знайти зображення, відео, аудіо. Публічні веб-форуми також створюють дані. Соціальні медіа, такі як YouTube, Facebook, обмін миттєвими повідомленнями, RSS та Twitter, додаються до даних, знайдених в Інтернеті. Більшість цих даних є неструктурованими, що означає, що класифікувати їх в базу даних непросто без певного типу обробки.

2.6. Веб-скрепінг

Веб-сторінки створені для надання інформації людям, а не машинам. Засоби "веб-скрепінгу" автоматично "витягують" дані з HTML-сторінок. Це схоже на веб-сканер або павук пошукової системи, який досліджує Інтернет для вилучення даних та створення бази даних, щоб відповідати на пошукові запити. Програмне забезпечення для скрепінгу веб-сторінок може використовувати протокол передачі гіпертексту або веб-браузер для доступу до мережі. Веб-сканування – це автоматизований процес, в якому використовується бот або веб-сканер. Конкретні

дані збираються та копіюються з Інтернету в базу даних або електронну таблицю. Дані можуть бути проаналізовані.

Для здійснення веб- скрепінгу спочатку потрібно завантажити веб-сторінку, а потім витягти з неї потрібні дані. Веб-скрепери зазвичай виймають щось із сторінки, щоб використовувати його з іншою метою в іншому місці (для пошуку та копіювання імен, номерів телефонів та адрес). Цей процес відомий як контактий скрепінг.

Окрім роботи з контактами, веб-скрепінг використовується для інших видів пошуку даних, таких як списки нерухомості, дані про погоду, дослідження та порівняння цін. Багато великих постачальників веб-сервісів, таких як Facebook, надають стандартизовані інтерфейси для автоматичного збору даних за допомогою API.

Найпоширеніший підхід – використання інтерфейсів прикладних програм RESTful (API). API RESTful використовують HTTP як протокол зв'язку та структуру JSON для кодування даних. Інтернет-сайти, такі як Google та Twitter, збирають велику кількість статичних даних та часових рядів. Знання API для цих сайтів дозволяє аналітикам даних та інженерам отримати доступ до великої кількості даних, які постійно генеруються в Інтернеті.

2.7. Витягування, перетворення та завантаження даних

Бази даних містять дані, вилучені, перетворені та завантажені (ETL – extract, transform, load). ETL – це процес "очищення" необроблених даних, щоб вони могли бути поміщені в базу даних. Дані часто зберігаються в декількох базах даних і повинні бути об'єднані в один набір даних для аналізу.

Більшість баз даних містять дані, які належать організації та є приватними.

Після доступу до даних з різних джерел, дані потребують підготовки до аналізу. Фахівці в галузі Data Science вважають, що підготовка даних може зайняти від 50 до 90 % часу, необхідного для проведення аналізу.

Оскільки дані, які включатимуть набір даних для аналізу, можуть надходити з різноманітних джерел, вони не обов'язково сумісні при їх поєднанні. Інша

проблема полягає в тому, що дані, які можуть бути представлені у вигляді тексту, повинні бути перетворені в числовий тип, якщо вони будуть використані для статистичного аналізу. Типи даних є важливими, коли для роботи з даними використовуються такі мови, як Python або R.

Окрім різних типів даних, один тип даних може бути відформатований по-різному, залежно від його джерела. Наприклад, різні мови можуть використовувати різні символи для представлення одного і того ж слова. Наприклад, британська англійська може використовувати різні написання, ніж американська англійська.

Формати часу та даних представляють складнощі. Хоча час і дати дуже конкретні, вони представлені в найрізноманітніших форматах. Час і дата є важливими для аналізу спостережень за часовими рядами. Тому вони повинні бути перетворені у стандартний формат, щоб аналіз мав якесь значення. Наприклад, дати можуть бути відформатовані у році, який спочатку слідує за днем та місяцем у деяких країнах, тоді як інші країни можуть представляти дані із місяцем, який слідує за днем та роком.

Аналогічно, час може бути представлений у 12-годинному форматі з позначенням AM та PM, або може бути представлений у 24-годинному форматі. Обговорюючи дані, ми можемо думати про ієрархію структур. Наприклад, сховище даних – це місце, яке зберігає безліч різноманітних баз даних таким чином, що можна отримати доступ до баз даних за допомогою тієї самої системи. База даних – це сукупність таблиць даних, які пов'язані одна з одною або декількома способами. Таблиці даних складаються з полів, рядків та значень, схожих на стовпці, рядки та комірки в електронній таблиці. Кожна таблиця даних може розглядатися як файл, а база даних – як колекція файлів. Інші структури даних або об'єкти використовуються Python. Наприклад, Python використовує рядки, списки, словники, кортежі та набори як основні структури даних. Кожна структура даних має власну групу функцій або методів, які можна використовувати для роботи з об'єктом. Крім того, популярна бібліотека аналізу даних Python під назвою "pandas " використовує інші структури даних, такі як дата

фрейми даних. Значна частина даних, які будуть розміщені в базі даних, щоб потім їх можна було запитати, надходить з різних джерел і в широкому діапазоні форматів.

Витягування, перетворення та завантаження (ETL) – це процес збору даних із різноманіття джерел, перетворення даних, а потім завантаження даних у базу даних. Дані однієї компанії можна знайти в документах Word, електронних таблицях, простому тексті, PowerPoints, електронних листах та PDF-файлах. Ці дані можуть зберігатися на різних серверах, які використовують різні формати.

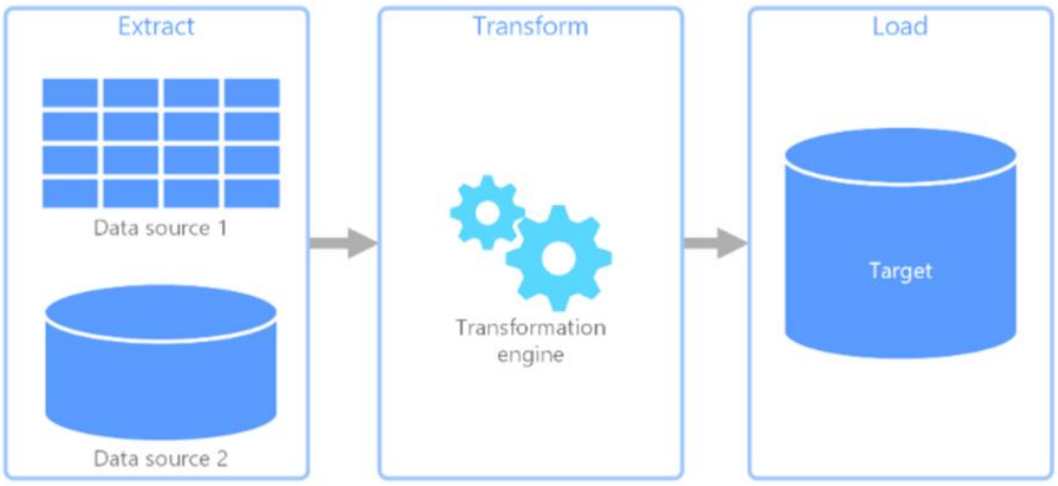

Рис. 2.5. Витягування, перетворення та завантаження даних [2]

Процес ETL містить наступні три основні кроки.

Крок 1. Витягування – дані збираються з кількох джерел.

Крок 2. Перетворення – після того, як дані будуть зібрані, вони повинні бути перетворені. Перетворення даних може включати агрегування, сортування, очищення та приєднання даних.

Крок 3. Завантаження – трансформовані дані завантажуються в базу даних для запитів.

Наведені вище описи трьох етапів процесу ETL спрощені. Насправді, перед тим, як дані можна завантажувати в базу даних, а потім запитувати їх, потрібно виконати дуже багато роботи. Етап вилучення збирає потрібні дані з джерела та робить їх доступними для обробки. Видобування перетворює дані в єдиний формат, який готовий до перетворення.

Наприклад, поєднання даних з сервера NOSQL та Oracle DB надасть дані в різних форматах. Ці дані повинні бути перетворені в єдиний формат. Також дані повинні бути перевірені, щоб переконатися, що вони мають бажаний тип інформації (значення). Це робиться за допомогою правил перевірки. Якщо дані не відповідають правилам перевірки, вони можуть бути відхилені. Іноді ці відхилені дані виправляються та підтверджуються.

Під час вилучення всі необхідні дані з джерела (джерел) отримують за допомогою мінімальних обчислювальних ресурсів, щоб не впливати на роботу мережі або комп'ютера.

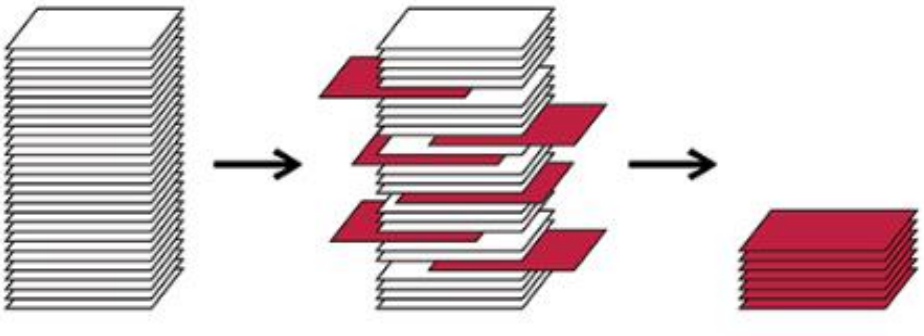

Рис. 2.6. Процес «витягування» даних [1]

На етапі перетворення використовуються правила для перетворення вихідних даних у тип даних, необхідних для цільової бази даних. Сюди входить перетворення будь-яких виміряних даних в один і той же вимір. Крок трансформації також вимагає декількох інших завдань. Деякі з цих завдань – це об'єднання даних із кількох джерел, їх агрегування, сортування, визначення нових значень, які обчислюються з агрегованих даних та застосування правил перевірки. Дані (включаючи деякі відхилені дані) можуть пройти через іншу частину етапу перетворення, відомий як «очищення» даних. Частина очищення етапу трансформації додатково забезпечує узгодженість вихідних даних. Крок завантаження – це коли трансформовані дані завантажуються в цільову базу даних.

Деякі організації можуть перезаписати наявні дані накопичувальними даними. Завантаження нових трансформованих даних може здійснюватися щогодини, щодня, щотижня або щомісяця. Це може статися лише тоді, коли в

трансформованих даних відбулася певна кількість змін. Під час кроку завантаження застосовуються правила, визначені у схемі бази даних. Деякі з цих правил перевіряють унікальність та послідовність даних, поля, які мають обов'язкове володіння, мають необхідні значення тощо. Ці правила допомагають забезпечити успішність завантаження та подальших запитів даних.

Висновок до лекції 2

Дані сьогодні не доцільно зберігати на кількох машинах та обробляти лише одним інструментом. Спеціалісти, що приймають рішення, все частіше покладаються на аналітику даних, щоб витягнути необхідну інформацію в потрібний час, у потрібному місці та прийняти правильне рішення.

Прогнозна аналітика передбачає результати та пропонує курси дій, які матимуть найбільшу користь для організації. Файли, дані з мережі Інтернет, датчики та бази даних – це приклади джерел даних. Витягування, перетворення та завантаження (ETL) – це процес збору даних з різних джерел, перетворення даних, а потім завантаження даних у базу даних.

Питання для закріплення

1. Яка роль Python в аналізі даних?
2. Опишіть традиційну аналітику великих даних та аналітику нового покоління.
3. Опишіть життєвий цикл аналізу даних.
4. Опишіть відкриті дані, їх формати та засоби обробки.
5. Що таке веб-скрепінг?
6. Поясніть процеси витягування, перетворення та завантаження даних.

Список рекомендованої літератури

1. IoT Fundamentals: Big Data & Analytics // Електронний ресурс. Режим доступу: https://www.netacad.com/courses/iot/big-data-analytics
2. Extract, transform, and load (ETL) // Електронний ресурс. Режим доступу: https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl