

КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

УДК 004.738.5

DOI <https://doi.org/10.32782/2521-6643-2023.2-66.6>

Поперешняк С. В., кандидат фізико-математичних наук, доцент,
доцент кафедри інформатики та програмної інженерії
Національного технічного університету України
«Київський політехнічний інститут імені Ігоря Сікорського»
ORCID: 0000-0002-0531-9809

Вєчерковська А. С., кандидат технічних наук,
доцент кафедри програмних систем і технологій
Київського національного університету імені Тараса Шевченка
ORCID: 0000-0003-2054-2715

ЦИФРОВІЗАЦІЯ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ

У статті проведено аналіз розвитку сучасних технологій штучного інтелекту. Проаналізовано основні сучасні рішення цифровізації комп'ютерних систем. Досліджено ключові проблеми розвитку даної концепції та шляхи їх вирішення, одним з яких є технологія цифрової імунної системи, наведено передумови її виникнення та сучасний стан. Досліджено взаємозв'язки основних характеристик сучасних рішень цифровізації. Проаналізовано переваги, які дає використання рішень на основі штучного інтелекту при вирішенні широкого спектра завдань, що виникають в реальному житті. Представлено теоретичну основу для аналізу рекомендацій щодо забезпечення кібербезпеки критичних інфраструктур на основі штучного інтелекту.

Зроблено розробку рішення по автоматичному аналізу LOG файлів на основі штучного інтелекту з використанням хмарної платформи Azure та інтерфейсу користувача Airflow.

Штучний інтелект та цифровізація комп'ютерних систем – дві технології, які мають великий потенціал, обидві вже широко використовуються. Поєднання штучного інтелекту та цифровізації комп'ютерних систем є актуальним, тому що може бути способом аналізу недоліків у застосуванні процесів і технологій.

Ключові слова: штучний інтелект, цифровізація, комп'ютерна інженерія, кібербезпека інфраструктур.

Popereshnyak S. V., Vecherkovskaya A. S. Digitization of computer systems based on artificial intelligence

The article investigated the use of artificial intelligence technology. It was determined that for the technology of artificial intelligence and machine learning there is no development of the role of people in the coexistence and use of the capabilities of the system of artificial intelligence and machine learning. We mean the systemic point of view of the interdependence of man and system. However, soon there will be a large number of systems using artificial intelligence technologies to help us with medical treatment, provide reliable transportation and support us in decision-making.

It is obvious that the opportunities provided by modern technologies of artificial intelligence and machine learning have both their advantages and disadvantages. It has been investigated that the main advantages of systems with artificial intelligence support are: the possibility of dynamism of the learning process and adaptation, the potential of compatibility, which significantly affects the learning process. The disadvantages include the following: the main problem of artificial intelligence technology is that there is currently no generally accepted approach to implementation.

The article analyzes the development of modern artificial intelligence technologies. The main modern solutions for digitalization of computer systems are analyzed. The key problems of the development of this concept and ways to solve them are studied, one of which is the technology of the digital immune system, the prerequisites for its emergence and the current state are given. The interrelationships of the main characteristics of modern digitalization solutions have been studied. The advantages of using solutions based on artificial intelligence in solving a wide range of tasks that arise in real life are analyzed. The theoretical basis for the analysis of recommendations for ensuring cyber security of critical infrastructures based on artificial intelligence is presented.

A solution was developed for automatic analysis of LOG files based on AI using the Azure cloud platform and the Airflow user interface.

Artificial intelligence and digitization of computer systems are two technologies that have great potential, both are already widely used. The combination of artificial intelligence and digitization of computer systems is relevant, because it can be a way to analyze shortcomings in the application of processes and technologies.

Key words: artificial intelligence, digitalization, computer engineering, cyber security of infrastructures.

Постановка проблеми. Комп'ютерна інженерія переживає цифрову трансформацію. Ця еволюція призведе до подальших трансформаційних досягнень у використанні технологій штучного інтелекту та машинного навчання для автоматизації багатьох рутинних інженерних завдань. Водночас застосування штучного інтелекту, машинного навчання та автономії до складних і критичних систем заохочує нові методи, процеси та інструменти комп'ютерної інженерії. Вкрай важливо, щоб працівники глибоко розуміли нові технології та принципи штучного інтелекту та машинного навчання, включали їх у методи та інструменти таким чином, щоб покращити дисципліну комп'ютерної інженерії, і гарантували, що відповідні підходи системної інженерії використовуються для того, щоб зробити системи штучного інтелекту етичними та надійними, надійно та безпечно.

У категорії робочої сили та культури багато інженерів походять з фундаментальних дисциплін інженерії та не мають деяких основ інформатики, які керують сферою дисциплін штучного інтелекту. Спеціалісти комп'ютерної інженерії потребують подальшого розвитку базових цифрових інженерних компетенцій у розробці та інженерії програмного забезпечення, інженерії даних та пов'язаних інформаційних технологіях. Системи штучного інтелекту та машинного навчання створюються в цих трьох дисциплінарних областях. Проте комп'ютерна інженерія може створити міцну основу в міждисциплінарних підходах до спільноти штучного інтелекту. З часом інструменти розробки штучного інтелекту включатимуть абстракції та шаблони проектування, які зроблять технологію більш доступною для широкого кола інженерів, покращуючи міждисциплінарне розуміння та використання технології. Чіткою проблемою розвитку робочої сили є інтеграція штучного інтелекту з системною інженерією та інтеграцією людських систем – набагато більше представлення когнітивних наук і когнітивної інженерії в наборі дисциплін комп'ютерної інженерії. Спеціальні системотехнічні вимоги, такі як безпека та надійність, повинні вийти на перший план. Нові підходи до тестування та оцінювання для навчання та адаптації значно вплинуть на ці дисципліни.

Аналіз останніх досліджень і публікацій. Однією з головних і важливих проблем для реалізації перспектив ШІ/МН є те, що наразі не існує загальноновизнаного підходу до реалізації, оскільки набагато більша частка реалізацій, реалізованих на сьогодні, є вузькоспеціалізованими [1]; є й інші виклики, і щодо них існує багато думок [2]. Розглянемо деякі з важливих питань, пов'язаних з розумінням поточного стану справ, і того, що можна зробити, щоб досягти певного прогресу в методах комп'ютерної інженерії. Один з оглядів був здійснений у [3], де розглядалося питання щодо належного рівня включення комп'ютерної інженерії як функції типу організації, яка виконує роботу, і, зокрема, щодо організації типу R&D. Можна стверджувати, що така суворість в науково-дослідних організаціях часто не є виправданою або недосяжною, особливо там, де немає впевненості, чи зможе нова технологія досягти ключових цілей ефективності. В [4], визначено три рівні комп'ютерної інженерії для підтримки правильного розміру системної інженерної діяльності: неформальний, напівформальний і формальний. В [5] переконані, що розробка вимог для систем машинного навчання є особливим завдяки іншій парадигмі, яка використовується для розробки рішень на основі даних. Вони аналізують вплив машинного навчання на розробку вимог, описуючи вплив на виявлення вимог, аналіз, специфікацію та перевірку та валідацію. Подібним чином [6] розробили еквівалентну оцінку за цими самими вимірами для викликів розробки вимог у випадку систем штучного інтелекту. Вони рекомендують орієнтований на ціль підхід до розробки вимог, який намагається зрівноважити неточність формулювання мети з точністю специфікації вимог. Додаткові документи, спрямовані на розробку вимог, стосуються визначення правової та етичної ефективності систем штучного інтелекту та машинного навчання [7]. Таким чином, ці важливі аспекти комп'ютерної інженерії залишаються на стадії вивчення.

У статті [8], обговорюються різні проблеми щодо технічної заборгованості, пов'язані з розробкою програмного забезпечення штучного інтелекту та машинного навчання. Публікація [9] фокусується на перешкодах підходу до оцінювання, орієнтованого на здібності, коли система характеризується своїми когнітивними здібностями, а не завданнями, для вирішення яких вона призначена.

Як бачимо, ці дослідження не враховують повністю технічні та технологічні можливості цифровізації комп'ютерних систем, тому дослідження щодо використання штучного інтелекту та машинного навчання в галузі комп'ютерної інженерії потребують більш глибокого підходу.

Мета статті: визначення перспектив та потенційних переваг використання технологій штучного інтелекту і підвищення ефективності комп'ютерної інженерії.

Виклад основного матеріалу дослідження. Дослідницька рада SERC розробила дорожню карту для структурування та керівництва дослідженнями штучного інтелекту та автономності. Ця дорожня карта була описана в [10] і представлений на низці форумів, включаючи обидві системні дисципліни [11], [12] та дисциплін штучного інтелекту [13]. Спеціальний семінар «SE4AI/AI4SE», спонсорований SERC та армією США, ще більше вдосконалив дорожню карту. Початкова версія була представлена на конференції Асоціації з розвитку штучного інтелекту (ARПП) восени 2020 року, а поточна версія буде опублікована в 2021 році в посібнику INCOSE AI для фахівців з KI. Ця дорожня карта публікується з метою пов'язати дисципліну комп'ютерної інженерії з різними тенденціями в штучному інтелекті та його застосуванням для автоматизації систем. Цей зв'язок надається як засіб для обговорення можливої еволюції технологій штучного інтелекту та машинного навчання, автономії та дисципліни комп'ютерної інженерії з часом. Рисунок 1 зображує поточну умовну дорожню карту.

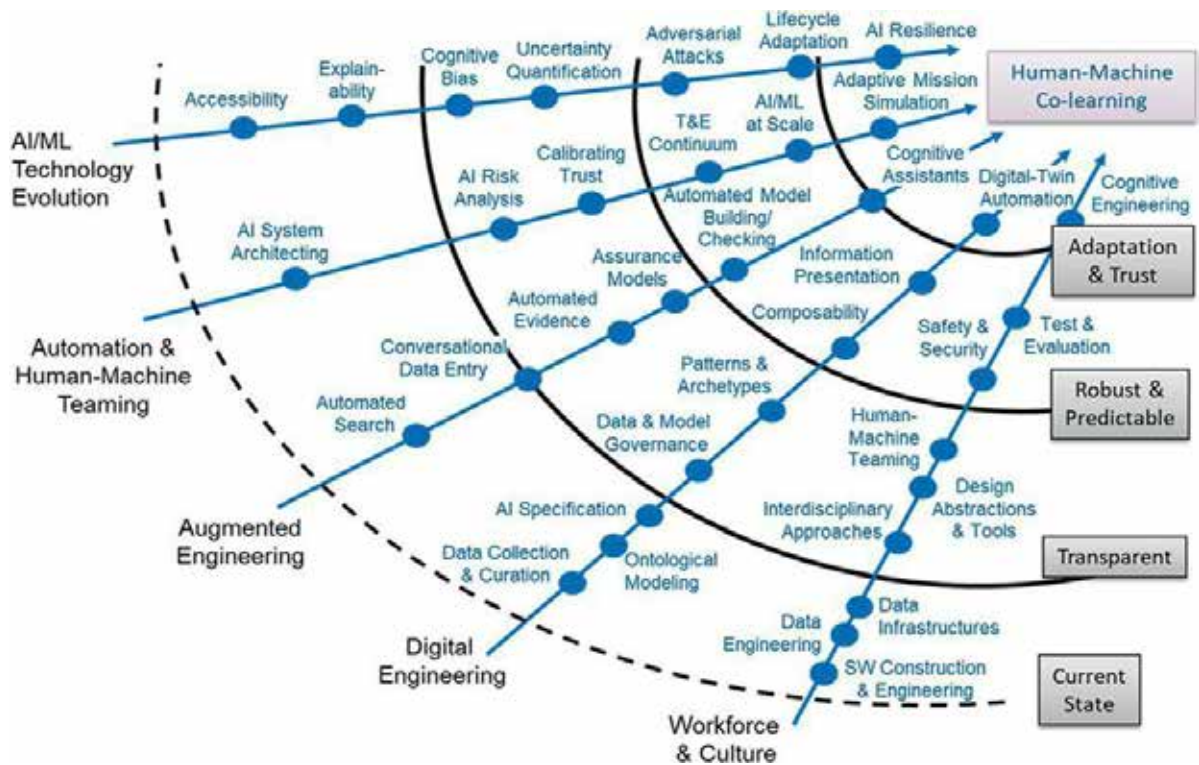


Рис. 1. Дорожня карта SERC AI4SE та SE4AI

Розглянемо еволюцію технології штучного інтелекту для цифрової комп'ютерної інженерії.

Цифрова інженерія є основою для перших трьох векторів дорожньої карти. Під час інженерного моделювання та аналізу буде зібрано більше основних даних, вони стануть навчальними даними для програм машинного навчання.

Є кілька поточних досягнень у цифровій інженерії, пов'язаних з AI4SE [10]:

1. Таксономії та онтології інструментів і доменів – інженерні та програмні дані отримують взаємодію завдяки онтологіям, орієнтованим на домен. Графові бази даних для пов'язаних даних стають все більш помітними в інструментах системної інженерії на основі моделей. Таксономії є відправною точкою для побудови онтологій, що зрештою дає змогу обґрунтувати базові дані на основі ШІ. Цей прогрес є трансформаційною інфраструктурою в AI4SE.

2. Інтеграція даних між підприємствами – основною метою цифрової інженерії є авторитетне джерело правдивих даних, які лежать в основі різних дій з розробки та управління програмами в складних інженерних системах. Оскільки програми та інженерно-конструкторська діяльність обмінюються даними, підприємства створюватимуть великі набори даних для передачі знань і повторного використання в різних програмах і проектах. Ці дані будуть доступні для автоматизації пошуку, побудови/перевірки моделі та прийняття рішень.

3. Семантичні правила в інженерних інструментах – засновані на представленнях знань, таких як онтології, семантичні правила забезпечать основу для міркувань (за допомогою ШІ) щодо повноти та узгодженості інженерних моделей.

4. Цифрова подвійна автоматизація (Digital Twin Automation) – розроблені системи підтримуватимуться двійниками – повністю динамічними віртуальними копіями систем, створеними з тих самих моделей, що й реальні системи, і працюють паралельно з фізичними системами. Дані про дизайн і збірку системи оновлюватимуться з тих самих каналів даних, що й їхні реальні аналоги. Цей динамічний процес створює початкову інфраструктуру для спільного навчання людини та машини.

Розробка рішення по автоматичному аналізу log файлів на основі штучного інтелекту

Рішення на основі штучного інтелекту по автоматичному аналізу LOG файлів повинно:

- максимально автоматизувати аналіз логів;
- обробляти усі події у наданому контексті аналізу;
- бути прозорим для людини (аналітика);
- мати можливість адаптуватися до певних налаштувань мережі (технології, топології мережі та інше);
- і, звичайно, самовдосконалюватися.

Як результат це рішення забезпечує легкість аналізу LOG файлів і скорочує час і людські зусилля. Автоматично розпаковує пакети LOG файлів і витягує з них попередньо визначені шаблони в таблиці. Зберігає всі дані LOG файлів та таблиці на централізованій або на хмарній платформі. Система усує несправності на основі правил і машинного навчання.

Рішення застосовує масштабовану архітектуру на основі мікросервісів та хмарні технології на базі Kubernetes та Gcloud Composer.

Для керування робочими процесами зі складними завданнями, аналізу помилок завдань, повторними спробами та залежностями задіяний дуже потужний інструмент – Apache Airflow [14].

Для автоматичного аналізу LOG файлів треба виконувати величезну кількість коротких запитів. Усі ці запити протягом години мають бути завершені, перш ніж можна буде запустити наступну годину. Як наслідок, потрібно стежити за успішністю чи невдачею запитів.

В Таблиці 1 наведені дані по максимальній кількості користувачів та вказана максимальна кількість оброблених пакетів кожним користувачем.

Це десь 2400 оброблених пакетів за годину. Вважаємо, що треба запустити 1000 запитів за менше ніж 15 хвилин, щоб було достатньо буфера для повторних невдалих запитів.

У рішенні застосовуються наступні моделі аналізу.

1. **Частотна модель:** ненормальну поведінку можна визначити як значне відхилення частоти тривоги від середнього (наприклад, простого або експоненціального) за допомогою стандартного порогу відхилення або верхньої межі IQR.

2. **Модель затримки очищення:** вивчення звичайної затримки очищення кожного аварійного сигналу та встановлення порогового значення. Очікується ненормальна поведінка, якщо затримка скасування тривоги перевищує встановлений поріг.

3. **Модель правил асоціації:** вивчення послідовностей тривог, які завжди з'являються разом перед ініціюванням інциденту тривоги.

4. **Текстова модель:** вивчення того, що повинно/не повинно бути в певному LOG файлі комп'ютера за допомогою підходів LSTM і вбудовування слів (тобто нечастий текст у LOG файлах вважається ненормальним).

5. **Модель помилкової аварії:** вивчення того, що має бути/не має бути в певному файлі журналу аварій за допомогою підходів логістичної регресії та вбудовування слів (тобто нечастий текст у файлах журналу аварій вважається ненормальним).

Таблиця 1

Вхідні дані для розрахунків ресурсів хмарної платформи Azure

	Локація № 1	Локація № 2	Локація № 3
Максимальна кількість користувачів за годину	20	15	20
Середня кількість користувачів за годину	8	8	8
Оброблено пакетів користувачем за годину, максимальна кількість	120	120	120
Оброблено пакетів користувачем за годину, середня кількість	8	8	8
Розмір пакета (extracted), максимальний	1.2GB	1.2GB	1.2GB
Розмір пакета (extracted), середній	450MB	450MB	450MB

Зробимо розрахунки ресурсів хмарної платформи Azure з використання інтерфейсу користувача Airflow, які треба задіяти для рішення на основі штучного інтелекту по автоматичному аналізу LOG файлів.

Таблиця 2 це перелік безкоштовних ресурсів для виконання одного завдання (обробка одного пакету будь-якого розміру).

Таблиця 2

Безкоштовні ресурси для виконання одного завдання

Розмір пакета	0-200 MB	201-400 MB	401-600 MB	≥ 601 MB
Driver core	5	5	5	8
EXEC core	4	4	5	7
EXEC вимог	3	4	6	6
Spark cluster / History Server	2	2	2	2

Загальна кількість EXEC core (TC_{EXEC}) така:

$$TC_{EXEC} = N_{EXEC} \times C_{EXEC} \quad (1)$$

де N_{EXEC} – це кількість EXEC вимог, а C_{EXEC} – це кількість EXEC core, дивись значення з Таблиці 2.
Загальний об'єм пам'яті (T_{MEM}) задається як:

$$T_{MEM} = (4 \times N_{DRIVE} + 4 \times TC_{EXEC}), \quad (2)$$

де N_{DRIVE} – це кількість Driver core, дивись значення з Таблиці 4.2.
Загальна кількість core (TC) така:

$$TC = (C_{DRIVE} + TC_{EXEC} + SC), \quad (3)$$

де SC – це кількість Spark cluster / History Server, дивись значення з Таблиці 2.

Виконуючи рівняння (1), (2) та (3), отримуємо загальну кількість ресурсів для обробки пакету будь-якого розміру (Таблиця 3).

Таблиця 3

Загальні ресурси для обробки пакету будь-якого розміру

Розмір пакета	0-200 MB	201-400 MB	401-600 MB	≥ 601 MB
Total EXEC core	12	16	30	42
Total Memory	68	84	140	200
Total core	19	23	37	52

В Таблиці 4 приведені дані розподілу пакетів по розмірам, які один користувач обробив за годину.

Таблиця 4

Кількість пакетів оброблених користувачем за годину

Розмір пакета	0-200 MB	201-400 MB	401-600 MB	≥ 601 MB
Оброблено пакетів користувачем за годину, максимальна кількість	37	37	37	9
Оброблено пакетів користувачем за годину, середня кількість	3	2	2	1

Загальна кількість core для одного користувача (TC_{USER}) така:

$$TC_{USER} = (P_{MAX} \times TC). \quad (4)$$

або

$$TC_{USER} = (P_{AVR} \times TC). \quad (5)$$

де P_{MAX} – це максимальна кількість пакетів оброблених користувачем за годину, а P_{AVR} – це середня кількість пакетів оброблених користувачем за годину, дивись значення з Таблиці 4.

Виходячи з рівнянь (4) та (5), можна обчислити загальну кількість core для одного користувача для обробки пакетів усіх розмірів.

Таблиця 5

Загальна кількість core для одного користувача

Розмір пакета	0-200 MB	201-400 MB	401-600 MB	≥ 601 MB
Для максимальної кількості пакетів за годину	703	851	1369	468
Для середньої кількості пакетів за годину	57	46	74	52

Загальна кількість core для кожної локації (TC_{LOC}) така:

$$TC_{LOC} = N_{USERMAX} \times \sum MAX TC_{USER} \quad (6)$$

або

$$TC_{LOC} = N_{USERAVR} \times \sum AVR TC_{USER} \quad (7)$$

де $N_{USERMAX}$ – це максимальна кількість користувачів за годину, а $N_{USERAVR}$ – це середня кількість користувачів за годину, дивись значення з Таблиці 1.

Виходячи з рівнянь (6) та (7), можна обчислити загальну кількість соге для кожної локації.

Таблиця 6

Загальна кількість соге для кожної локації

	Локація № 1	Локація № 2	Локація № 3
Для максимальної кількості користувачів за годину	67820	50865	67820
Для середньої кількості користувачів за годину	1832	1832	1832

Таким чином, зроблено розробку рішення по автоматичному аналізу LOG файлів на основі штучного інтелекту з використанням хмарної платформи Azure та інтерфейсу користувача Airflow.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі. У даній статті було досліджено технології штучного інтелекту. Визначено, що поки-що для технології штучного інтелекту та машинного навчання відсутні розробки ролі людей у співіснуванні та використанні можливостей системи штучного інтелекту та машинного навчання, тут мається на увазі системна точка зору взаємозалежності людини та системи. Але незабаром буде задіяна велика кількість систем, які допоможуть нам з медичним лікуванням, зможуть забезпечити надійне транспортування та підтримати нас під час прийняття рішень.

Очевидно, що можливості, які дають сучасні технології штучного інтелекту та машинного навчання, мають як свої переваги, так і недоліки.

Досліджено, що основними перевагами систем з підтримкою штучного інтелекту є: можливість динамізму процесу навчання та адаптації, потенціал сумісності, що значно впливає на процес навчання.

До недоліків можна віднести наступне: основна проблема технології штучного інтелекту полягає в тому, що наразі немає загальноприйнятого підходу до реалізації.

Проаналізовано, що технології штучного інтелекту та машинного навчання будуть продовжувати розвиватися та вдосконалюватися для підтримки більшої кількості функцій та розширення сфер впливу.

Зроблено розробку рішення по автоматичному аналізу LOG файлів на основі штучного інтелекту з використанням хмарної платформи Azure та інтерфейсу користувача Airflow. Проте перевагою використання технології штучного інтелекту є можливість об'єднати потужність кількох ключових ідей, щоб допомогти керувати дедалі складнішими системами з підтримкою штучного інтелекту.

Окреслено, що взаємозв'язок штучного інтелекту з людством робить світ більш інтегрованим та має все більший вплив на кожну частину нашого життя, починаючи від машинного навчання до розумного моніторингу здоров'я та навколишнього середовища.

Список використаних джерел:

1. Vora T. Design thinking for AI: Sustainable AI solution design. 2019. URL: <https://www.cuelogic.com/blog/design-thinking-for-ai> (дата звернення 20.11.2023).
2. Software Engineering for AI-Enabled Systems (SE4AI). CMU 17-445/645, Summer 2020. URL: <https://scaestne.github.io/seai> (дата звернення 20.11.2023).
3. Lombardo N., Millard D., Sturges M. A systems engineering framework for R&D organizations. *INCOSE International Symposium*. 2015. Vol. 25, No 1. P. 1020–1034.
4. Anderson N., Nolte W. Systems engineering principles applied to basic research and development. Session Title: Systems Analysis and Systems Engineering. URL: <https://repository.gatech.edu/server/api/core/bitstreams/aa7c17ee-96cd-4eef-9144-5e5bf6d31b59/content> (дата звернення 20.11.2023).
5. Vogelsang A., Borg M. Requirements Engineering for Machine Learning: Perspectives from Data Scientists. // 2019 IEEE 27th International Requirements Engineering Conference Workshops (REW). Jeju, Korea (South), 2019. pp. 245–251. DOI: 10.1109/REW.2019.00050.
6. Belan H., Vuković M., Car Z. Requirements engineering challenges in building AI-based complex systems. 2019. URL: [arXiv:1908.11791](https://arxiv.org/abs/1908.11791) (дата звернення 20.11.2023).
7. Guizzardi R., Amaral G., Guizzardi G., Mylopoulos J. Ethical requirements for AI systems. *Advances in artificial intelligence*. 2020. pp. 251–256. DOI: 10.1007/978-3-030-47358-7_24.
8. Kruchten P., Nord R., Ozkaya I. Technical debt: From metaphor to theory and practice. *IEEE Software*, 2012. Vol. 29 No 6. pp. 18–21.
9. Hernández-Orallo, J. Evaluation in artificial intelligence: From task-oriented to ability-oriented measurement. *Artificial Intelligence Review*, 2017. Vol. 48. P 397–447. URL: <https://doi.org/10.1007/s10462-016-9505-7>.
10. McDermott T., DeLaurentis D., Beling P., Blackburn M., Bone, M. AI4SE and SE4AI: A research roadmap. *IncoSE Insight*, 2020. Vol. 23 No 1. P. 8–14.
11. McDermott T. A framework to guide AI/ML and autonomy research in systems engineering. *Systems and Mission Engineering*. 2019. Vol. 2. P. 234–241.

-
12. McDermott T. Digital engineering and AI - Transformation of systems engineering. *International Council on Systems Engineering*. 2020. Vol. 1. P. 78–85.
 13. McDermott T. Digital engineering and AI. *AI welcomes systems engineering: Towards the science of interdependence for autonomous human-machine teams*. 2020. Vol. 3. P. 163–171.
 14. Chan K. Scaling Airflow to 1000 tasks/hour on Google Cloud Composer and Kubernetes, 2020. URL: <https://medium.com/@keozchan/-airflow-to-1000-tasks-hour-aac3207b26ec> (дата звернення 20.11.2023).

References:

1. Vora T. Design thinking for AI: Sustainable AI solution design. 2019. URL: <https://www.cuelogic.com/blog/design-thinking-for-ai>.
2. Software Engineering for AI-Enabled Systems (SE4AI). CMU 17-445/645, Summer 2020. URL: <https://ckaestne.github.io/seai>.
3. Lombardo N., Millard D., Sturges M. A systems engineering framework for R&D organizations. *INCOSE International Symposium*. 2015. Vol. 25, No 1. P. 1020–1034.
4. Anderson N., Nolte W. Systems engineering principles applied to basic research and development. Session Title: Systems Analysis and Systems Engineering. URL: <https://repository.gatech.edu/server/api/core/bitstreams/aa7c17ee-96cd-4eef-9144-5e5bf6d31b59/content>.
5. Vogelsang A., Borg M. Requirements Engineering for Machine Learning: Perspectives from Data Scientists. // 2019 IEEE 27th International Requirements Engineering Conference Workshops (REW). Jeju, Korea (South), 2019. pp. 245–251. DOI: 10.1109/REW.2019.00050.
6. Belan H., Vuković M., Car Z. Requirements engineering challenges in building AI-based complex systems. 2019. URL: [arXiv:1908.11791](https://arxiv.org/abs/1908.11791) (дата звернення 20.11.2023).
7. Guizzardi R., Amaral G., Guizzardi G., Mylopoulos J. Ethical requirements for AI systems. *Advances in artificial intelligence*. 2020. pp. 251–256. DOI: 10.1007/978-3-030-47358-7_24. [in English].
8. Kruchten P., Nord R., Ozkaya I. Technical debt: From metaphor to theory and practice. *IEEE Software*, 2012. Vol. 29 No 6. pp. 18–21. [in English].
9. Hernández-Orallo, J. Evaluation in artificial intelligence: From task-oriented to ability-oriented measurement. *Artificial Intelligence Review*, 2017. Vol. 48. P. 397–447. URL: <https://doi.org/10.1007/s10462-016-9505-7>. [in English].
10. McDermott T., DeLaurentis D., Beling P., Blackburn M., Bone, M. AI4SE and SE4AI: A research roadmap. *IncoSE Insight*, 2020. Vol. 23 No 1. P. 8–14. [in English].
11. McDermott T. A framework to guide AI/ML and autonomy research in systems engineering. *Systems and Mission Engineering*. 2019. Vol. 2. P. 234–241. [in English].
12. McDermott T. Digital engineering and AI - Transformation of systems engineering. *International Council on Systems Engineering*. 2020. Vol. 1. P. 78–85. [in English].
13. McDermott T. Digital engineering and AI. *AI welcomes systems engineering: Towards the science of interdependence for autonomous human-machine teams*. 2020. Vol. 3. P. 163–171. [in English].
14. Chan K. Scaling Airflow to 1000 tasks/hour on Google Cloud Composer and Kubernetes, 2020. URL: <https://medium.com/@keozchan/-airflow-to-1000-tasks-hour-aac3207b26ec>. [in English].