Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

АПЕПС

Кафедра автоматизації проектування енергетичних процесів і систем

ОБРОБКА ПОТОКОВОЇ ІНФОРМАЦІЇ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Другий (магістерський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем і веб-технологій
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)
Рік підготовки, семестр	1 курс весняний семестр
Обсяг дисципліни	5 кредитів (150 год)
Семестровий контроль/ контрольні заходи	Екзамен
Розклад занять	Науково-педагогічний працівник
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: д.ф.-м.н., с.н.с., Матичин Іван Іванович, matychyn@gmx.ch, тел. 050-872-32-17 Практичні: д.ф.-м.н., с.н.с., Матичин Іван Іванович, matychyn@gmx.ch, тел. 050-872-32-17
Розміщення курсу	Кампус

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчання та результати навчання

Потокові дані можна вважати різновидом великих даних (big data), але звичайні методи і підходи, що застосовуються до пакетної обробки (batch processing) великих даних не можуть бути безпосередньо застосовані для обробки потокових даних (streaming processing), враховуючи їхні особливості. Особливість потокових даних полягає в тому, що дані надходять у вигляді одного або кількох потоків і, якщо не обробити або не зберегти їх негайно, то вони будуть втрачені назавжди. Більше того, в цьому курсі ми припускаємо, что дані надходять так швидко, що практично нереально зберегти їх усі в традиційній базі даних, а обробляти пізніше, коли буде зручно. Це вимагає застосування спеціальних підходів до обробки потокових даних, що реалізують компроміс між точністю, швидкодією та ресурсоємністю.

Метою дисципліни є ознайомлення студентів з основними концепціями, алгоритмами та програмними інструментами для побудови систем обробки потокових даних в режимі реального часу.

Завдання. В результаті вивчення дисципліни у студентів повинні сформуватися наступні компетентності:

загальні:

- здатність до абстрактного мислення, аналізу та синтезу (ЗК 1),
- здатність проводити дослідження на відповідному рівні (ЗК3),

фахові:

- здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення (ФК 1),

- здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення (ФК 2),
- здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах (ФК 7),
- здатність розробляти програмне забезпечення на основі моделей процесів та систем з використанням нечіткої логіки.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі програмні результати навчання:

- будувати і досліджувати моделі інформаційних процесів у прикладній області (ПРН 3),
- виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення (ПРН 4),
- розробляти математичне і програмне забезпечення для наукових досліджень в галузі інженерії програмного забезпечення (ПРН 18),
- знати основні алгоритми аналізу і обробки потокових даних у реальному часі,
- вміти розробляти програмне забезпечення для обробки потокових даних у реальному часі.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Пререквізити дисципліни. Знання та вміння, отримані на попередньому рівні освіти при вивченні дисциплін математичної підготовки, комп'ютерної дискретної математики, програмування, алгоритмів та структур даних.

Постреквізити дисципліни. Отримані при вивченні дисципліни «Обробка потокової інформації» знання формують базові знання для вивчення дисциплін, пов'язаних з моделюванням та розробкою програмного забезпечення для обробки та аналізу великих даних.

3. Зміст навчальної дисципліни

Тема 1. Потокові дані та великі дані
Тема 2. Збір даних
Тема 3. Черга повідомлень
Тема 4. Аналіз потокових даних
Тема 5. Алгоритми аналізу даних
Тема 6. Аналіз часових рядів
Тема 7. Зберігання даних
Тема 8. Доступ до даних

4. Навчальні матеріали та ресурси

Основна література

1. Пселтис Э.Дж. Потоковая обработка данных. М.: ДМК Пресс, 2018. 218 с.
2. Нархид Н., Шапира Г., Палино Т. Apache Kafka. Потоковая обработка и анализ данных. СПб.: Питер, 2019. 320 с.

Додаткова література

3. Лесковец Ю., Раджараман А., Ульман Дж. Анализ больших наборов данных. М.: ДМК Пресс, 2016. 498 с.

4. Saxena S., Gupta S. Practical Real-Time Data Processing and Analytics. Distributed Computing and Event Processing using Apache Spark, Flink, Storm, and Kafka – Birmingham, Mumbai: Packt Publishing, 2017. – 422 с.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Тема 1. Потокові дані та великі дані

Лекція 1. Великі дані та потокова інформація.

Вступ до курсу лекцій. Необхідність паралельних і розподілених обчислень. MapReduce. Hadoop. Apache Spark. Пакетна обробка. Потокова обробка. Масштабування.

Тема 2. Збір даних

Лекція 2. Збір даних

Типові підходи до збору даних: запит-відповідь, запит-підтвердження, видавець-підписник, одностороння взаємодія, потік. Масштабування патернів взаємодії. Відмовостійкість.

Лекція 3. Знайомство з мовою програмування R.

Історія виникнення й основні принципи організації середовища R. Об’єкти і типи даних в R. Експорт/імпорт даних в R. Функції і конструкції в R. Аналіз даних в R. Візуалізація даних в R.

Тема 3. Черга повідомлень

Лекція 4. Черга повідомлень.

Компоненти черги повідомлень. Ізоляція виробників (producers) та споживачів (consumers). Довготривалі повідомлення. Семантика доставки повідомлень. Безпека. Відмовостійкість. Приклади застосування.

Лекція 5. Apache Kafka.

Історія появи Apache Kafka. Встановлення Kafka. Кластери Kafka. Виробники (producers): запис повідомлень в Kafka. Споживачі (consumers): читання даних з Kafka. Відмовостійкість. Реплікація. Створення конвейєрів даних. Адміністрування та моніторинг.

Тема 4. Аналіз потокових даних

Лекція 6. Аналіз даних у русі.

Архітектури розподіленої обробки потоків. Ключові функції систем потокової обробки. Семантика доставки повідомлень. Керування станом. Відмовостійкість.

Лекція 7. Фреймворки обробки потокових даних.

Spark Streaming. Apache Storm. Apache Flink. Apache Samza.

Тема 5. Алгоритми аналізу даних

Лекція 8. Вибірка даних з потоку.

Побудова репрезентативної вибірки. Алгоритм резервуарної вибірки. Загальна постановка задачі про вибірку. Динамічна зміна розміру вибірки.

Лекція 9. Фільтрація потоків та підрахунок різних елементів

Приклад: фільтрація спаму. Фільтр Блума. Аналіз фільтру Блума. Проблема Count-Distinct. Алгоритм Флажоле-Мартена. Покращення точності алгоритму Флажоле-Мартена.

Лекція 10. Оцінювання моментів.

Означення моментів. Алгоритм Алона-Матіаса-Сегеді для моментів другого порядку. Обгрунтування алгоритму Алона-Матіаса-Сегеді. Моменти вищих порядків. Обробка нескінченних потоків.

Лекція 11. Підрахунок одиниць у вікні.

Проблема точного підрахунку. Алгоритм DGIM (Datar-Gionis-Indyk-Motwani Algorithm). Вимоги щодо обсягу пам’яті в алгоритмі DGIM. Дотримання умов алгоритму DGIM. Зменшення похибки. Узагальнення алгоритму підрахунку одиниць.

Лекція 12. Затухаючі вікна.

Задача про найчастіші елементи. Означення затухаючого вікна. Знаходження найпопулярніших елементів.

Тема 6. Аналіз часових рядів

Лекція 13. Аналіз часових рядів.

Означення часового ряду. Стаціонарні випадкові процеси. Білий шум. Моделі тренду і сезонності. Декомпозиція: ідентифікація тренду і сезонності. Моделі стаціонарних випадкових процесів: авторегресійні моделі (AR), моделі рухомого середнього (MA), модель ARMA. Моделі нестаціонарних випадкових процесів: модель ARIMA, випадкове блукання (random walk). Тестування стаціонарності. Прогнозування.

Лекція 14. Аналіз та моделювання часових рядів в R.

Декомпозиція часових рядів: ідентифікація тренду і сезонності. Моделювання процесів ARIMA. Ідентифікація моделі ARIMA. Тестування стаціонарності. Прогнозування.

Тема 7. Зберігання даних

Лекція 15. Зберігання даних.

Довготривале сховище. Зберігання даних в пам'яті. Вбудовані сховища. Системи кешування. Стратегії читання. Стратегії запису. Бази даних та решітки даних в пам'яті.

Тема 8. Доступ до даних

Лекція 16. Доступ до даних.

Патерни взаємодії. Синхронізація. Віддалений виклик методу (RMI) та віддалений виклик процедури (RPC). Простий обмін повідомленнями. Видавець-підписник. Технології відправлення даних клієнтам: веб-хуки, http long-poll, події, що надсилаються сервером, веб-сокети.

Лекція 17. Обробка потокових даних за допомогою R, Spark Streaming та Kafka.

Інтеграція Apache Spark та R. Читання потокових даних в R. Перетворення потокових даних за допомогою R та Apache Spark Streaming. Аналіз потокових даних за допомогою R та Apache Spark Streaming. Організація взаємодії Apache Spark Streaming та Apache Kafka на основі R.

Лекція 18. Патерни та сценарії обробки потокових даних у реальному часі

Попередня обробка. Alerts and Thresholds. Простий підрахунок та підрахунок у вікні. Поєднання потоків. Корельовані дані, відсутні події та помилкові дані. Взаємодія з базами даних. Виявлення закономірностей у послідовності подій. Відстежування. Виявлення трендів. Виконання запиту як у конвеєрі (pipeline) реального часу, так і в пакетному конвеєрі. Виявлення аномалій та перехід до детального аналізу. Використання моделей машинного навчання. Онлайн-керування.

6. Самостійна робота студента

Написання коду продюсера, що імітує потік даних, який генерується в певній предметній області, та надсилає її до певного топіку Apache Kafka.

Написання коду конс'юмера, що отримує потік даних, який генерується продюсером, та здійснює репрезентативну вибірку кортежів потоку.

Програмна реалізація алгоритму резервуарної вибірки (Reservoir Sampling).

Програмна реалізація алгоритму DGIM.

Програмна реалізація алгоритму Флажоле-Мартена.

Обробка потоку даних з Apache Kafka в середовищі R за допомогою Spark Streaming.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Відвідування лекційних та практичних занять є обов'язковим за винятком поважних причин (хвороби, форс-мажорних обставин).

В разі пропущення занять з поважних причин викладач надає можливість студенту виконати усі або деякі завдання практичних занять (винятком є виконання деяких завдань у зв'язку із закінченням навчального процесу).

В разі пропущення занять без поважних причин, а також через порушення граничного терміну виконання завдання (deadline) студент може отримати зменшену кількість балів від максимальної оцінки за відповідне завдання.

Протягом семестру студенти:

- виконують та захищають завдання практичних занять (комп'ютерні практикуми) у відповідні терміни,
- пишуть модульну контрольну роботу,
- повинні позитивно закрити дві атестації (в кінці березня та в середині травня),
- по закінченні навчального процесу складають екзамен.

8. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Система рейтингових (вагових) балів та критерії оцінювання

Максимальна кількість балів з кредитного модуля дорівнює 100.

Рейтинг студента з дисципліни складається з балів, що він отримує за:

- виконання та захист лабораторних робіт,
- дві модульні контрольні роботи (МКР) тривалістю 1 акад. година кожна.
- складання екзамену.

1. <u>Виконання завдань практичних занять</u> (комп'ютерні практикуми)

Завдання практичного заняття являє собою індивідуальне виконання лабораторних робіт, що пов'язані з рішенням на ЕОМ заданої задачі комп'ютерного моделювання.

Вагові бали завдань наведено у таблиці.

Види завдань	*Внесок до семестрового рейтингу балів*
Лабораторна робота №1. Написання коду продюсера, що імітує потік даних, який генерується в певній предметній області, та надсилає її до певного топіку Apache Kafka.	5
Лабораторна робота №2. Написання коду конс'юмера, що отримує потік даних, який генерується продюсером, та здійснює репрезентативну вибірку кортежів потоку.	5
Лабораторна робота №3. Програмна реалізація алгоритму резервуарної вибірки (Reservoir Sampling).	5
Лабораторна робота №4. Програмна реалізація алгоритму DGIM.	5
Лабораторна робота №5. Програмна реалізація алгоритму Флажоле-Мартена.	5
МКР №1	10
Лабораторна робота №6. Обробка потоку даних з Apache Kafka в середовищі R за допомогою Spark Streaming.	5
МКР №2	10

Максимальна кількість балів за всі завдання дорівнює 50 балів.

Критерії оцінювання

Підготовка до роботи (у відсотках від максимальної кількості балів за відповідну роботу):

- протокол відповідає вимогам, охайний – 20 %;
- протокол відповідає вимогам, але є чисельні виправлення – 10 %;

Виконання завдання:

- робота виконана повністю і вірно протягом відведеного часу – 50 %;
- робота виконана пізніше зазначеного терміну – 20 %;

Якість захисту роботи:

- студент вірно і повністю відповів на запитання – 30 %;
- студент при відповіді допустив несуттєві неточності – 20 %;
- студент при відповіді на запитання допустив суттєві неточності, але самостійно виправив їх – 10 %.

2. Модульний контроль

Ваговий бал – 10 × 2.

Контрольна робота складається з 10 тестових завдань. За кожну вірну відповідь на запитання надається 1 бал.

3. Екзамен

Ваговий бал – 50.

Сума вагових балів контрольних заходів протягом семестру складає:

R = 30 +20+ 50 = 100 балів.

Необхідною умовою допуску до екзамену є зарахування усіх завдань практичних занять, а також стартовий рейтинг (r_c) не менше 40% від R, тобто 40 балів.

Сума балів переводиться до екзаменаційної оцінки згідно з таблицею:

Бали (RD)	Традиційна оцінка
95..100	Відмінно
85…94	Дуже добре
75…84	Добре
65…74	Задовільно
60…64	Достатньо
RD<=60	Незадовільно
RD < 40 або не виконані інші умови допуску до екзамену	Не допущений

Робочу програму навчальної дисципліни (силабус):

Складено професор, д.ф.-м.н., с.н.с., Матичин Іван Іванович

Ухвалено кафедрою _____ (протокол № ___ від _____)

Погоджено Методичною комісією факультету[1] (протокол № __ від _____)

[1] Методичною радою університету – для загальноуніверситетських дисциплін.