

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ВІЗУАЛЬНОГО ПРОГРАМУВАННЯ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ

Марценюк Є.О.¹⁾, Бережний О.І.²⁾, Мацьків В.І.³⁾, Микитюк В.А.⁴⁾

*Західноукраїнський національний університет
1)к.т.н., доцент; 2)магістрант; 3)аспірант; 4)аспірант*

I. Постановка проблеми

Сучасний штучний інтелект (ШІ) у багатьох оперує моделями на даних, які будуються методами машинного навчання. Для роботи з такими моделями потрібні особливі спеціалісти – аналітики даних (datascientist), які не є повноцінними програмістами або математиками, але часто добре розбираються в предметах, для яких необхідно побудувати моделі і вирішувати на їх основі прикладні задачі. Запит на таких спеціалістів постійно росте, що мотивується в першу чергу зростаючим обсягом генерованих даних і великим потенціалом їх використання [1,2]. В цих умовах потрібні спеціалісти, які вміють вирішувати відповідний клас задач за розумний час з прийнятними для замовника результатами, які можна об'єктивно підтвердити.

II. Мета роботи

Метою дослідження є розробка методів та засобів для аналізу та тестування програмного коду з використанням генетичного алгоритму.

III. Загальна архітектура системи

Web-технології є природним вибором для створення програмної системи, що реалізує запропоновані принципи. Важливо при цьому враховувати, що моделювання в галузі машинного навчання - це досить ресурсомісна задача, що вимагає потужні обчислювальні ресурси. Виходячи з цього, було прийнято рішення реалізувати мікросервісну архітектуру, яка поєднуватиме гнучкі можливості веб-інтерфейсів, інструмент у вигляді сервера для зберігання та взаємодії та хмари обчислень для навчання моделей (див.рис.1). Така архітектура дозволяє поєднувати безліч необхідних технологій, що включають: web-технології для управління робочим процесом, інструменти для опису та зберігання бази даних, бібліотеки для моделювання та обробки даних, модулі для розподілених і відкладених обчислень.

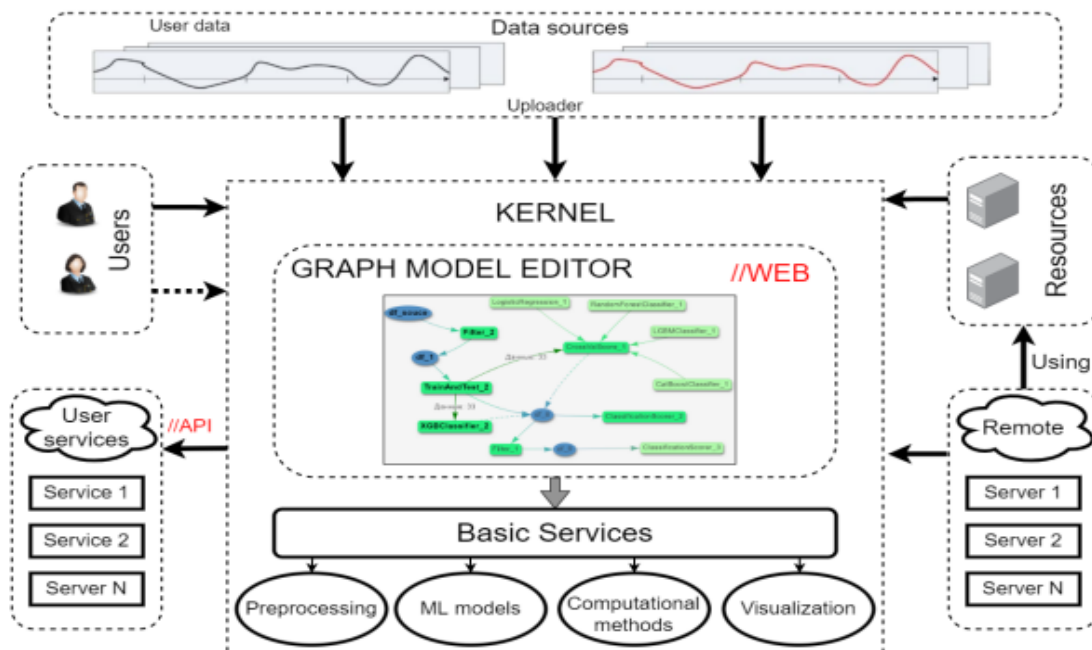


Рисунок 1 –Архітектура системи візуальної розробки моделей машинного навчання

Ядром програмної системи є веб-сервер, який містить та оперує базовими сервісами: управлінням обчислювальних процесів, організацією розподілених обчислень для вирішення завдань

у галузі машинного навчання, обміном даними з зовнішніми програмами за допомогою API, візуалізацією даних та іншими допоміжними засобами, такими як бази даних, черги та ін. На окремі обчислювальні сервери розгорнули мікро-сервіси для виконання важких обчислень, які організують запуск процесів та контролюють черговість та пріоритет завдань залежно від ресурсів. Web-сервіс, обмінюючись повідомленнями з обчислювальними серверами, організує розподіл завдань обчислення серверів залежно від навантаження кожного з них. Реалізація ядра платформи побудована на засадах об'єктно-орієнтованого програмування. Кожна математична абстракція описується класом з програмним кодом. Таким чином, було створено інтерфейси для головних сутностей: обчислювального графа, вузла обчислювального графа та ребра обчислювального графа. Для кожного інтерфейсу визначається основна поведінка об'єкта. Поведінка графа визначається методами обходу графа в ширину та глибину, прямого доступу до вузлів та ребрах, а також ініціалізації пріоритету обходу вузлів. Поведінка інтерфейсу вузла задається у методах для процедур навчання, трансформації та застосування. Рисунок 2 містить діаграму класів архітектури обчислювального графа.

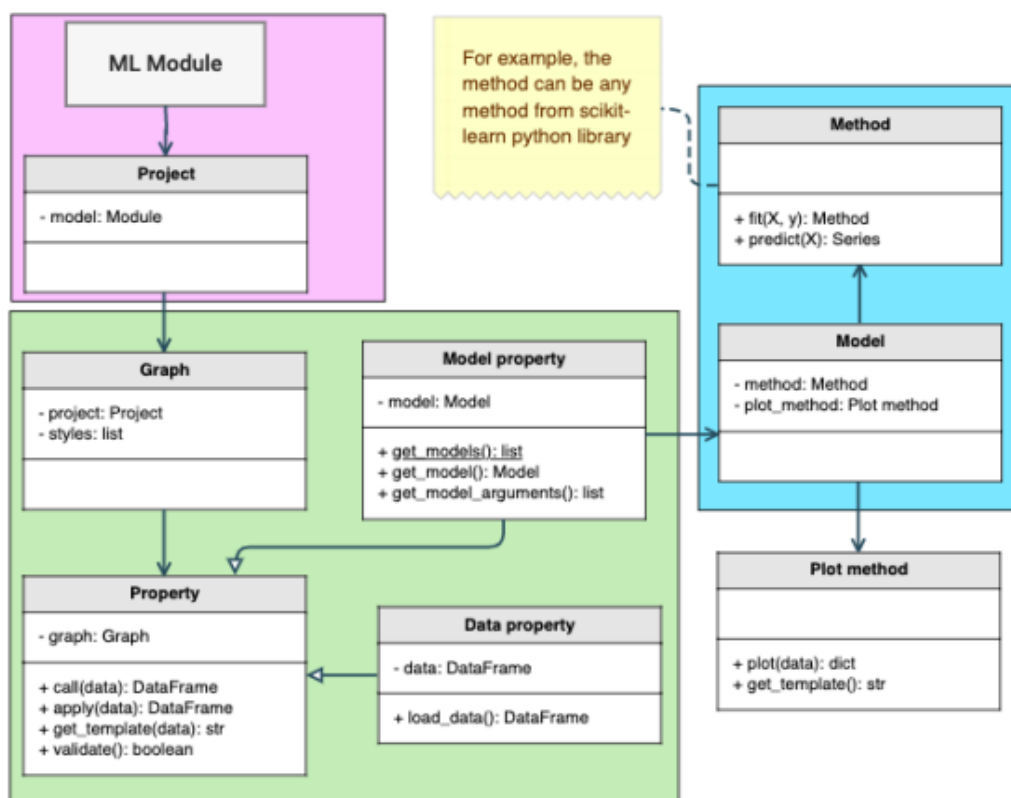


Рисунок 2 – Архітектура ядра веб-сервера

Поведінка вузлів даних та моделей відрізняється. Вузлам даних характерна робота з базою даних для підвантаження наборів даних із бази даних або з альтернативних джерел, а також операції фільтрації чи відображення. Вузли графа виконують роботу з вибору певної моделі, а також з параметрами та поведінкою обраної моделі. Відмінність у поведінці визначається за допомогою інтерфейсу якості. Він ділиться на властивість моделі та властивість даних. Властивість даних має знання про наявні дані та оперує базою даних. Властивість моделі має знання про наявні моделі, а також про роботу з параметрами кожної з них.

Висновок

Розроблений інструмент був апробований на різних прикладних задачах різних предметних областей, а також у галузі освіти. За результатами досліджень можна зробити висновок, що розроблений засіб корисний як для розробників із штучного інтелекту, так і для галузевих фахівців, які не мають впевнених навичок з програмування та DataScience.

Список використаних джерел

1. , Christopher M. Pattern Recognition and Machine Learning. Springer, 2006. ISBN: 978-0387310732.
2. Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron. Deep Learning. MIT Press, 2016. ISBN: 978-0262035613.