

Список використаних джерел

1. Математичні основи криптографії: URL: <https://cutt.ly/SwYzQnGY> (дата звернення: 13.11.2023).
2. Wikipedia. URL: <https://cutt.ly/lwYzQ2mz> (дата звернення: 13.11.2023).
3. Оптимальне кодування. URL: <https://cutt.ly/gwYzWSOm> (дата звернення: 13.11.2023).
4. Математична криптографія та оптимальне кодування. URL: <https://cutt.ly/gwqzMSDa> (дата звернення: 13.11.2023).

УДК 004.8

Бурківський О. С., здобувач 2 курсу спеціальності 122 Комп'ютерні науки, Зелінська О. В., канд. техн. наук, доцент, в. о. завідувача кафедри інформаційних технологій

СУЧАСНІ ФРЕЙМВОРКИ МАШИННОГО НАВЧАННЯ

Донецький національний університет імені Василя Стуса, м. Вінниця

Сучасний розвиток технологій у галузі машинного навчання визначається більшою мірою використанням потужних фреймворків. Ці інструменти не лише прискорюють розробку моделей, а й надають розробникам та дослідникам гнучкість і можливості для ефективного впровадження інтелектуальних систем. У публікації ми розглянемо кілька ключових сучасних фреймворків машинного навчання.

TensorFlow є високопродуктивним відкритим вихідним фреймворком для машинного та глибокого навчання, розробленим і підтримуваним Google. Запущений у 2015 р., він завоював широку популярність завдяки своїй гнучкості, ефективності та розширеною підтримкою глибокого навчання.

Особливості TensorFlow:

1. TensorFlow дає змогу розробникам вибирати між статичною та динамічною обчислювальною графікою. Це забезпечує гнучкість під час вибору оптимального підходу до розробки моделей.
2. Вона також має великий набір заздалегідь вбудованих шарів для різних видів мереж, а також оптимізаторів для тренування моделей із різною складністю.
3. Підтримка роботи на різноманітних апаратних платформах дає змогу ефективно використовувати ресурси для тренування великих моделей.
4. TensorFlow має активну та велику спільноту користувачів та розробників. Це забезпечує доступ до різноманітних ресурсів, зокрема документації, навчальних матеріалів та багатофункціональних модулів.

5. TensorFlow широко використовується у наукових дослідженнях, оскільки надає зручний інтерфейс для реалізації та тестування нових ідей у галузі машинного навчання. Багато компаній використовують TensorFlow для розробки та впровадження своїх продуктів, оскільки він забезпечує потужність і масштабованість [1, 2].



Рисунок 1. Логотип фреймворка «TensorFlow»

Scikit-learn є високорівневою бібліотекою для машинного навчання, розробленою на мові програмування Python. Вона надає простий і ефективний інтерфейс для роботи з різноманітними алгоритмами машинного навчання, що робить її популярною серед дослідників, студентів та інженерів з області даних.



Рисунок 2. Логотип фреймворка «Scikit-learn»

Особливості Scikit-learn:

1. Scikit-learn ставить перед собою мету забезпечити чіткий та однорідний інтерфейс для різних методів машинного навчання, що допомагає швидко розробляти й експериментувати з моделями.

2. Бібліотека передбачає широкий набір алгоритмів для класифікації, регресії, кластеризації, вимірювання важливості ознак та багато інших завдань машинного навчання.

3. Scikit-learn надає інструменти для оптимізації параметрів моделей, крос-валідації та оцінки їх продуктивності. Це допомагає забезпечити надійні та оптимальні результати.

4. Scikit-learn є популярним інструментом для вивчення основ машинного навчання завдяки своїй простоті та інтуїтивному інтерфейсу. Широкий набір

методів дає змогу використовувати бібліотеку для аналізу даних та прогнозування різноманітних явищ. Їх легко інтегрувати з іншими бібліотеками Python, як-от Matplotlib та Seaborn, що допомагає створювати візуалізації результатів та аналізу даних [3].

Сучасні фреймворки машинного навчання відкривають нові можливості для розробників та дослідників, даючи змогу їм ефективно впроваджувати та вдосконалювати моделі. Вибір конкретного фреймворка залежить від завдань, типів моделей, а також особистих вподобань. Загальна тенденція полягає в тому, що розвиток машинного навчання значною мірою зумовлений появою потужних та гнучких фреймворків, які продовжують визначати та трансформувати цю захоплюючу галузь.

Список використаних джерел

1. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. URL: <https://www.tensorflow.org/about/bib> (дата звернення: 06.12.2023).

2. Introduction to TensorFlow. URL: <https://www.tensorflow.org/learn> (дата звернення: 06.12.2023).

3. Scikit-learn: machine learning in Python – scikit-learn 1.3.2 documentation. URL: <https://scikit-learn.org/stable/index.html> (дата звернення: 06.12.2023).

УДК 004.021

*Бурківський О. С., здобувач 2 курсу спеціальності 122 Комп'ютерні науки,
Потапова Н. А., канд. екон. наук, доцент, доцент кафедри інформаційних
технологій*

ЗАСТОСУВАННЯ ДИНАМІЧНОГО ПРОГРАМУВАННЯ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ПРО НАЙДОВШУ ПОСЛІДОВНІСТЬ

Донецький національний університет імені Василя Стуса, м. Вінниця

Задача про найдовшу послідовність є однією з класичних задач алгоритмізації, яка виникає в різних контекстах, як-от обробка рядків, генетика, аналіз текстів тощо. Через це використання методів динамічного програмування може суттєво полегшити розв'язання проблеми та забезпечити ефективний алгоритм для пошуку найдовшої послідовності.

Динамічне програмування – це ефективний метод розв'язання складних задач, який має бути застосований у різних галузях, зокрема алгоритмізації, оптимізації та штучному інтелекті. Основні ідеї динамічного програмування включають: