



НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
«КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

# НАУКОВІ ЗАПИСКИ НаУКМА КОМП'ЮТЕРНІ НАУКИ

Том 6 ♦ 2023

Науковий журнал ♦ Щорічник ♦ Заснований у 1996 р.

Київ  
2023

**Національний університет «Києво-Могилянська академія»  
заснував видання «Наукові записки НаУКМА» 1996 року**

«Наукові записки НаУКМА. Комп'ютерні науки» (англ. NaUKMA Research Papers. Computer Science) — науковий рецензований журнал відкритого доступу, що висвітлює результати наукових досліджень у галузі комп'ютерних наук та інформаційних технологій.

До 2018 р. виходив друком як частина багатосерійного видання «Наукові записки НаУКМА» (серія «Комп'ютерні науки»). З 2018 р. — «Наукові записки НаУКМА. Комп'ютерні науки». Виходить раз на рік.

Пропонований том висвітлює творчі здобутки вчених, аспірантів, студентів факультету інформатики Національного університету «Києво-Могилянська академія», а також фахівців з інших наукових центрів України, що співпрацюють з НаУКМА в галузі інформатики, кібернетики, програмування, за останній рік. Дослідження авторів статей спрямовані на розв'язання теоретичних проблем і практичних завдань у галузі сучасної інформатики.

Журнал адресовано науковцям, фахівцям, викладачам, докторантам, аспірантам, студентам, а також усім зацікавленим у наукових дослідженнях у галузі комп'ютерних наук та інформаційних технологій.

**Мови видання:** українська, англійська

**Редакційна колегія**

*Глибовець Микола Миколайович*, доктор фізико-математичних наук, професор кафедри інформатики (НаУКМА) — головний редактор

*Кириєнко Оксана Валентинівна*, старший викладач кафедри інформатики (НаУКМА) — відповідальний секретар  
*Анісімов Анатолій Васильович*, доктор фізико-математичних наук, професор, декан факультету кібернетики (Київський національний університет імені Тараса Шевченка)

*Буй Дмитро Борисович*, доктор фізико-математичних наук, професор, старший науковий співробітник (Київський національний університет імені Тараса Шевченка)

*Глибовець Андрій Миколайович*, доктор технічних наук, доцент, декан факультету інформатики (НаУКМА)

*Горlach Сергій*, Dr. rer. nat., професор (Університет Мюнстера, Німеччина)

*Дорошенко Анатолій Юхимович*, доктор фізико-математичних наук, професор (НаУКМА)

*Іваненко Віктор Іванович*, доктор технічних наук, професор (НТУУ «КПІ» імені Ігоря Сікорського)

*Кондратенко Юрій Пантелійович*, доктор технічних наук, професор (Чорноморський державний університет імені Петра Могили)

*Малашонко Геннадій Іванович*, доктор фізико-математичних наук, професор, завідувач кафедри мережних технологій (НаУКМА)

*Терещенко Василь Миколайович*, доктор фізико-математичних наук, професор, завідувач кафедри математичної інформатики, професор (Київський національний університет імені Тараса Шевченка)

*Фісун Микола Тихонович*, доктор технічних наук, професор (Чорноморський державний університет імені Петра Могили)

*Шкільняк Степан Степанович*, доктор фізико-математичних наук, професор кафедри теорії та технології програмування (Київський національний університет імені Тараса Шевченка)

**Здійснюється подвійне анонімне рецензування матеріалів**

**Засновник і видавець:**

Національний університет  
«Києво-Могилянська академія»

Свідectво про державну реєстрацію  
друкованого ЗМІ  
Серія КВ № 23188-13028Р від 22.03.2018

Внесено до Переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук і ступеня доктора філософії, категорія «Б» (наказ МОН України від 02.07.2020 № 886)

## ПЕРЕДМОВА

*Пропонуємо вашій увазі новий випуск наукового журналу «Наукові записки НаУКМА. Комп'ютерні науки». Цей збірник відображає творчі досягнення вчених, аспірантів і магістрів факультету інформатики Національного університету «Києво-Могилянська академія», а також провідних науковців із різних наукових центрів України, які взаємодіють з нашим університетом у галузі інформатики, кібернетики та програмування.*

*У статтях цього випуску розглянуто теоретичні аспекти кібернетики і програмування, а також практичні застосування програмних технологій у галузі сучасної інформатики та прикладної математики. Цей номер є результатом досліджень наших учених і втілює наше прагнення збагатити та розширити горизонти знань у цих стратегічних галузях.*

*Висловлюємо щирю вдячність керівництву НаУКМА за їхню щедру фінансову та організаційну підтримку. Це видання стало можливим завдяки їх допомозі та вірі у значущість наших досліджень.*

*До публікації у збірнику редакційною колегією рекомендовано 12 статей за такими тематичними напрямками: програмна інженерія; комп'ютерні науки.*

*Щиро дякуємо всім авторам за співпрацю.*

Голова редколегії збірника  
М. М. Глибовець

Олецький О. В., Франчук І. О., Гуминський В. В.

## ПРО ПІДХІД ДО ФОРМУВАННЯ ДВОРІВНЕВОЇ МОДЕЛІ «СТАН-ІМОВІРНІСТЬ ДІЇ» НА ОСНОВІ ПОПАРНИХ ПОРІВНЯНЬ ТА МЕТОДУ АНАЛІЗУ ІЄРАРХІЙ

У рамках підходу до моделювання процесів, пов'язаних із недетермінованим прийняттям рішень на основі моделі «стан-імовірність дії», запропоновано підхід до побудови матриць «стан-імовірність дії», що базується на попарних порівняннях. Розглянуто вибір між двома альтернативами на основі розгляду багатьох критеріїв, які можуть суперечити один одному. Показано зв'язок із добре відомою дворівневою схемою методу аналізу ієрархій.

Наведено ілюстративний приклад, на якому показано як ситуацію рівноваги альтернатив, так і можливість відходу від рівноваги за рахунок зміни матриць попарних порівнянь.

**Ключові слова:** ранжування альтернатив, модель «стан-імовірність дії», рівновага альтернатив, попарні порівняння, метод аналізу ієрархій.

### 1. Вступ

У роботах [7; 11–14] розвивається формалізований підхід до моделювання процесів індивідуального та колективного недетермінованого прийняття рішень, який можна охарактеризувати як модель «стан-імовірність дії».

Нехай є  $n$  альтернатив, які утворюють множину  $A = \{a_1, \dots, a_n\}$ , і агент має прийняти рішення про вибір однієї з цих альтернатив. Вводиться випадкова величина  $\xi$ , яка змістовно означає альтернативу, яку вибирає агент у ситуації прийняття рішень. Вводиться система станів  $S = \{s_1, \dots, s_m\}$ , у яких може перебувати агент; існує значна свобода у заданні як самої системи станів, так і їх кількості  $m$ . Розглядається випадкове блукання між станами, в рамках якого моделюється колективне прийняття рішень на основі індивідуальних виборів окремих агентів. Вводиться також випадкова величина  $\eta$  — стан, у якому перебуває агент.

У базовому, найпростішому випадку кожний стан просто задає розподіл ймовірностей, з якими вибирається та чи та альтернатива. З системою станів  $S$  пов'язується матриця «стан-імовірність дії»  $H = \{h_{ij}\}$ , де  $h_{ij} = P(\xi = a_j | \eta = s_i)$ ,  $i = \overline{1, m}, j = \overline{1, n}$  — імовірність того, що агент, який перебуває в  $i$ -му стані, вибере  $j$ -ту альтернативу. Матриця  $H$  належить до класу, який отримав назву прямокутних стохастичних матриць [1; 11], тобто матриць, сума елементів кожного рядка яких дорівнює 1.

Розглядається також вектор  $\bar{p} = (\bar{p}_1, \dots, \bar{p}_m)$ , де  $\bar{p}_i = P(\eta = s_i)$  — ймовірність перебування агента в  $i$ -му стані. Замість явного задання цих ймовірностей їх можна отримати на основі аналізу деякого марковського ланцюга переходів між станами.

Ставиться задача отримання остаточних ймовірностей прийняття рішень, які утворюють вектор  $p = (p_1, \dots, p_n)$ , де  $p_j = P(\xi = a_j)$  — імовірність прийняття  $j$ -го рішення. Як показано в [11],

$$P(\xi = a_j) = \sum_{i=1}^m P(\eta = s_i) \cdot P(\xi = a_j | \eta = s_i), i = \overline{1, m}, j = \overline{1, n}$$

або у векторно-матричному вигляді

$$p = \bar{p}H.$$

Можливі стани моделі «стан-імовірність дії» і, відповідно, елементи матриці  $H$  можуть обиратися досить довільно, і в описаному вище базовому варіанті вони не мають зрозумілої змістовної інтерпретації. Важливо розвивати загальні принципи, на основі яких можуть будуватися матриці «стан-імовірність дії». Крім того, в [7] запропоновано підхід до побудови більш структурованих багаторівневих моделей «стан-імовірність дії» на основі розгляду пов'язаних систем станів.

Приймаючи рішення, часто доводиться оперувати неформальними інтуїтивними уявленнями про те, що деяка альтернатива є кращою або гіршою за іншу. Ця обставина дає можливість говорити про можливість застосування попар-

них порівнянь і методу аналізу ієрархій (МАІ) [17; 2 та ін.], що базується на них. У цій статті запропоновано підхід до побудови багаторівневих моделей «стан-імовірність дії» на основі парних порівнянь.

Слід зауважити, що найважливішим є випадок двох альтернатив ( $n = 2$ ). У статті буде передусім розглянуто саме цей випадок, якщо не буде явно зазначено щось інше.

## 2. Парні порівняння та матриці «стан-імовірність дії»

Матриця парних порівнянь (МПП) — це  $(n \times n)$ -матриця  $M = (m_{ij}, i, j = \overline{1, n})$ , яка описує відношення переваг між альтернативами. Нагадаємо, що через  $n$  ми позначаємо кількість альтернатив. Зазвичай приймається, що якщо  $i$ -та альтернатива краща за  $j$ -ту (позначається  $a_i > a_j$ ), то  $m_{ij} > 1$ . Якщо альтернативи рівнозначні ( $a_i \sim a_j$ ), то  $m_{ij} = 1$ . Якщо ж  $i$ -та альтернатива гірша за  $j$ -ту (позначається  $a_i < a_j$ ), то  $m_{ij} < 1$ . МПП зазвичай будують експертним шляхом, і експерти часто формулюють свої уподобання в словесній формі.

Існують різні підходи до побудови МПП, і з цим пов'язана низка серйозних питань [17; 3 та ін.]. Класична шкала градацій, запропонована Т. Сааті (1 — варіанти рівнозначні, 2 і 3 — слабка перевага і т. д.), не завжди призводить до гарних результатів. Деякою мірою це можна пояснити органічною проблемою, пов'язаною з мультиплікативним характером порівнянь. Зокрема, якщо перевагу варіанта А над варіантом В оцінюють як 2, то мультиплікативність неявно означає, що варіант А має бути кращим за В у 2 рази, а це далеко не завжди відповідає уявленню про слабку перевагу. Набули поширення транзитивні шкали [3 та ін.], для яких наступна градація переваг більша за попередню в  $\tau$  разів;  $\tau$  — деяке задане число. З цим, хоч і не лише з цим, тісно пов'язане логарифмічне подання МПП, для якого в МПП фігурують не самі оцінки переваг, а їх логарифми за тією чи іншою основою.

Транзитивну шкалу з параметром  $\tau$  називатимемо  $\tau$ -транзитивною шкалою. Тоді міру переваги однієї альтернативи над іншою можна оцінювати як цілочисельну кількість градацій між ними. Наприклад, якщо перевагу 1-ї альтернативи над 2-ю оцінити як мінімальну (1 градація), а 2-ї над 3-ю як трохи сильнішу (2 градації), то матриця переваг у термінах градацій (в логарифмічному поданні) може мати вигляд [13]

$$\begin{pmatrix} 0 & 1 & 3 \\ -1 & 0 & 2 \\ -3 & -2 & 0 \end{pmatrix}.$$

Повна ж МПП при застосуванні  $\tau$ -транзитивної шкали запишеться у вигляді

$$\begin{pmatrix} 1 & \tau & \tau^3 \\ \frac{1}{\tau} & 1 & \tau^2 \\ \frac{1}{\tau^3} & \frac{1}{\tau^2} & 1 \end{pmatrix}.$$

Більш загально, якщо  $c_{ij}$  — перевага  $i$ -го варіанта над  $j$ -м у термінах градацій, то відповідний елемент МПП при застосуванні  $\tau$ -транзитивної шкали обчислюється як [13]

$$m_{ij} = \tau^{c_{ij}}.$$

Слід зазначити, що в логарифмічній шкалі рівнозначність альтернатив — це 0, а не 1.

Міркування про можливість застосування парних порівнянь, зокрема на основі транзитивних шкал, до побудови матриць «стан-імовірність дії», пов'язані з природним уявленням про те, що можливі стани моделі «стан-імовірність дії» можуть відповідати кількостям градацій переваг однієї альтернативи над іншою.

## 3. Системи станів і матриця «стан-імовірність дії» на основі градацій переваг

У рамках запропонованого підходу розглядатимемо наступну систему станів із параметром  $q$ , який змістовно означає максимальну кількість градацій переваг (нагадаємо, що йдеться про випадок двох альтернатив, назвемо ці альтернативи  $A1$  та  $A2$ ):

$A1$  краще за  $A2$  на  $q$  градацій;

$A1$  краще за  $A2$  на  $q-1$  градацій;

...

$A1$  краще за  $A2$  на 1 градацію;

$A1$  та  $A2$  рівнозначні (чисельне значення в логарифмічній шкалі — 0).

Для відношення «гірше» чисельні значення кількостей градацій розставляють симетрично, але зі знаком мінус.

Вибір параметра  $q$ , очевидно, має істотне значення. Його збільшення має призводити до більш гнучких і тонко диференційованих систем станів. З іншого боку, в разі збільшення  $q$  різниця між сусідніми станами стає менш помітною, і експертам стає важче провести диференціацію і дати точну оцінку ступеня переваги.

На основі описаної системи станів легко отримати матрицю «стан–імовірність дії», якщо використовувати транзитивну шкалу з тим чи іншим параметром. Основна частина матриці формується на основі такого алгоритму:

- для всіх  $k$  від  $q$  до  $-q$ ;
- отримуємо вектор  $v = (v_1, v_2)$  з компонентами

$$v_1 = \tau^k, v_2 = \tau^{-k};$$

- вектор  $v$  нормалізується так, щоб сума його компонент дорівнювала 1;
- результуючий вектор стає черговим рядком матриці «стан–імовірність дії».

Крім того, як перший і останній рядки матриці «стан–імовірність дії» доцільно взяти рядки, які відповідають визначальній перевазі однієї альтернативи над іншою (відповідні рішення приймаються з імовірністю 0 або 1).

Побудована таким чином матриця «стан–імовірність дії» за  $q = 3$ ,  $\tau = 1.3$  має вигляд (наближено):

$$H = \begin{pmatrix} 1 & 0 \\ 0.8284 & 0.1716 \\ 0.7407 & 0.2593 \\ 0.6283 & 0.3717 \\ 0.5 & 0.5 \\ 0.3717 & 0.6283 \\ 0.2593 & 0.7407 \\ 0.1716 & 0.8284 \\ 0 & 1 \end{pmatrix}$$

Отримана матриця  $H$  належить до класу так званих центросиметричних матриць, опис яких можна знайти в [20; 9]. Аналіз центросиметричних матриць має суттєве значення для дослідження симетричних ситуацій рівноваги альтернатив [11], тобто ситуацій, коли жодна з альтернатив не має переваги над іншими.

Проілюструємо на цьому ж прикладі обчислення ймовірностей прийняття рішень за умови, що відомий вектор  $\bar{p}$ , який задає ймовірності перебування агента у вищеписаних станах. Нехай

$$\bar{p} = (0.5, 0.1, 0.15, 0.15, 0.2, 0.15, 0.15, 0.05, 0)$$

Тоді рішення приймаються з імовірностями (наближено):

$$p = \bar{p}H = (0.5414, 0.4586).$$

Це означає, що агент з імовірністю 0.5414 вибере першу альтернативу, а з імовірністю 0.4586 — другу.

#### 4. Багатофакторне прийняття рішень і дворівневий метод аналізу ієрархій

Вибір між альтернативами, як правило, здійснюється на основі тих чи тих факторів — критеріїв та/або ознак. Зокрема, на цьому ґрунтується класична дворівнева схема прийняття рішень у рамках методу аналізу ієрархій. Оцінки альтернатив за цим підходом обчислюють за формулою

$$u(a_i) = \sum_k \lambda_k \beta_{ik},$$

де  $\beta_{ik}$  — оцінка  $i$ -ї альтернативи, отримана на основі попарних порівнянь окремо за  $k$ -м критерієм;

$\lambda_k$  — вага  $k$ -го критерію, отримана на основі МПП між критеріями.

Вектор оцінок альтернатив на основі МПП найчастіше беруть як перронів (нормалізований головний власний вектор) МПП. Часто використовують також підхід, за якого як оцінки використовують середні геометричні рядків МПП. Популярним є спосіб отримання оцінок на основі логарифмічного методу найменших квадратів [4], особливо якщо МПП є неповною.

У [7] запропоновано підхід до формування більш структурованих моделей «стан–імовірність дії» на основі введення взаємопов'язаних систем станів. У загальних рисах, кожна система станів пов'язується з окремим фактором, задають матриці, що описують вплив одного фактора на інший. Застосуємо цей підхід до побудови дворівневої моделі «стан–імовірність дії» на основі попарних порівнянь альтернатив із метою моделювання багатофакторного недетермінованого прийняття рішень; це наблизить підхід, що розвивається, до методу аналізу ієрархій. Деякі міркування щодо цього наведено в [14], нижче дамо певну їх формалізацію.

#### 5. Дворівнева модель «стан–імовірність дії» на основі попарних порівнянь

В основі запропонованої моделі лежать такі міркування. Розглядається  $K$  критеріїв, за якими порівнюють альтернативи, а також система пов'язаних із ними правил на зразок «якщо альтернатива  $A$  краща за альтернативу  $B$  за окремо взятим критерієм, то  $A$  краща за  $B$  в цілому». Зрозуміло, що висновки, зроблені за одним правилом, можуть суперечити висновкам, зробленим за іншим; пропонується здійснювати комбінування висновків на основі тих чи тих вагових коефіцієнтів критеріїв. Опишемо ці міркування більш формалізовано.

Оскільки йдеться про порівняння двох альтернатив, то можна вводити стани, які відповідають мірам переваг однієї альтернативи над іншою в термінах градацій переваг подібно до того, як це зроблено в п. 3. Система станів  $S^* = \{s_i^*, i = \overline{1, m^*}\}$  відповідає загальній мірі переваги на основі сукупності критеріїв, а системи станів  $S^{(k)} = \{s_i^{(k)}, i = \overline{1, m^{(k)}}\}, k = \overline{1, K}$  — мірам переваг за окремим критерієм. Легко бачити, що між  $S^*$  та  $S^{(k)}, k = \overline{1, K}$  існує ієрархічний зв'язок.

Введемо набір випадкових величин  $\varphi_k, k = \overline{1, K}$ , які пов'язані з окремими критеріями та змістовно означають, в якому з відповідних станів перебуває агент. Введемо також матриці  $Y^{(k)}$ , які пов'язують систему станів  $S^*$  з системами станів  $S^{(k)}$ :

$$Y_{ij}^{(k)} = P(\eta = s_j^* | \varphi_k = s_i^{(k)}, k = \overline{1, K}, i = \overline{1, m^{(k)}}, j = \overline{1, m^*}).$$

Тоді вектор ймовірностей прийняття рішень можна обчислити за формулою

$$p = \overline{p^*} H, \quad (1)$$

$$\overline{p^*} = \sum_{k=1}^K \lambda_k p^{(k)}, \quad (2)$$

$$p^{(k)} = \overline{p^{(k)}} Y^{(k)}, \quad (3)$$

$p^{(k)}$  — задані ймовірності перебування агента в станах відповідної системи  $S^{(k)}$ ,

$\lambda_k$  — ваговий коефіцієнт, який задає міру важливості  $k$ -го критерію. Ці коефіцієнти можна отримати на основі заданої матриці попарних порівнянь між критеріями, як це робиться в класичному МАІ.

У принципі, можна взяти однакову кількість градацій переваг (тобто станів) для всіх систем станів, а всі матриці  $Y_{ij}^{(k)}$  — одиничними (що означає детермінований зв'язок між системами станів на різних рівнях). Але гнучкість моделі можна підвищити, якщо дозволити різні кількості градацій переваг, а також недетермінований зв'язок між системами станів. Саме так ми зробимо в наступному прикладі.

## 6. Ілюстративний приклад

Промодельємо процес президентських виборів у деякій гіпотетичній країні. Нехай за перемогу змагаються кандидати від двох партій — Партії Утопічних Ідеалів (ПУІ) та Партії Гіркої Правди (ППП).

Нехай порівняння здійснюються за трьома критеріями:

- $K1$  — харизматичність кандидатів;
- $K2$  — привабливість політичної програми;
- $K3$  — реалістичність політичної програми.

Імовірно, в реальності критерії можуть бути дещо іншими, але наш приклад має ілюстративний характер.

Систему станів, безпосередньо пов'язану з прийняттям рішення, тобто з волевиявленням виборця, та відповідну матрицю «стан–імовірність дії» візьмемо такі самі, як і в п. 3.

Із кожним критерієм пов'яжемо систему станів із параметром  $q = 2$ . Змістовно відповідні градації можна проінтерпретувати так:

0 — рівнозначність за даним критерієм;

1 — незначна перевага;

2 — помітна перевага.

Матриці  $Y^{(k)}$ , які матимуть розмір  $(m^{(k)} \times m^*)$ ,  $m^{(k)} = m^{(K1)} = m^{(K2)} = m^{(K3)} = 2q + 1$ , тобто в цьому випадку  $(5 \times 9)$ , також візьмемо однаковими, наприклад:

$$Y = Y^{(K1)} = Y^{(K2)} = Y^{(K3)}$$

$$= \begin{pmatrix} 0.9 & 0.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0.2 & 0.4 & 0.2 & 0.1 & 0 & 0 & 0 \\ 0.01 & 0.02 & 0.07 & 0.2 & 0.4 & 0.2 & 0.07 & 0.02 & 0.01 \\ 0 & 0 & 0 & 0.1 & 0.2 & 0.4 & 0.2 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0.9 \end{pmatrix}$$

Зазначимо, що вибрано також центросиметричну матрицю.

Тоді результат волевиявлення по суті визначається заданням векторів ймовірностей перебування агента у станах систем, пов'язаних з окремими критеріями, а також ваговими коефіцієнтами критеріїв.

Нехай вектори ймовірностей перебування у відповідних станах дорівнюють

$$\overline{p^{(K1)}} = (0., \quad 0.3, \quad 0.4, \quad 0.3, \quad 0.)$$

$$\overline{p^{(K2)}} = (0.7, \quad 0.2, \quad 0.1, \quad 0., \quad 0.)$$

$$\overline{p^{(K3)}} = (0., \quad 0., \quad 0.1, \quad 0.2, \quad 0.7)$$

Змістовно це означає, що за критерієм привабливості програми агент більш схильний віддати перевагу кандидатові від ПУІ, за критерієм реалістичності — кандидатові від ППП, а за ризматичністю обидва кандидати рівноцінні.

Тоді вектори ймовірностей перебування агента в станах системи  $S^*$ , обчислені на основі формули (3) за кожним критерієм окремо, дорівнюють

$$p^{(K1)} = (0.004, \quad 0.038, \quad 0.088, \quad 0.23, \quad 0.28, \quad 0.23, \quad 0.088, \quad 0.038, \quad 0.004)$$

$$p^{(K2)} = (0.631, \quad 0.092, \quad 0.047, \quad 0.1, \quad 0.08, \quad 0.04, \quad 0.007, \quad 0.002, \quad 0.001)$$

$$p^{(K3)} = (0.001, \quad 0.002, \quad 0.007, \quad 0.04, \quad 0.08, \quad 0.1, \quad 0.047, \quad 0.092, \quad 0.631)$$

Для отримання комбінованого вектора ймовірностей за формулою (2) потрібно знати вагові коефіцієнти  $\lambda_k$ , які відображають міри важливості критеріїв. Ці коефіцієнти можуть бути задані безпосередньо, але для МАІ більш характерним є їх отримання на основі матриці попарних порівнянь. Для задання МПП будемо використовувати транзитивні шкали, а самі МПП задаватимемо в термінах кількостей градацій, як це було описано в п. 2. Для визначеності візьмемо параметр  $\tau^{(k)} = 1.5$ . Розмір такої МПП буде  $3 \times 3$  (за кількістю критеріїв).

Спочатку розглянемо ситуацію, коли всі три критерії рівнозначні. Тоді МПП у логарифмічній формі складатиметься з одних нулів, а відповідна традиційна МПП — з одних одиниць. Зрозуміло, що перронів вектор такої матриці і, відповідно, вектор вагових коефіцієнтів дорівнює

$$\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$$

(тобто всі критерії мають однакові вагові коефіцієнти).

Комбінований вектор, обчислений за формулою (2), у цьому випадку дорівнює,

$$\bar{p}^* = (0.212, 0.044, 0.0473, 0.1233, 0.1467, 0.1233, 0.0473, 0.044, 0.212),$$

а вектор ймовірностей прийняття рішень, обчислений за формулою (1) —

$$p = \bar{p}^* H = (0.5, 0.5)$$

Це й означає рівновагу альтернатив: жодна альтернатива не має переваги над іншою; ймовірності того, що виборець проголосує за будь-яку з двох партій, однакові.

Змінимо МПП для критеріїв: нехай тепер критерій привабливості має перевагу над критерієм реалістичності на одну градацію. Тепер МПП у логарифмічній формі матиме вигляд

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix},$$

а традиційна МПП —

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1.5 \\ 1 & 0.667 & 1 \end{pmatrix}.$$

У результаті обчислень, аналогічних вищеведеним, отримуємо ймовірності прийняття рішень (наближено):

$$p \approx (0.5325, 0.4675).$$

Рівновагу альтернатив порушено. ПУІ має перевагу за індивідуального вибору і, як проілюстровано в [11], практично завжди перемагатиме за колективного вибору більшістю голосів, якщо кількість виборців є достатньо великою.

## 7. Висновки та обговорення

У статті здійснено розвиток підходу до моделювання процесів недетермінованого прийняття рішень на основі багаторівневої моделі «стан–імовірність дії». Ідеться перш за все про вибір між двома альтернативами, які можна порівнювати за довільною кількістю критеріїв. Описано, яким чином можна будувати матриці «стан–імовірність дії» на основі попарних порівнянь між самими альтернативами та між критеріями. При цьому використовуються транзитивні шкали, і можна задавати дискретні значення градацій переваг. У цьому контексті показано зв'язок підходу, що розвивається, з класичним дворівневим багатофакторним методом аналізу ієрархій. Наведено ілюстративний приклад.

Видається доцільним розвивати методики, спрямовані на зменшення ступеня довільності при формуванні матриць «стан–імовірність дії». З цією метою варто звернути увагу на підходи, пов'язані з застосуванням нечітких величин та описом на цій основі ступенів переваг, а також з урахуванням ступенів рішучості агентів при заданні ймовірностей прийняття рішень.

Звертає на себе увагу ще й така обставина. У нашому прикладі ми довільно змінювали міру переваги одного критерію над іншим незалежно від інших елементів матриці. Часто це призводить до зниження узгодженості МПП, і для подолання цього ефекту слід робити належні корективи, зокрема застосовувати процедури, спрямовані на отримання більш узгоджених матриць попарних порівнянь [8; 10; 16 та ін.]. При використанні цих процедур виникає низка проблем, і механічне їх застосування може призвести до небажаних результатів; це було проілюстровано, зокрема, в [15]. Зокрема, слід розрізняти «доброякісні» помилки та «злроякісні» маніпуляції з боку недостатньо добросовісних експертів і залежно від цього приймати різні рішення.



## Список літератури

- Beneduci R. Stochastic matrices and a property of the infinite sequences of linear functionals / R. Beneduci // *Linear Algebra and its Applications*. — 2010. — Vol. 433. — Pp. 1224–1239.
- Brunelli M. Introduction to the Analytic Hierarchy Process / M. Brunelli. — Springer, Cham, 2015.
- Choo E. A Common Framework for Deriving Preference Values from Pairwise Comparison Matrices / E. Choo, W. Wedley // *Comput. Oper. Res.* — 2004. — Vol. 31 (6). — Pp. 893–908.
- Crawford G. A note on the analysis of subjective judgment matrices / G. Crawford, C. Williams // *Journal of Mathematical Psychology*. — 1985. — Vol. 29 (4). — Pp. 387–405. [https://doi.org/10.1016/0022-2496\(85\)90002-1](https://doi.org/10.1016/0022-2496(85)90002-1).
- Ho W. Integrated analytic hierarchy process and its applications. A literature review / W. Ho // *European Journal of Operational Research*. — 2008. — Vol. 186 (1). — Pp. 211–228.
- Ishizaka A. Review of the main developments in the analytic hierarchy process / A. Ishizaka, A. Labib // *Expert Syst. Appl.* — 2011. — Vol. 38. — Pp. 14336–14345.
- Ivokhin E. V. Restructuring of the Model “State-Probability of Choice” Based on Products of Stochastic Rectangular Matrices / E. V. Ivokhin, O. V. Oletsky // *Cybern Syst Anal.* — 2022. — Vol. 58 (2). — Pp. 242–250. <https://doi.org/10.1007/s10559-022-00456-z>.
- Koczkodaj W. The limit of inconsistency reduction in pairwise comparisons / W. Koczkodaj, J. Szybowski // *Int. J. Appl. Math. Comput. Sci.* — 2016. — Vol. 26. — Pp. 721–729.
- Melman A. Symmetric centrosymmetric matrix-vector multiplication / A. Melman // *Linear Algebra Appl.* — 2000. — Vol. 320. — Pp. 193–198.
- Mikhailov L. Improving the Ordinal Consistency of Pairwise Comparison Matrices / L. Mikhailov, S. Siraj // *Proceedings of the XI International Symposium for the Analytic Hierarchy Process ISAHp-2011*.
- Oletsky O. Formalizing the Procedure for the Formation of a Dynamic Equilibrium of Alternatives in a Multi-Agent Environment in Decision-Making by Majority of Votes / O. Oletsky, E. Ivohin // *Cybern Syst Anal.* — 2021. — Vol. 57. — Pp. 47–56. <https://doi.org/10.1007/s10559-021-00328-y>.
- Oletsky O. A Model of Information Influences on the Base of Rectangular Stochastic Matrices in Chains of Reasoning with Possible Contradictions / O. Oletsky // *CEUR Workshop Proceedings*. — 2021. — Vol. 3179. — Pp. 354–361.
- Oletsky O. On Constructing Adjustable Procedures for Enhancing Consistency of Pairwise Comparisons on the Base of Linear Equations / O. Oletsky // *CEUR Workshop Proceedings*. — 2021. — Vol. 3106. — Pp. 177–185.
- Oletsky O. On Applying the Structured Model “State-Probability of Action” to Multi-Criteria Decision Making and Contradictory Reasoning / O. Oletsky, I. Peleshchak // *CEUR Workshop Proceedings*. — 2022. — Vol. 3312. — Pp. 189–199.
- Oletsky O. Some Ways of Counteracting Possible Manipulations Within the AHP on The Base of Weighted Linear Equations / O. Oletsky, D. Dosyn // *CEUR Workshop Proceedings*. — 2022. — Vol. 3347. — Pp. 185–194.
- Potomkin M. M. Improvement of Analytic Hierarchy Process based on the Refinement of the Procedures for the Formation of Pairwise Comparison Matrices / M. M. Potomkin, M. V. Nikolaienko, D. I. Grazion // *Cybern Syst Anal.* — 2020. — Vol. 56. — Pp. 603–610. <https://doi.org/10.1007/s10559-020-00277-y>.
- Saaty T. L. The Analytic Hierarchy Process / T. L. Saaty. — McGraw-Hill, New York, 1980.
- Tsyganok V. V. Usage of Scales with Different Number of Grades for Pair Comparisons in Decision Support Systems / V. V. Tsyganok, S. V. Kadenko, O. V. Andriichuk // *International Journal of the Analytic Hierarchy Process*. — 2016. — Vol. 8 (1). — Pp. 112–130.
- Vaidya O. S. Analytic hierarchy process: An overview of applications / O. S. Vaidya, S. Kumar // *European Journal of Operational Research*. — 2006. — Vol. 169 (1). — Pp. 1–29.
- Weaver J. R. Centrosymmetric (cross-symmetric) matrices, their basic properties, eigenvalues and eigenvectors / J. R. Weaver // *Amer. Math. Monthly*. — 1985. — Vol. 92. — Pp. 711–717.
- Yu O. Assessing and Improving Consistency of a Pairwise Comparison Matrix in the Analytic Hierarchy Process / O. Yu // *Portland International Conference on Management of Engineering and Technology*. — 2017. — Pp. 1–6.

## References

- Beneduci, R. (2010). Stochastic matrices and a property of the infinite sequences of linear functionals. *Linear Algebra and its Applications*, 433, 1224–1239.
- Brunelli, M. (2015). *Introduction to the Analytic Hierarchy Process*, Springer, Cham.
- Choo, E., & Wedley, W. (2004). A Common Framework for Deriving Preference Values from Pairwise Comparison Matrices. *Comput. Oper. Res.*, 31 (6), 893–908.
- Crawford, G., & Williams, C. (1985). A note on the analysis of subjective judgment matrices. *Journal of Mathematical Psychology*, 29 (4), 387–405. [https://doi.org/10.1016/0022-2496\(85\)90002-1](https://doi.org/10.1016/0022-2496(85)90002-1).
- Ho, W. (2008). Integrated analytic hierarchy process and its applications. A literature review. *European Journal of Operational Research*, 186 (1), 211–228.
- Ishizaka, A., & Labib, A. (2011). Review of the main developments in the analytic hierarchy process. *Expert Syst. Appl.*, 38, 14336–14345.
- Ivokhin, E., & Oletsky, O. (2022). Restructuring of the Model “State-Probability of Choice” Based on Products of Stochastic Rectangular Matrices. *Cybern Syst Anal.*, 58 (2), 242–250. <https://doi.org/10.1007/s10559-022-00456-z>.
- Koczkodaj, W., & Szybowski, J. (2016). The limit of inconsistency reduction in pairwise comparisons. *Int. J. Appl. Math. Comput. Sci.*, 26, 721–729.
- Melman, A. (2000). Symmetric centrosymmetric matrix-vector multiplication. *Linear Algebra Appl.*, 320, 193–198.
- Mikhailov, L., & Siraj, S. (2011). Improving the Ordinal Consistency of Pairwise Comparison Matrices. *Proceedings of the XI International Symposium for the Analytic Hierarchy Process ISAHp-2011*.
- Oletsky, O., & Ivohin, E. (2021). Formalizing the Procedure for the Formation of a Dynamic Equilibrium of Alternatives in a Multi-Agent Environment in Decision-Making by Majority of Votes. *Cybern Syst Anal.*, 57, 47–56. <https://doi.org/10.1007/s10559-021-00328-y>.
- Oletsky, O. (2021a). A Model of Information Influences on the Base of Rectangular Stochastic Matrices in Chains of Reasoning with Possible Contradictions. *CEUR Workshop Proceedings*, 3179, 354–361.
- Oletsky, O. (2021b). On Constructing Adjustable Procedures for Enhancing Consistency of Pairwise Comparisons on the Base of Linear Equations. *CEUR Workshop Proceedings*, 3106, 177–185.
- Oletsky, O., & Peleshchak, I. (2022). On Applying the Structured Model “State-Probability of Action” to Multi-Criteria Decision Making and Contradictory Reasoning. *CEUR Workshop Proceedings*, 3312, 189–199.
- Oletsky, O., & Dosyn, D. (2022). Some Ways of Counteracting Possible Manipulations Within the AHP on The Base of Weighted Linear Equations. *CEUR Workshop Proceedings*, 3347, 185–194.
- Potomkin, M., Nikolaienko, M., & Grazion, D. (2020). Improvement of Analytic Hierarchy Process based on the Refinement of

- the Procedures for the Formation of Pairwise Comparison Matrices. *Cybern Syst Anal*, 56, 603–610. <https://doi.org/10.1007/s10559-020-00277-y>.
- Saaty, T. L. (1980). *The Analytic Hierarchy Process*. McGraw-Hill, New York.
- Tsyganok, V., Kadenko, S., & Andriichuk, O. (2016). Usage of Scales with Different Number of Grades for Pair Comparisons in Decision Support Systems. *International Journal of the Analytic Hierarchy Process*, 8 (1), 112–130.
- Vaidya, O., & Kumar, S. (2006). Analytic hierarchy process: An overview of applications. *European Journal of Operational Research*, 169 (1), 1–29.
- Weaver, J. (1985). Centrosymmetric (cross-symmetric) matrices, their basic properties, eigenvalues and eigenvectors. *Amer. Math. Monthly*, 92, 711–717.
- Yu, O. (2017). Assessing and Improving Consistency of a Pairwise Comparison Matrix in the Analytic Hierarchy Process. *Portland International Conference on Management of Engineering and Technology*, 1–6.

*O. Oletsky, I. Franchuk, V. Humynskyi*

## ON AN APPROACH TO FORMING TWO-LEVEL MODEL “STATE-PROBABILITY OF ACTION” ON THE BASE OF PAIRWISE COMPARISONS ON THE AND ANALYTIC HIERARCHY PROCESS

*An approach to modeling non-deterministic and probabilistic decision making on the base of the model “state-probability of choice” is being developed in the paper. A way to forming a specific type of such a model on the base of pairwise comparisons and the Analytic Hierarchy Process is suggested.*

*The regarded case is that an agent is to choose one of two available alternatives, but this choice depends on different criteria. Some systems of states connected to the hierarchy are suggested. The first-level system is the basis one, states of which correspond to probabilities of choosing the two options. Each second-level system corresponds to a separate criterion which may affect the final choice.*

*For forming systems of states, applying pairwise comparisons is suggested. Each state corresponds to a certain grade of preference between alternatives – either in general or by separate criterion; transitive scales for quantifying preference values are used. Within the framework of the model “state-probability of action” that actually corresponds to the following rule of making decisions: “if an alternative has the preference over some other alternative with respect to a separate criteria then it has an overall preference over that alternative”.*

*Decisions made by separate criteria probably shall contradict to each other. The suggested way to getting a combined decision is based on pairwise comparisons among the criteria like the approach common to the two-level Analytic Hierarchy Process. Weighting coefficients reflecting degrees of importance for each criterion are being found as the Perronian, i.e. the normalized eigenvectors of the pairwise comparison matrix though there are some other approaches.*

*An illustrative example involving two alternatives and three criteria is provided. This example illustrates both a situation of equilibrium between alternatives and breaking it by means of changing the matrix of pairwise comparisons among criteria.*

*Some ways of developing the suggested approach as well as some arising problems are discussed. This includes but is not limited to possible using of fuzzy estimations, non-linear transformations of grading scales, different way of treating different types of inconsistencies in PCMs etc.*

**Keywords:** ranking of alternatives, model “state-probability of action”, equilibrium of alternatives, pairwise comparisons, Analytic Hierarchy Process.

*Матеріал надійшов 19.08.2023*



## ОПТИМІЗАЦІЙНІ ЗАДАЧІ ДЛЯ ОЦІНОК ВІДНОСНОЇ ВАЖЛИВОСТІ ЕЛЕМЕНТІВ У МЕТОДІ АНАЛІЗУ ІЄРАРХІЙ<sup>1</sup>

Розглянуто задачу знаходження та задачу коригування коефіцієнтів у матриці попарних порівнянь для елементів методу аналізу ієрархій, де критерієм є мінімізація індексу узгодженості коефіцієнтів (максимального власного числа обернено-симетричної матриці) за обмежень на відхилення від заданих пріоритетів елементів. Наведено розрахунки оптимальних коефіцієнтів в обох задачах для прикладу Т. Сааті, пов'язаного з дослідженням інтенсивності освітлення та закону оберненого квадрата відстані до джерела освітлення. Розглянуто приклад застосування другої оптимізаційної задачі для формування компонентів мультимедійного психомодельовального середовища для підвищення ефективності реабілітації пацієнтів.

**Ключові слова:** шкала Сааті, матриця попарних порівнянь, пріоритет елементів, обернено-симетрична матриця, число Фробеніуса, власний вектор.

### Вступ

Метод аналізу ієрархій [1], який запропонував американський учений Томас Сааті, є замкненою логічною конструкцією, яка за допомогою ієрархій, пріоритетів і простих математичних правил забезпечує аналіз складних проблем у всьому їх різноманітті. Метод аналізу ієрархій з успіхом використовують для розв'язання практичних задач, і він досить часто приводить до оптимальних відповідей для задач планування, прогнозування та інших [1]. Він також може бути застосований для системного аналізу використання компонентів мультимедійного психомодельовального середовища для підвищення ефективності реабілітації пацієнтів [2].

Метод аналізу ієрархій базується на двох складниках. Перший складник пов'язаний зі шкалою попарних порівнянь для представлення результатів оцінок у кількісному вигляді. При проведенні попарних порівнянь ставлять такі запитання: «Який з елементів є важливішим? Який є найімовірнішим? Який із них є найефективнішим?». Відносну важливість елементів оцінюють у балах від 1 до 9. Якщо за порівняння одного елемента з іншим було отримано одне із зазначених чисел (1–9), то за порівняння цього іншого з першим матимемо обернену величину.

Другим складником методу аналізу ієрархій є встановлення пріоритетів елементів за допомогою матриці їх попарних порівнянь, яка є обернено-симетричною, тобто для її будь-яких коефіцієнтів  $i$  та  $j$  виконується співвідношення  $a_{ji} = 1/a_{ij}$ . Якщо  $n \times n$ -матриця  $A$  — обернено-симетрична матриця попарних порівнянь, то для  $n$  елементів пріоритет визначають за допомогою  $\lambda_A$  — максимального власного числа матриці  $A$  та відповідного йому власного вектора  $x_A$ , який задовольняє умові

$$Ax_A = \lambda_A x_A, \quad \sum_{i=1}^n (x_A)_i = 1. \quad (1)$$

Для матриць, компоненти якої є невід'ємними, число  $\lambda_A$  називають числом Фробеніуса, а вектор  $x_A$  — вектором Фробеніуса. Число  $\lambda_A$  визначає величину  $(\lambda_A - n)/(n - 1)$ , яку називають індексом узгодженості (ІУ) матриці  $A$ . Вважають, що якщо ІУ не перевищує 0,10, то можна бути задоволеним ступенем узгодженості суджень. Значення компонент вектора  $x_A$  задають пріоритети відповідних їм елементів, які встановлюють на основі аналізу обернено-симетричної матриці  $A$ .

Нижче розглянемо дві оптимізаційні задачі для знаходження та коригування коефіцієнтів у матриці попарних порівнянь для елементів методу аналізу ієрархій, де критерієм є мінімізація індексу узгодженості коефіцієнтів при обмеженнях на відхилення від заданих пріоритетів елементів. Матеріал статті викладено в трьох

<sup>1</sup> Роботу підтримано грантом Національного фонду досліджень України № 2021.01/0136.

розділах. У розділі 1 описано приклад Т. Сааті з дослідженням інтенсивності освітлення та закону оберненого квадрата відстані до джерела освітлення, у розділі 2 — задачу розрахунку оптимальних коефіцієнтів матриці попарних порівнянь для прикладу Т. Сааті, а в розділі 3 — задачу приведення уже наявної матриці попарних порівнянь до матриці попарних порівнянь із кращим індексом узгодженості.

### 1. Експеримент Сааті щодо закону оберненого квадрата оптики

У книжці Т. Сааті [1, с. 39–40] наведено приклад застосування методу аналізу ієрархій для вивчення закону оберненого квадрата відстані для освітлення чотирьох ідентичних стільців, які розташовані на прямій від джерела світла на відстані 9, 15, 21 та 28 ярдів. Нормалізований вектор пріоритетів для обернених квадратів відстаней до чотирьох стільців  $(1/5)^2, (1/15)^2, (1/21)^2$  та  $(1/28)^2$  вважатимемо таким  $p_4 = (0,61; 0,22; 0,11; 0,06)^T$ . Компоненти вектора  $p_4$  отримані за допомогою округлення до двох цифр після коми компонент вектора пріоритетів  $\tilde{p}_4 = (0,607168; 0,218581; 0,111521; 0,062730)^T$ , що задані з точністю до шести цифр після коми.

Метою експерименту було: спробувати порівняти попарно відносну візуальну освітленість стільців, якщо дивитися на них від джерела світла, заповнити матрицю попарних порівнянь освітленості чотирьох стільців та отримати взаємозв'язок між візуальною освітленістю стільців і відстанню від них до джерела світла. Цей експеримент повторювали двічі з різними експертами, матриці попарних порівнянь яких наведено нижче (див. матриці  $A$  та  $B$ ). Експертами для побудови першої матриці (матриця  $A$ ) були маленькі діти Т. Сааті — 5 і 7 років, які представляли свої якісні судження. Експертом для побудови другої матриці (матриця  $B$ ) була дружина Т. Сааті, яка не була присутня, коли діти виносили судження.

Результати цих двох експериментів (матриці попарних порівнянь  $A$  і  $B$ , числа  $\lambda_A$  і  $\lambda_B$ , вектори  $x_A$  і  $x_B$ , індекси узгодженості матриць  $IY_A$  і  $IY_B$ ) наведені нижче у співвідношеннях (1) та (2).

$$A = \begin{pmatrix} 1 & 5 & 6 & 7 \\ 1/5 & 1 & 4 & 6 \\ 1/6 & 1/4 & 1 & 4 \\ 1/7 & 1/6 & 1/4 & 1 \end{pmatrix}, \lambda_A = 4,39, x_A = \begin{pmatrix} 0,61 \\ 0,24 \\ 0,10 \\ 0,05 \end{pmatrix}, \quad (1)$$

$$IY_A = 0,13,$$

$$B = \begin{pmatrix} 1 & 4 & 6 & 7 \\ 1/4 & 1 & 3 & 4 \\ 1/6 & 1/3 & 1 & 2 \\ 1/7 & 1/4 & 1/2 & 1 \end{pmatrix}, \lambda_B = 4,10, x_B = \begin{pmatrix} 0,62 \\ 0,22 \\ 0,10 \\ 0,06 \end{pmatrix}, \quad (2)$$

$$IY_B = 0,03.$$

Тут числа та вектори Фробеніуса, індекси узгодженості суджень взяті такими, якими вони наведені у книжці Сааті, хоча потрібно зробити застереження щодо першого експерименту, де є невелика неточність у визначенні вектора  $x_A$ , враховуючи, що його компоненти представлені з точністю до двох цифр після коми. В дійсності число Фробеніуса  $\lambda_A$  та відповідні йому компоненти вектора  $x_A$  приймають такі значення

$$\tilde{\lambda}_A = 4,3907, \tilde{x}_A = (0,618669, 0; 235323; 0,100934; 0,045074)^T. \quad (3)$$

У результаті округлення вектора  $\tilde{x}_A$  до двох цифр після коми отримуємо вектор  $\bar{x}_A = (0,62; 0,22; 0,11; 0,06)^T$ , сума компонент якого є рівною 1,01. Тому щоб забезпечити суму компонент вектора  $x_A$  рівною одиниці, його першу компоненту, рівну 0,62, Т. Сааті зменшив на величину 0,01. Число та вектор Фробеніуса для співвідношення (3) обчислені за допомогою octave-програми `frobenius` [3]. Її також будемо використовувати для подальших розрахунків, які наведено в розділах 2 та 3.

Якщо порівняти власні вектори для першого та другого випробування з вектором  $p_4$ , обчисленим із закону оберненого квадрата оптики, то можна зазначити, що в обох випадках матриці попарних порівнянь відобразили закон природи. Меншою мірою це стосується експерименту (1), де індекс узгодженості матриці  $IY_A = 0,13$  є більшим за 0,1, та більшою мірою експерименту (2), де індекс узгодженості  $IY_B = 0,03$  є меншим за 0,1.

### 2. Перша оптимізаційна задача

Розглянемо наступну оптимізаційну задачу. Нехай із точністю до двох знаків після коми задано вектор  $p_A$  — вектор пріоритетів елементів методу аналізу ієрархій. Потрібно знайти коефіцієнти матриці попарних порівнянь елементів, де не менші за 1 коефіцієнти є цілими числами в діапазоні від 1 до 9 (шкала Сааті), щоб мінімального значення досягав індекс узгодженості матриці (максимальне власне число обернено-симетричної матриці), а округлені з точністю до двох цифр після коми компоненти вектора пріоритетів матриці були або рівними компонентам вектора  $p_A$ , або такими, що допускають незначні

відхилення від компонент вектора  $p_A$ , наприклад,  $|x_A - p_A| \leq 0,02$  (тобто не більше ніж 2 %).

Результати розв'язання цієї задачі для прикладу Т. Сааті, де відносну важливість елементів оцінюють у балах від 1 до 9, а округлені до двох цифр після коми коефіцієнти вектора  $x_A$  збігаються з вектором пріоритетів  $p_A$ , відображено в табл. 1. Тут було використано метод повного перебору невідомих коефіцієнтів від 1 до 9, число  $\lambda_A$  та вектор  $x_A$  обчислювали за допомогою оставе-функції **frobenius** [3], а для округлення чисел після коми було використано стандартну функцію **round**.

Таблиця 1

Відносна важливість елементів у прикладі Т. Сааті, якщо  $|x_A - p_A| = 0$

Варіанти $A, \lambda_A, x_A$	#0	#1	#2	#3	#4
$A(1, 2)$	2,7727	3	3	4	4
$A(1, 3)$	5,5455	6	7	5	5
$A(1, 4)$	10,1667	9	7	8	7
$A(2, 3)$	2,0000	2	2	2	2
$A(2, 4)$	3,6667	4	4	5	5
$A(3, 4)$	1,8333	2	2	2	2
$\lambda_A$	4,0000	4,0104	4,0501	4,0735	4,0954
$x_A(1)$	0,61	0,613	0,609	0,614	0,605
$x_A(2)$	0,22	0,219	0,221	0,216	0,220
$x_A(3)$	0,11	0,109	0,107	0,112	0,114
$x_A(4)$	0,06	0,059	0,064	0,057	0,061
$IY_A$	0,00	0,003	0,017	0,024	0,024

У таблиці 1 для чотирьох найкращих за індексом узгодженості варіантів наведено наддіагональні коефіцієнти матриці попарних порівнянь  $A(1,2), A(1,2), \dots, A(3,4)$ , які задають відносні показники візуальної освітленості стільців 1, 2, 3 і 4, відповідні їм числа  $\lambda_A$  та вектори пріоритетів  $x_A$ . Для варіанта #0 представлено дійсні з точністю до чотирьох цифр після коми коефіцієнти повністю узгодженої матриці  $A$ , яка відповідає вектору пріоритетів  $p_A$ . Для всіх чотирьох варіантів компоненти векторів  $x_A$  наведено з точністю до трьох чисел після коми, що дає змогу оцінити, як відбувалось округлення компонент векторів до двох цифр після коми.

Із таблиці 1 видно, що найкращим за індексом узгодженості є варіант #1, де наддіагональні коефіцієнти матриці попарних порівнянь отримані округленням наддіагональних коефіцієнтів повністю узгодженої матриці для варіанта #0. Йому відповідають такі параметри

$$A^* = \begin{pmatrix} 1 & 3 & 6 & 9 \\ 1/3 & 1 & 2 & 4 \\ 1/6 & 1/2 & 1 & 2 \\ 1/9 & 1/4 & 1/2 & 1 \end{pmatrix}, \lambda_A^* = 4,01, x_A^* = \begin{pmatrix} 0,61 \\ 0,22 \\ 0,11 \\ 0,06 \end{pmatrix}, \quad (4)$$

$IY_A^* = 0,003,$

звідки видно, що індекс узгодженості матриці є значно меншим за 0,1. Наступними за критерієм після варіанта (4) йдуть варіанти #2, #3 і #4, які мають індекс узгодженості на порядок більший, ніж варіант #1. І хоча для всіх цих варіантів індекси узгодженості менше ніж 0,1, вони містять значні неточності щодо порівняння інтенсивності освітлення у випадках, коли об'єкт перебував дуже близько до джерела світла. Причина цього в тому, що на відносні показники візуальної освітленості сильно впливають неточності визначення абсолютних величин. Невелика помилка в оцінці відстаней до джерела світла дає значну помилку в результатах.

Кращі за індексом узгодженості варіанти матриць попарних порівнянь можна отримати, якщо дозволити компонентам вектора пріоритетів незначно відрізнятись від компонент вектора  $p_A$ . Про це свідчать представлені в табл. 2 результати розрахунків, де для всіх варіантів індекси узгодженості рівні 0,003.

Таблиця 2

Результати розрахунків для прикладу Т. Сааті, якщо  $|x_A - p_A| \leq 0,02$

Варіанти $A, \lambda_A, x_A$	#1	#2	#3	#4	#5
$A(1, 2)$	3	3	3	3	3
$A(1, 3)$	5	5	6	6	6
$A(1, 4)$	9	9	9	9	8
$A(2, 3)$	2	2	2	2	2
$A(2, 4)$	4	3	4	3	3
$A(3, 4)$	2	2	2	2	2
$\lambda_A$	4,0080	4,0080	4,0104	4,0104	4,0164
$x_A(1)$	0,60	0,60	0,61	0,62	0,61
$x_A(2)$	0,22	0,21	0,22	0,21	0,21
$x_A(3)$	0,12	0,12	0,11	0,11	0,11
$x_A(4)$	0,06	0,07	0,06	0,06	0,07
$IY$	0,003	0,003	0,003	0,003	0,003

У таблиці 2 наведено такі самі величини, як і в табл. 1. Тут варіанти #1, #2, #4 і #5 такі, що компоненти вектора пріоритетів для них відрізняються всього на 2 % від ідеальних, які відповідають варіанту #3, який збігається з ідеальним випадком (4). Зауважимо, що малі індекси узгоджень коефіцієнтів матриці забезпечують відсутність значних неточностей щодо порівняння інтенсивності освітлення у випадках, коли об'єкт був дуже близько до джерела світла.

### 3. Друга оптимізаційна задача

Розглянемо оптимізаційну задачу подібного типу, але для корегування коефіцієнтів уже заданої матриці попарних порівнянь елементів. Тут

критерієм буде мінімізація індексу узгодженості коефіцієнтів матриці (максимального власного числа обернено-симетричної матриці) за двох обмежень. Перше обмеження задає максимальне відхилення від заданих пріоритетів елементів, так само, як у першій задачі, а друге обмеження задає границю зверху на сумарне відхилення коефіцієнтів матриці попарних порівнянь від заданих. Результати розв'язання цієї задачі для корегування коефіцієнтів матриці  $A$ , яку задано згідно з формулою (1), за двох значень максимальної границі наведено в табл. 3.

Таблиця 3

Скороговані коефіцієнти до матриці  $A$  згідно з (1) у прикладі Т. Сааті

Варіанти	#0	#1a	#2a	#1b	#2b
$A, \lambda_A, x_A$					
$A(1, 2)$	5	4	4	4	4
$A(1, 3)$	6	6	6	6	6
$A(1, 4)$	7	7	7	7	7
$A(2, 3)$	4	2	3	2	3
$A(2, 4)$	6	6	5	5	4
$A(3, 4)$	4	3	3	2	2
$\lambda_A$	4,3907	4,1597	4,1793	4,0970	4,1022
$x_A(1)$	0,62	0,61	0,61	0,62	0,62
$x_A(2)$	0,24	0,22	0,23	0,22	0,22
$x_A(3)$	0,10	0,12	0,11	0,11	0,10
$x_A(4)$	0,05	0,05	0,05	0,06	0,06
IY	0,13	0,053	0,060	0,032	0,034

У таблиці 3 наведено по два найкращі за індексом узгодженості матриці варіанти — варіанти #1a і #2a для максимальної границі, рівної 4, та варіанти #1b і #2b для максимальної границі, рівної 6. Тут представлено наддіагональні коефіцієнти матриці попарних порівнянь  $A(1,2)$ ,  $A(1,2), \dots, A(3,4)$ , які задають відносні показники візуальної освітленості стільців 1, 2, 3 і 4, відповідні їм числа  $\lambda_A$  та вектори пріоритетів  $x_A$ . Для всіх чотирьох розрахованих варіантів компоненти векторів  $x_A$  наведено з точністю до двох цифр після коми. Як варіант #0 тут розглянуто матрицю  $A$ , яка відповідає формулі (1), з округленим до двох цифр після коми вектором пріоритетів.

Із таблиці 1 видно, що для всіх чотирьох варіантів індекси узгодженості є меншими за 0,1 та компоненти вектора пріоритетів відрізняються всього на 2 % від ідеальних, які відповідають варіанту #3 із табл. 2. При цьому є невеликі неточності щодо порівняння інтенсивності освітлення у випадках, коли стілець був дуже близько до джерела світла. Зауважимо, що варіант #2b збігається з результатом другого експерименту в прикладі Т. Сааті, заданого згідно з формулою (2). Тому можна вважати, що в результаті

розв'язання другої оптимізаційної задачі отримано в прикладі Сааті перехід від гіршого за індексом узгодженості першого експерименту (варіант #0) до кращого за індексом узгодженості другого експерименту (варіант #2b).

За аналогією з такими самими параметрами, як у табл. 3, за допомогою другої оптимізаційної задачі проведено розрахунки для корегування матриці попарних порівнянь, яку побудовано лікарем фізичної реабілітаційної медицини для оцінки важливості чотирьох компонентів (відео, зображення, зони Захар'їна — Геда, концепція У-Сін) мультимедійного середовища за включення їх у реабілітаційну програму [2]. Матриці попарних порівнянь, яка корегувалася, відповідають такі параметри

$$A = \begin{pmatrix} 1 & 4 & 3 & 2 \\ 1/4 & 1 & 3 & 2 \\ 1/3 & 1/3 & 1 & 4 \\ 1/2 & 1/2 & 1/4 & 1 \end{pmatrix}, \lambda_A = 4,68, x_A = \begin{pmatrix} 0,47 \\ 0,24 \\ 0,18 \\ 0,11 \end{pmatrix}, \quad (5)$$

$$IY_A = 0,23,$$

які наведено як варіант #0 в табл. 4. В табл. 4 наведено результати розв'язання другої оптимізаційної задачі — варіанти #1a і #2a для максимальної границі, рівної 4, та варіанти #1b і #2b для максимальної границі, рівної 6. Із таблиці видно, що для того щоб забезпечити індекс узгодженості матриці меншим за 0,1, потрібно змінити вихідну вагомість компонентів мультимедійного середовища при його розробці, через корегування коефіцієнтів матриці (5).

Таблиця 4

Скороговані коефіцієнти матриці  $A$  згідно з (5) для лікаря фізичної реабілітаційної медицини

Варіанти	#0	#1a	#2a	#1b	#2b
$A, \lambda_A, x_A$					
$A(1, 2)$	4	2	3	2	2
$A(1, 3)$	3	3	2	4	4
$A(1, 4)$	2	4	3	3	3
$A(2, 3)$	3	2	2	2	2
$A(2, 4)$	2	2	2	2	2
$A(3, 4)$	4	2	3	3	4
$\lambda_A$	4,6780	4,1241	4,1431	4,2072	4,3145
$x_A(1)$	0,47	0,46	0,46	0,47	0,047
$x_A(2)$	0,24	0,25	0,24	0,25	0,24
$x_A(3)$	0,18	0,18	0,19	0,17	0,19
$x_A(4)$	0,11	0,10	0,12	0,11	0,10
IY	0,226	0,041	0,048	0,069	0,105

Розрахунки для таблиць 3 і 4 було отримано за методом повного перебору невідомих коефіцієнтів від 1 до 9, та відсіву тих варіантів, які не задовольняють заданим обмеженням. Зауважи-

мо, що для розв'язання другої оптимізаційної задачі можна використати метод вектора спаду [5], який має бути значно ефективнішим, ніж метод повного перебору. При цьому околom початкової точки (стартова матриця попарних порівнянь) можна визначити множину таких коефіцієнтів матриці, відстань від яких до початкової матриці відрізняється не більше ніж на границю зверху на сумарне відхилення коефіцієнтів матриці попарних порівнянь від заданих.

### Висновки

У статті запропоновано дві оптимізаційні задачі для аналізу коефіцієнтів у матриці попарних порівнянь елементів для методу аналізу ієрархій, де критерієм є мінімізація індексу узгодженості матриці. Перша задача пов'язана зі знаходженням цілих коефіцієнтів у шкалі Сааті за об-

межень на незначні відхилення округлених до двох цифр після коми компонент вектора пріоритетів елементів від заданих. Друга задача пов'язана з корегуванням заданих коефіцієнтів матриці попарних порівнянь за границі зверху на сумарне відхилення коефіцієнтів матриці від заданих та умові на відхилення округлених компонентів вектора пріоритетів елементів від заданих компонент.

Наведено розрахунки оптимальних коефіцієнтів в обох оптимізаційних задачах для прикладу Сааті, пов'язаного з дослідженням інтенсивності освітлення та закону оберненого квадрата відстані до джерела освітлення. Розглянуто приклад застосування другої оптимізаційної задачі для формування вагомості компонентів мультимедійного психомодельовального середовища для підвищення ефективності реабілітації пацієнтів.

### Список літератури

1. Саати Т. Принятие решений. Метод анализа иерархий / Т. Саати. — Москва : Радио и связь, 1993. — 278 с.
2. Вакуленко Д. В. Системно-аналітичне обґрунтування застосування мультимедійного середовища для профілактики та реабілітації різних захворювань / Д. В. Вакуленко // Медична інформатика та інженерія. — 2015. — № 1. — С. 89–96.
3. Стецюк П. І. О спектральных свойствах матриц Леонтьева / П. И. Стецюк // Статистика. Моделирование.

4. Сергиенко И. В. Математические модели и методы решения задач дискретной оптимизации / И. В. Сергиенко. — Киев : Наукова думка, 1988. — 471 с.

### References

- Saati, T. (1993). *Prinyatie reshenii. Metod analiza ierarkhii*. Radio i svyaz [in Russian].
- Sergienko, I. V. (1988). *Matematicheskie modeli i metodi resheniya zadach diskretnoi optimizatsii*. Naukova dumka [in Russian].
- Stetsyuk, P. I. (2011). O spektralnykh svoystvakh matrits Leonteva. In *Statistika. Modelirovanie. Optimizatsiya: sbornik trudov Vserossiiskoi konferentsii (Chelyabinsk, 28 noyabrya —*

- 3 dekabrya 2011 g.)* (pp. 173–178). Izdatelskii tsentr YuUrGU [in Russian].
- Vakulenko, D. V. (2015). Systemno-analitychne obgruntuvannia zastosuvannia multymediinoho seredovyscha dlia profilaktyky ta reabilitatsii riznykh zakhvoriuvan. *Medychna informatyka ta inzheneriia*, 2, 89–96 [in Ukrainian].

P. Stetsyuk, D. Vakulenko, V. Lyashko

## OPTIMIZATION PROBLEMS FOR ASSESSING THE RELATIVE IMPORTANCE OF ELEMENTS IN THE ANALYTIC HIERARCHY PROCESS

*Analytic hierarchy process (AHP) was proposed by American scientist Thomas Saati. It is a closed logical structure that, with the help of hierarchies, priorities and simple mathematical rules, provides analysis of complex problems in all their diversity. Analytic hierarchy process is successfully used for solving practical problems and quite often leads to optimal answers for planning, forecasting and other problems. It can also be applied to system analysis of the using multimedia factors for increasing the effectiveness of patients' rehabilitation at various stages of disease.*

*The article proposes two optimization problems for coefficients analysis in the matrix of pairwise comparisons of elements for analytic hierarchy process, with criterion is minimization of the matrix consistency index. The first problem is related to finding integer coefficients in the scale of T. Saati subject to insignificance of components deviations of the vector of priorities of elements rounded to two digits from the given*

ones. The second problem is related to correction of the specified pairwise comparisons matrix coefficients subject to upper bound on total deviation of the matrix coefficients from the specified ones and subject to deviation of the rounded components of the element priorities vector from the specified components.

Calculations of the optimal coefficients in both optimization problems for the example of T. Saati related to study of lighting intensity and the law of the inverse square of the distance to the light source are given. An example of the second optimization problem application for correcting matrix of pairwise comparisons, which was built by a rehabilitation doctor to assess the importance of four elements of the multimedia environment when including them in the rehabilitation program, is considered.

The material of the article is presented in three sections. Section 1 describes T. Saati's example from the study of light intensity and the law of the inverse square of the distance to the light source. Section 2 describes the first optimization problem and provides calculations of the optimal coefficients for the pairwise comparisons matrix for the example of T. Saati. Section 3 describes the second optimization problem and gives examples of its application to bring the already existing pairwise comparisons matrix to the pairwise comparisons matrix with better consistency index both for the experiment of T. Saati and for the analysis of using of multimedia factors for improving efficiency of patients' rehabilitation.

**Keywords:** Saati's scale, matrix of pairwise comparisons, priority of elements, inverse-symmetric matrix, Frobenius number, eigenvector.

*Матеріал надійшов 15.07.2023*



Creative Commons Attribution 4.0 International License (CC BY 4.0)



Жежерун О. П., Смиш О. Р., Пруднікова А. О.

## ПІДХОДИ ДО ПОБУДОВИ ВИВОДУ В ОНТОЛОГІЧНІЙ БАЗІ ЗНАНЬ

У статті наведено два приклади реалізації виводу в онтологічній базі знань. Один — із використанням SWRL-правил, другий — як систему з елементами обробки природної мови. Наведено опис створених фрагментів таксономічної ієрархії для предметної галузі (планіметрії). Як приклади використано прості задачі відкритого типу зі шкільного підручника з геометрії. Продемонстровані процедури є частиною рекомендаційної навчальної системи, яку розробляють на факультеті інформатики Національного університету «Києво-Могилянська академія».

**Ключові слова:** обробка природної мови, рекомендаційна система, геометрія, база знань, онтологія, Protégé, OWL, Reasoner, SWRL.

### Вступ

Сьогодні дедалі гостріше формується попит на системи, які полегшують процес навчання, автоматизують його та уможливають самостійність учня. Інтелектуальні навчальні системи — це один із перспективних напрямів, і створення таких систем сприяє якіснішому навчанню. Ця робота стосується розвитку навчальної рекомендаційної системи, яку реалізують на факультеті інформатики Національного університету «Києво-Могилянська академія». Система орієнтована на допомогу у вивченні геометрії учням середньої школи та має на меті:

- навчити правильно аналізувати умову задачі, будувати послідовність міркувань, які забезпечують розв'язок задачі, правильно будувати графічний малюнок, коректно документувати процес розв'язку;
- сприймати текст задачі, поданий природною українською мовою (зі шкільного підручника), розв'язувати задачу автоматизовано, будувати графічний малюнок, формувати протокол розв'язку природною мовою в спосіб, зрозумілий учневі;
- здійснювати загальну консультативну підтримку процесу навчання, забезпечуючи учня необхідним теоретичним матеріалом.

Зазвичай формування системи відбувається покроково, тому зараз розробки зосереджені на матеріалі 7 класу, але надалі заплановано дійти до рівня допомоги учневі під час підготовки до ЗНО/НМТ.

Необхідно пояснити, що ідеологічно система бере свій початок від ідей В. Глушкова, запропонованих у концепції «Алгоритму Очевидності»

[2]. Розвиваючи в Києві в Інституті кібернетики систему автоматичного доведення теорем у математиці, Віктор Михайлович робив акцент не на створенні «ефективних універсальних процедур» доведення теорем, а на розвитку програмних засобів, які співпрацюють із математиком у процесі доведення теореми, забезпечують інформаційну підтримку й можуть бути орієнтовані на програмування рішень навіть «для однієї важкої теореми». Відповідно до планів В. Глушкова [1], у пам'ять комп'ютера заноситься певний фрагмент математичної теорії, наприклад основи теорії груп. Математичні тексти представляються спеціальною мовою для запису таких текстів. Твердження, яке математик вводить у систему, ставало «очевидним», якщо машинний алгоритм встановлював, що воно є логічним наслідком накопиченого в системі матеріалу. Якщо система доводила певне твердження, то вона видавала протокол доведення у форматі, який відповідав науковій публікації.

Проводячи такі аналогії, варто сказати, що теперішня мета — це зробити очевидним для учня процес розв'язку задачі, використовуючи різноманітні процедури, забезпечити інформаційну підтримку. У цій статті є розглянуто дві різні процедури, які використовуються в системі: одна базується на використанні логічних правил, а інша поєднує аналіз української природної мови з онтологічною базою знань.

### Онтологічні бази знань

Онтологія — це концептуалізація певної галузі знань і значущий напрям у галузі інтелектуаль-

них систем. Концептуалізацією називають процес створення та встановлення зв'язків між основними поняттями, які виділено на основі класифікації базових термінів у певній предметній галузі. Онтології уможливають представлення знань про поняття та відношення між ними.

Онтологія формується з таких елементів:

1. Поняття: терміни або поняття, що використовують у певній царині, які можуть мати ієрархічну структуру та впорядковані за принципом «батько–дитина».
2. Відносини: зв'язки між поняттями. Вони можуть бути бінарними (між двома поняттями) або багатозначними. Наприклад: «має атрибут», «є частиною» тощо.
3. Властивості: використовують для опису та характеристики термінів.
4. Аксиоми: певні правила або обмеження, що встановлюють на відносини, властивості або взаємодію між термінами.
5. Логічні правила: правила, які використовують для формалізації логічних відносин і виведення нових знань із наявних.

Онтологію можна представити в графічному вигляді або описати формальною мовою. Для формального опису онтології використано мову OWL (Ontology Web Language). Це мова семантичного вебу, яка призначена для представлення складних знань про предметну галузь та її зв'язків. Знання, виражені в OWL, можна використовувати в комп'ютерних програмах, наприклад, щоб переконатись в узгодженості цих знань [4].

Для створення бази знань спершу потрібно конкретизувати задачу та створити детальний опис кроків. Цей алгоритм допоможе створити структуру бази знань і визначити необхідні елементи, властивості та залежності між ними для моделювання геометричних задач. Здійснено такі етапи:

1. Проаналізовано та відібрано клас задач, що використовуватимуться в цій базі знань. Для початку обрано порівняно прості задачі, як-от задачі на знаходження кутів у трикутнику, задачі на знаходження площі та периметра.
2. Створено концептуальну модель, тобто визначено основні класи, які будуть залучені для розв'язування задач. Визначено властивості для цих класів.
3. Встановлено ієрархію класів, створено підкласи, визначено обмеження та певні правила для властивостей.
4. Додано екземпляри (індивіди). Встановлено зв'язки між індивідами.
5. Зроблено перевірку та тестування. Це значущий крок, адже на ньому відбувається перевірка правильності моделі через тестування різних шляхів.
6. Визначено запити, що можуть бути виконані до бази знань (наприклад, запит на отримання всіх кутів конкретного трикутника та їхньої градусної міри). Реалізовано логіку для виконання цих запитів.
7. Використано створену базу знань для геометричних задач, для надання необхідної інформації. Також застосовано систему для отримання нової інформації на основі встановлених властивостей та правил.

Першим кроком для створення онтології є формування ієрархії класів. Необхідно визначити основні класи, що використовуватимуться в базі знань.

Надкласом є «Пряма», підкласами якої є «Відрізок» і «Промінь». «Геометрична фігура», що належить до класу «Відрізок», має один підклас.

Свою чергою, «Трикутник» має такі шість підкласів: «Гострокутний трикутник», «Прямокутний трикутник», «Рівнобедрений трикутник», «Рівносторонній трикутник», «Тупокутний трикутник», «Різносторонній трикутник».

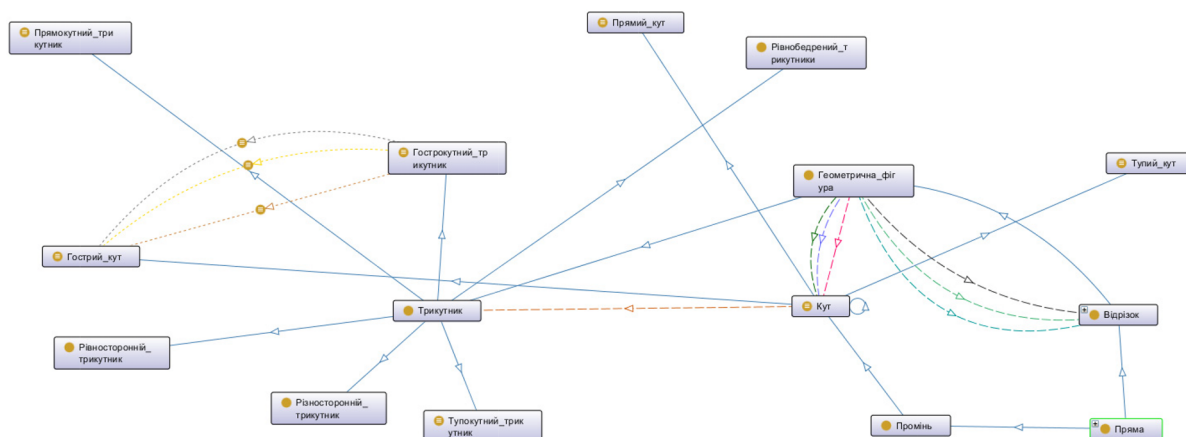


Рис. 1. Ієрархія класів і їхніх зв'язків

До класу «Промінь» належить підклас «Кут», який має три підкласи: «Гострий\_кут», «Прямий\_кут», «Тупий\_кут».

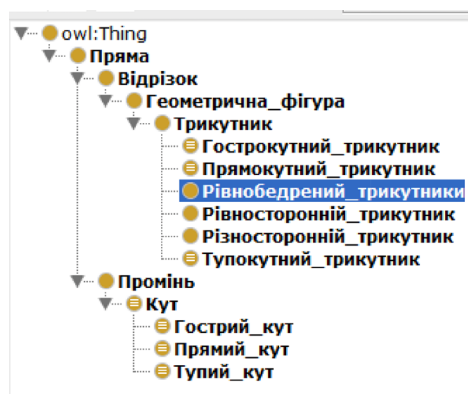


Рис. 2. Ієрархія класів

На другому кроці варто визначити та додати властивості для кожного класу, щоб описати їхні характеристики. У Protégé властивості бувають двох типів: «Object Properties» і «Data Properties». До першого типу належать властивості, які використовують для встановлення взаємозв'язків між класами.

Після створення властивостей треба встановити обмеження та визначення для властивостей і класів. Прикладом може бути визначення для класу «Тупий\_кут».

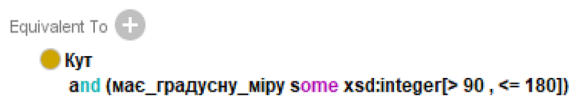


Рис. 3. Клас «Тупий\_кут»

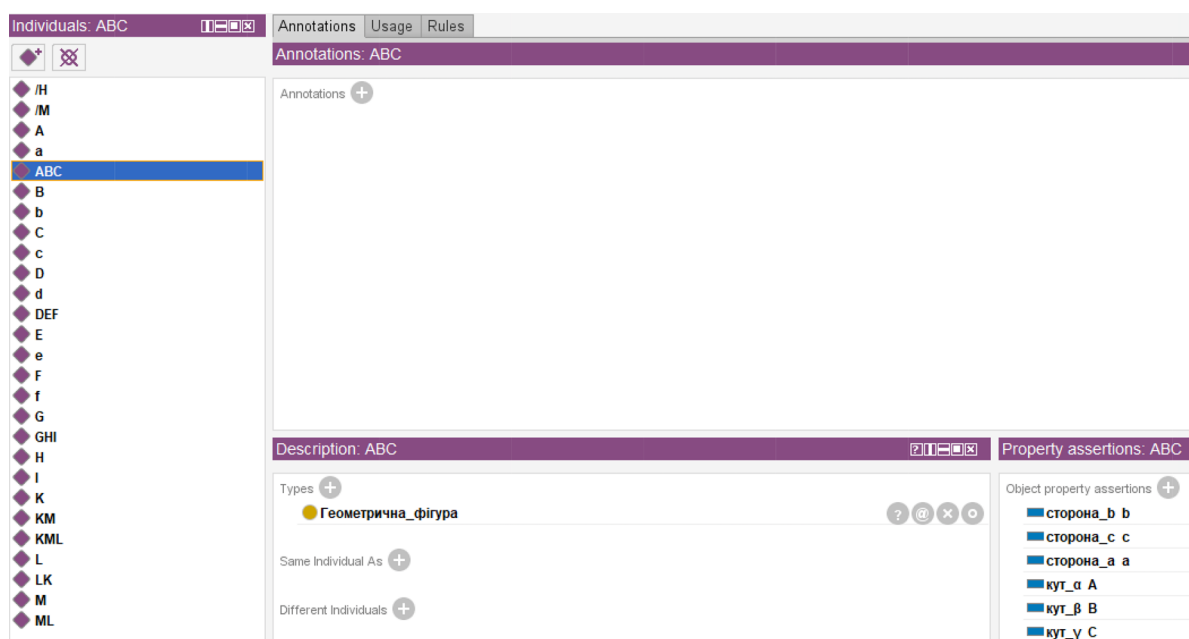


Рис. 4. Індивіди

Щоб зазначити обмеження, що в кута може бути тільки одне значення градусної міри, потрібно обрати Functional для властивості «має\_градусну\_міру».

Для реалізації наступного кроку треба створити індивіди.

Для встановлення властивості «сторона\_a» потрібно створити відповідний відрізок, аналогічно зробити й для кутів. Тобто, щоб створити повноцінний трикутник, необхідно додати до індивідів три відрізки та три кути.

У Protégé існують вбудовані інструменти для валідації. «OWL DL reasoner» — це механізм для перевірки бази знань на відповідність стандарту OWL. Серед них є Pellet, HermiT та FaCT++ [5].

SWRL (Semantic Web Rule Language) [7] є мовою правил, яку використовують для виконання логічних виводів за допомогою наявних знань, що зберігаються в онтології. Правило визначається двома основними елементами: антецедентом (тілом) і консеквентом (головою). Тіло визначає умови, що мають виконатись, а голова відповідає за дії, що буде зроблено, якщо умова виконається.

### Приклади розв'язування задач у Protégé системі

**Приклад задачі № 1.** Для розв'язання задачі на знаходження периметра правило матиме такий вигляд:

**Rule:** Трикутник(?t), сторона\_a(?t, ?a), сторона\_b(?t, ?b), сторона\_c(?t, ?c), має\_довжину(?a, ?l1), має\_довжину(?b, ?l2), має\_довжину(?c, ?l3), add(?res, ?l1, ?l2, ?l3) -> периметр(?t, ?res)

**Рис. 5.** Правило для знаходження периметра

У цьому SWRL правилі визначено, що якщо об'єкт «?t» є трикутником і має сторони «?a», «?b», «?c», що своєю чергою мають відповідні довжини: «?l1», «?l2», «?l3», то тоді цей трикутник має периметр «?res», який обраховано за допомогою вбудованої математичної функції «add».

SPARQL є мовою запитів, яка уможлиблює виконання пошуку інформації в RDF-графах. Її використовують для створення та формулювання складних запитів, щоб отримати дані. У SPARQL, як і в SQL, запити побудовані з певних ключових слів, як-от SELECT, WHERE, FILTER, ORDER BY. Їх застосовують для визначення шаблонів запитів та умов фільтрації вихідних даних.

У Protege є декілька табів правил SPARQL: «SPARQL Query» та «SNAP SPARQL Query». Перше використовують, якщо потрібно отримати результати, що наявні в базі знань, а SNAP тоді, коли потрібно досягти висновків, що зробив Reasoner. Важливо пам'ятати, що при використанні цього табу Reasoner має бути активним. У цій роботі використано SNAP, адже важливим є саме результат виконання Reasoner.

Наприклад:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://www.semanticweb.org/nasty/ontologies/new.owl#>

SELECT ?s ?o
WHERE { ?s :периметр ?o }
```

**Рис. 6.** Приклад «SNAP SPARQL Query»

Результатом виконання цього запиту буде перелік трикутників та їхній периметр:

	?s	?o
:ABC		12
:DEF		26
:KML		18

**Рис. 7.** Результат виконання для прикладу

**Приклад задачі № 2.** Текст задачі: «Периметр рівнобедреного трикутника = 40 см, а основа = 10 см. Знайдіть бічні сторони трикутника.».

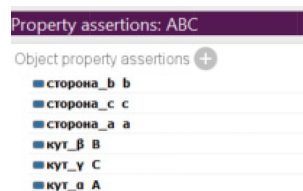
Для розв'язання задачі на знаходження бічних сторін трикутника знадобиться SWRL-правило:

```
Трикутник(?t) ^ має_основу(?t, ?o) ^ має_периметр(?t, ?p) ^
swrlb:subtract(?p, ?o, ?) ^ swrlb:divide(? , 2, ?res) ->
результат(?res)
```

**Рис. 8.** SWRL-правило

Це правило виходить із того, що бічну сторону трикутника може бути знайдено через різницю між периметром та основою, поділену удвічі.

Наступним кроком є створення відповідних індивідів. Додано та створено трикутник ABC. Також додано його сторони та кути. У результаті отримано такі взаємозв'язки:



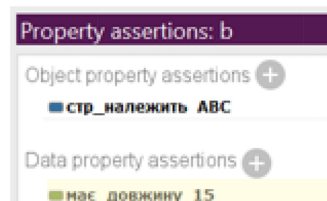
**Рис. 9.** Додавання трикутника

З умови задачі відомо, що периметр 40, а основа 10:



**Рис. 10.** Встановлення відомих даних

Після запуску Reasoner, виведено такі висновки:



**Рис. 11.** Вивід відповіді

Для сторони b Reasoner визначив, що довжина дорівнюватиме 15, таку саму довжину матиме й сторона c.

Отже, у такому разі, після введення інформації про трикутник, його основу та периметр, можна використати SWRL-правило для автоматизованого знаходження бічних сторін трикутника на основі заданих умов.

```
# text = Периметр рівнобедреного трикутника дорівнює 40 см, а основа дорівнює 10 см.
1 Периметр периметр NOUN Ncmsgn Animacy=Inan|Case=Nom|Gender=Masc|Number=Sing 4 nsubj _ _
2 рівнобедреного рівнобедрений ADJ Ap-msgf-ep Aspect=Perf|Case=Gen|Gender=Masc|Number=Sing|VerbForm=Part|Voice=Pass 3 amod _ _
3 трикутника трикутник NOUN Ncmsgn Animacy=Inan|Case=Gen|Gender=Masc|Number=Sing 1 nmod _ _
4 дорівнює дорівнювати VERB Vmpip3s Aspect=Imp|Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin 0 root _ _
5 40 40 NUM MlC-a Case=Acc|NumType=Card|Uninflct=Yes 6 nummod:gov _ _
6 см см NOUN Y Abbr=Yes|Animacy=Inan|Case=Gen|Gender=Masc|Number=Plur|Uninflct=Yes 4 obj _ SpaceAfter=No
7 , , PUNCT U _ 10 punct _ _
8 а а CCONJ Ccs _ 10 cc _ _
9 основа основа NOUN Ncfsnn Animacy=Inan|Case=Nom|Gender=Fem|Number=Sing 10 nsubj _ _
10 дорівнює дорівнювати VERB Vmpip3s Aspect=Imp|Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin 4 conj _ _
11 10 10 NUM MlC-a Case=Acc|NumType=Card|Uninflct=Yes 12 nummod:gov _ _
12 см см NOUN Y Abbr=Yes|Animacy=Inan|Case=Gen|Gender=Masc|Number=Plur|Uninflct=Yes 10 obj _ SpaceAfter=No
13 . . PUNCT U _ 4 punct _ _

# sent_id = 2
# text = Знайдіть бічні сторони трикутника.
1 Знайдіть знайти VERB Vmem-2p Aspect=Perf|Mood=Imp|Number=Plur|Person=2|VerbForm=Fin 0 root _ _
2 бічні бічний ADJ Ao--pasn Animacy=Inan|Case=Acc|Number=Plur 3 amod _ _
3 сторони сторона NOUN Ncfsnn Animacy=Inan|Case=Acc|Gender=Fem|Number=Plur 1 obj _ _
4 трикутника трикутник NOUN Ncmsgn Animacy=Inan|Case=Gen|Gender=Masc|Number=Sing 3 nmod _ SpaceAfter=No
5 . . PUNCT U _ 1 punct _ SpaceAfter=No
```

Рис. 12. Результат виконання UDPipe 2.12 моделі

### NLP система

Далі розглянуто роботу інтелектуальної системи зі сформованою геометричною онтологією та елементами обробки природної української мови. Сирий текст задачі, що надходить до системи: «Периметр рівнобедреного трикутника = 40 см, а основа = 10 см. Знайдіть бічні сторони трикутника.»

Для виведення розв’язку цієї задачі програма проходить декілька етапів: попереднє оброблення сирого тексту, де спеціальні символи, що властиві математичним задачам, замінюються на буквену форму (наприклад, знак « $\Leftrightarrow$ » — на «дорівнює»); оброблення тексту за допомогою UDPipe 2.12 моделі [6], що розбиває текст на токени, далі формує лемми слів, за допомогою лематизації, визначає морфологічні риси кожного токена, проставляє залежності між словами, за допомогою додатково встановлених частин мови (цей етап є значущим, оскільки мінімізує словоформи та створює зв’язки між токенами); додаткова перевірка лем за допомогою словника ВЕСУМ [3], для зменшення похибки моделі, що використано.

Після етапів, що присвячено попередньому обробленню сирого тексту, відбувається етап,

який намагається вичленити з підготовленого тексту всі можливі дані, які прописано в програмі, зокрема в геометричній онтології.

У такий спосіб програма виокремить ключові токени, як-от «трикутник», «периметр», «основа», «рівнобедрений», а також числа, які можна пов’язати зі словами-токенами.

Треба також зазначити, що в програмі сформовано додатковий модуль, який відповідає за те, щоб визначити, що є шуканим у задачі. У випадку задачі, яку розглянуто, програма поверне «['plur', 'side', 'side', 'polygon']». Це позначення означає, що програмі потрібно повернути бічні сторони фігури, у множині. Для формування такого позначення використано створений алгоритм, на основі слів, що вжито у тексті.

Після етапів опрацювання тексту починається етап застосування сформованої геометричної онтології. На базі цієї онтології до програми спершу заносяться всі вичленені дані з підготовленого тексту, із зіставленням фігури, їхніх вказаних елементів і вказаних величин цих елементів. Наприклад, у випадку з задачею, що використано як приклад, програма формує екземпляр класу «Трикутник», де вказується атрибут «Рівнобедрений», як різновид фігури. Також значен-

```
if self.perimeter > 0 and hasattr(self, 'sort') and self.sort == "рівнобедрений" and 1 <= known_s < 3:
    base = self.sides[1]
    side1 = self.sides[0]
    side2 = self.sides[2]
    if base.length == 0:
        base.length = self.perimeter - 2 * side1.length
    else:
        length = (self.perimeter - base.length) / 2
        side1.length = length
        side2.length = length
```

Рис. 13. Фрагмент коду

ня периметра та основи теж вносяться до системи у відповідні змінні. Далі, програма із наявними значеннями потрапляє на перевірку.

У кодї на рис. 13 видно, що програма визначає значення двох бічних сторін рівнобедреного трикутника та вносить їх до відповідних змінних.

У результаті виконання програма створить назву для заданого рівнобедреного трикутника, оскільки цього не було дано в умові, та поверне, що  $AB = 15.0$  см,  $CA = 15.0$  см. Відповідь коректна.

### Висновок

У статті описано та продемонстровано роботу двох процедур, які базуються на побудованій онтології. Одна із використанням SWRL-правил, інша — з елементами обробки природної мови. Описано застосовані фрагменти таксономічної

ієрархії для предметної онтології планіметрії. На прикладах простих задач відкритого типу з короткою відповіддю проілюстровано їхні можливості. Висвітлено особливості роботи кожної системи.

Під час тестування задач Reasoner у системі Protégé не виявив суперечностей або неточностей. Натомість він знайшов шукане та визначив тип класу, застосовуючи SWRL-правила й обмеження.

Продемонстровано елементи обробки природної української мови, на якій сформульовано задачу.

Варто зазначити, що дві системи впоралися із поставленою задачею та знайшли розв'язок тому, що додатково було прописано певні правила, за допомогою яких це вдалося. Отже, чим повнішою буде побудована онтологія, тим більшу кількість класів задач система зможе охопити.

### Список літератури

1. Глушков В. М. Некоторые проблемы теории автоматов и искусственного интеллекта / В. М. Глушков // Кибернетика. — 1970. — № 2. — С. 3–13.
2. Летишевский А. А. Алгоритм Очевидности Глушкова / А. А. Летишевский, А. В. Лялецкий, М. К. Мороховец // Кибернетика и системный анализ. — 2013. — № 4. — С. 3–16.
3. Рисін А. Великий електронний словник української мови (VESUM) [Електронний ресурс] / А. Рисін, В. Старко. Веб-версія 6.0.1. 2005–2022. Режим доступу: <https://r2u.org.ua/vesum/>.
4. Bechhofer S. OWL: Web Ontology Language / S. Bechhofer // Liu L., Özsu M. T., eds. Encyclopedia of Database Systems. — Springer, Boston, MA, 2009. [https://doi.org/10.1007/978-0-387-39940-9\\_1073](https://doi.org/10.1007/978-0-387-39940-9_1073).
5. List of Reasoners [Electronic resource] // OWL @ Manchester. — Mode of access: <http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/>.
6. Straka Milan. UDPipe 2.0 Prototype at CoNLL 2018 UD Shared Task / Milan Straka // Proceedings of CoNLL 2018: The SIGNLL Conference on Computational Natural Language Learning. — Association for Computational Linguistics, Stroudsburg, PA, USA, 2018. — Pp. 197–207.
7. SWRL: A Semantic Web Rule Language Combining OWL and RuleML [Electronic resource]. — Mode of access: [w3.org](http://w3.org).

### References

- Bechhofer, S. (2009). OWL: Web Ontology Language. In Liu, L., Özsu, M. T. (Eds.), *Encyclopedia of Database Systems*. Springer, Boston, MA. [https://doi.org/10.1007/978-0-387-39940-9\\_1073](https://doi.org/10.1007/978-0-387-39940-9_1073).
- Glushkov, V. M. (1970). Nekotore probleme teorii avtomatov i iskusstvennogo intellekta. *Kibernetika*, 2, 3–13 [in Russian].
- Letichevskii, A. A., Lyaletskii, A. V., & Morokhovets, M. K. (2013). Algoritm Ochevidnosti Glushkova. *Kibernetika i sistemnyi analiz*, 4, 3–16 [in Russian].
- List of Reasoners. OWL @ Manchester. <http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/>.
- Rysin, A., & Starko, V. Velykyi elektronnyi slovnyk ukrainskoi movy (VESUM). Web version 6.0.1. 2005–2022. Mode of access: <https://r2u.org.ua/vesum/>.
- Straka, Milan. (2018). UDPipe 2.0 Prototype at CoNLL 2018 UD Shared Task. In *Proceedings of CoNLL 2018: The SIGNLL Conference on Computational Natural Language Learning*, 197–207. Association for Computational Linguistics, Stroudsburg, PA, USA.
- SWRL: A Semantic Web Rule Language Combining OWL and RuleML. <http://w3.org>.

O. Zhezherun, O. Smysh, A. Prudnikova

## APPROACHES TO INFERENCE SEARCH IN THE ONTOLOGICAL KNOWLEDGE BASE

*The article provides two approaches for the implementation of the inference search procedure in the ontological base. One is based on the SWRL-rules, the other is a system with the natural language processing elements. The procedures have been established as a part of the recommendation system, which is developed at the Faculty of Informatics at National University of Kyiv-Mohyla Academy.*

We also add a description of the created approaches with their fragments of the taxonomic hierarchy for the planimetry ontology. For the performance examples, simple open-type problems with a short answer taken from the school geometry textbooks are used. The features of the approaches, how they work, as well as the capabilities they have are reviewed.

The approach with natural language processing capabilities has a module for preprocessing raw Ukrainian text using the UDPipe 2.12 model, a module for rechecking the lemmas by using VESUM dictionary, a module with a described planimetry ontology, and a module for creating an illustration of the figures (triangles).

To better illustrate the capabilities of the approaches on equal terms, we tried to use the same geometric problem. English translation of the problem: «Perimeter of an isosceles triangle = 40 cm and base = 10 cm. Find the legs of the triangle.». To solve this problem, systems should have a rule that subtracts the base from the perimeter, divides it by two, and sets the result to the correct variables (in our case, the legs of the triangle). We demonstrated that both approaches solved the problem successfully. But in order to achieve it, minor changes were added. Therefore, the more complete the ontology is, the greater the number of problem types the systems are able to cover.

Having analyzed the results of the study, we can conclude that the systems are effective for solving geometric problems. The next step may be to combine the capabilities of the approaches to form a more complete knowledge base.

**Keywords:** natural language processing, recommendation system, geometry, knowledge base, ontology, Protégé, OWL, Reasoner, SWRL.

Матеріал надійшов 15.11.2023



Creative Commons Attribution 4.0 International License (CC BY 4.0)

Бучко О. А., Нгуєн С. Б. В.

## ПОРІВНЯЛЬНИЙ АНАЛІЗ АЛГОРИТМІВ СУПЕРРОЗДІЛЬНОЇ ЗДАТНОСТІ ДЛЯ СТИСНЕННЯ ЗОБРАЖЕНЬ

*У статті розглянуто підхід до стиснення зображень, який передбачає використання методів суперроздільної здатності. Запропонований підхід має потенціал підвищити візуальну якість відтворення зображень і забезпечити ефективніший результат стиснення порівняно з традиційними методами стиснення. Наведено експериментальну оцінку запропонованого алгоритму на основі FSRCNN, EDSR, Real-ESRGAN мереж суперроздільної здатності з використанням кількісних та якісних показників.*

**Ключові слова:** стиснення зображень, суперроздільна здатність, нейронні мережі, комп'ютерний зір.

### Вступ

Стиснення зображень є важливим процесом у сучасній цифровій епосі, коли обмін і зберігання високоякісних зображень набуває все більшого значення. Зростає попит на візуально привабливий контент, а тому зростає і потреба в ефективних методах стиснення зображень, які допомагають зберігати зображення без втрати візуальних деталей. Стиснення зображень балансує між зменшенням розміру файлу та прийнятним рівнем візуальної якості, оскільки значне стиснення може призвести до погіршення якості зображення.

Основним недоліком традиційних методів стиснення є те, що вони часто погіршують якість зображень, призводять до появи артефактів, спричиняють втрату текстури та кольору. Така проблема може бути суттєвою у сферах, де є важливою висока якість зображень, як-от медична візуалізація, супутникові знімки та професійна фотографія.

Методи суперроздільної здатності (англ. Super-Resolution) було винайдено для вирішення цих проблем. Алгоритм покращує якість зображення та відновлює дрібні деталі й текстури, збільшуючи роздільну здатність. Таке інтелектуальне відновлення відбувається за рахунок навчання на великих обсягах даних.

У статті проведено дослідження різновидів алгоритмів суперроздільної здатності та їхню роль у покращенні процесу стиснення зображень. Проаналізовано основні принципи цих методів, їх застосування в процесі стиснення зображень, поточний стан досліджень у цій галузі. За мету було поставлено показати потенці-

ал, який мають алгоритми суперроздільної здатності у розрізі процесів стиснення інформації, а саме зображень і відео, та дати поштовх подальшим дослідженням у цій темі.

### Різновиди алгоритму суперроздільної здатності

Методи суперроздільної здатності не обов'язково використовують нейронні мережі (наприклад, бікубічна інтерполяція), однак дослідження показують, що найбільш ефективні з них базуються на глибокому навчанні. Більшість із методів суперроздільної здатності можна використати у стисненні зображень.

Умовно реалізації суперроздільної здатності можна поділити за такими архітектурами: згорткові, залишкові, рекурсивні, селективні, генеративні, генеративні змагальні мережі.

Згорткові мережі (англ. Convolutional Neural Network) є найбільш популярними з огляду на простоту їх реалізації і гарні результати порівняно з традиційними методами. Найтиповішим прикладом такої категорії є SRCNN (Super-Resolution Convolutional Neural Network) [9]. Цю архітектуру було використано першою у задачі суперроздільної здатності, і вона стала поштовхом до реалізації нових складніших архітектур. Складається така мережа зазвичай із трьох шарів — вилучення, відображення та реконструкції. Навчання відбувається через середню квадратичну помилку (MSE) і пікове співвідношення сигнал/шум (PSNR) [13]. Іншими представниками цієї категорії є VDSR (Very Deep Super Resolution) [11], FSRCNN (Fast Super-Resolution Convolutional Neural Network) [4] та



ESPCN (Efficient Sub-Pixel Convolutional Neural Network) [16].

Залишкові мережі (англ. Residual Networks) використовують комбінацію згорткових шарів і пропускових з'єднань (skip connections). Першою реалізацією цієї архітектури була SRResNet (Super-Resolution Residual Network) [10], однак найбільш популярною стала мережа EDSR (Enhanced Deep Super-Resolution) [5] за рахунок оптимізованого використання ресурсів. Такий підхід також використовують мережі MDSR (Multi-scale Deep Super-Resolution) та CARN (Cascading Residual Network).

Рекурсивні мережі (англ. Recursive Network) [2] мають спільні згорткові шари, які використовуються кілька разів, що дає змогу зменшити обсяг пам'яті. Представниками такого підходу є DRCN (Deep Recursive Convolutional Network), а також DRRN (Deep Recursive Residual Network) [19], який використовує залишкові блоки (англ. residual blocks).

Селективні мережі (англ. Attention-Based Networks) спираються на принцип уваги (англ. attention), що дає можливість виділити деякі ділянки зображення, для яких треба приділити більше зусиль задля кращого результату. У мережі SelNet (Deep Convolutional Neural Network with Selection Units) [1] було запроваджено спеціальні селективні блоки (англ. selection unit), які вибирають, яку інформацію пропускати далі. Схожий принцип має мережа RCAN (Residual Channel Attention Networks), яка використовує RIR (residual in residual) [10] блоки з окремим каналом для уваги.

Генеративні моделі (англ. Generative Models) [7] намагаються оптимізувати візуальне сприйняття, створюючи приємні для людського ока зображення. Такі мережі не намагаються оптимізувати різницю між пікселями, а використовують більш складні метрики для оцінювання. Якщо розглянути значення PSNR, то така модель матиме гірші показники, ніж розглянуті раніше архітектури. Однак така мережа матиме значно більше значення MOS (Mean Opinion Score) [18], тобто кращу візуальну якість.

Генеративні змагальні нейронні мережі (англ. Generative Adversarial Network, GAN) навчаються без учителя, складаються з двох підмереж: «генератора» та «дискримінатора». Генератор генерує зображення, а дискримінатор має «розпізнати» та вирішити, чи є зображення істинним. Найбільш відомими представниками цього класу є SRGAN (Super-Resolution Generative Adversarial Network, генеративна змагальна мережа для підвищення роздільної здатності) [15],

EnhanceNet [17] та ESRGAN (Enhanced Super-Resolution Generative Adversarial Networks) [6].

Згадані мережі гарно інтегруються у процес стиснення зображень. Використання цих мереж дає змогу стискати зображення сильніше, тобто ефективніше, передбачаючи, що ці зображення буде відновлено за допомогою суперроздільної здатності у майбутньому. Найбільш цікавими є генеративні змагальні мережі, оскільки вони спрямовані на відновлення візуально привабливих зображень, що може бути перевагою у деяких сферах.

### Інтеграція суперроздільної здатності у процес стиснення зображень

Алгоритм передбачає стиснення зображень за допомогою відповідних методів, а потім їх відновлення, коли це необхідно, із застосуванням алгоритму суперроздільної здатності для покращення візуальних характеристик. Процес стиснення зображення за допомогою суперроздільної здатності можна описати такими кроками:

1. Стиснення — застосування традиційних методів стиснення, наприклад, JPEG [20] або HEVC [14]. На цьому кроці зменшується розмір файлу. Варто зазначити, що це призведе до втрати дрібних деталей і погіршить загальну якість зображення.
2. Суперроздільна здатність — застосування алгоритму до стисненого зображення. На цьому кроці збільшується роздільна здатність і відновлюються дрібні деталі.
3. Постоброблення — застосування методів оброблення зображень для подальшого вдосконалення відновленого зображення, наприклад збільшення різкості зображення, кольорова корекція, покращення обличчя.

Використовуючи алгоритми суперроздільної здатності в процесі стиснення зображень, можна досягти кращого балансу між розміром файлу та якістю зображення. Стислі зображення можна ефективно передавати та зберігати, забезпечуючи кращу візуальну привабливість. Крім того, алгоритм суперроздільної здатності можна застосовувати в сценаріях реального часу, наприклад для потокового медіа або відеоспостереження, оскільки кадри з низькою роздільною здатністю можна покращувати без затримок завдяки цьому алгоритму. До переваг інтеграції алгоритму суперроздільної здатності до процесу стиснення можна віднести:

- а) покращену якість зображення;
- б) підвищену ефективність стиснення;
- в) адаптивність.

Цікавою особливістю є адаптивність, оскільки запропонований алгоритм можна налаштувати для роботи з різними алгоритмами стиснення та різними мережами суперроздільної здатності. Це дає змогу підлаштовувати процес стиснення до різних вимог. Проте основною перевагою є кращі показники стиснення/якість, ніж у традиційних алгоритмах. Хоча інтеграція алгоритмів суперроздільної здатності дає значні переваги, варто враховувати ризики та проблеми:

- а) обчислювальна складність: чим складніша архітектура мережі суперроздільної здатності, тим краще результат, але тим довше і більше ресурсів буде використовуватись. Це може значно вплинути на використання алгоритму у сферах, де затримки можуть значно погіршити користувацький досвід;
- б) універсальність: зважаючи на різноманітність вимог до стиснення зображень (наприклад, зображення з текстом, пейзажі, медичні знімки тощо), може постати проблема в універсальності алгоритму, оскільки різні поєднання алгоритмів стиснення та суперроздільної здатності можуть давати різні результати для кожного випадку;
- в) компроміс між якістю та ефективністю: як і у традиційних методах стиснення, може виникнути потреба у більшому стисненні зображень, що призведе до погіршення якості зображення. За таких сценаріїв алгоритм суперроздільної здатності може не відновити втрачені деталі й відновлені зображення будуть не лише естетично непривабливими, а й навіть спотвореними.

### Експериментальна оцінка

Щоб переконатися в ефективності запропонованого алгоритму, було проведено експеримент. За допомогою кількісного та якісного аналізу було оцінено потенціал такого підходу до стиснення зображень порівняно з традиційними методами.

Складові експерименту можна поділити на такі основні частини:

- а) тестовий набір даних: набір даних DIV2K [3], який містив 900 зображень високої роздільної здатності. Цей набір зображень часто використовують для тренування мереж суперроздільної здатності. До набору входять різноманітні зображення — тварини, люди, архітектура, природа тощо. Розмір набору даних становить 3 98 ГБ;
- б) метод стиснення: JPEG і зменшення роздільності у 4 рази за допомогою інтерполяції;

- в) мережа суперроздільної здатності: вибрано кілька популярних мереж для порівняння — FSRCNN, EDSR, Real-ESRGAN;
- г) оцінка: щоб оцінити ефективність алгоритму, використовували метрику Frechet Inception Distance (FID) [12].

Варто зазначити, що метрику FID було спеціально підібрано до алгоритму суперроздільної здатності, оскільки вона дає змогу оцінити ефективність мереж з архітектурою GAN. Такі типові метрики, як пікове відношення сигнал/шум (PSNR) або індекс структурної подібності (SSIM) [8], не є об'єктивними у цьому випадку, оскільки не враховують комплексну інформацію про якість зображень.

Оцінка FID базується на претренованій нейронній мережі, через яку пропускають окремі набори зображень, які порівнюють. FID визначається як відстань між розподілом відповідних передавальних шарів (функція активацій). Якщо відстань мала, можна зробити висновок, що зображення подібні.

Використовуючи вищезгаданий набір алгоритмів і даних, було сформовано окремі множини зображень:

- а) оригінальні зображення (ground truth);
- б) зображення, стиснуті у 4 рази;
- в) зображення, відновлені за допомогою суперроздільної здатності (до оригінального розміру);
- г) стиснуті зображення у форматі jpeg, із меншою якістю (5 %).

Щоб порівняти запропонований алгоритм із традиційним методом стиснення, було підібрано параметри, які створили приблизно однакові групи за сумою розмірів файлів (табл. 1). Для алгоритму суперроздільної здатності зображення було стиснуто в 4 рази без зміни параметра якості JPEG.

Таблиця 1

Середнє значення розміру зображень після стиснення відповідними методами

Метод стиснення	Розмір, МБ	Коефіцієнт стиснення
JPEG (5 %)	43,8	9,07
Запропонований	33,7	11,79

Нагадаємо, що коефіцієнт стиснення обчислюють як відношення початкового розміру до стиснутого. Відповідно до експерименту, запропонований алгоритм стиснув зображення приблизно у 12 разів.

Кількісний аналіз засвідчив, що запропонований алгоритм має кращі показники, ніж традиційний метод (табл. 2).

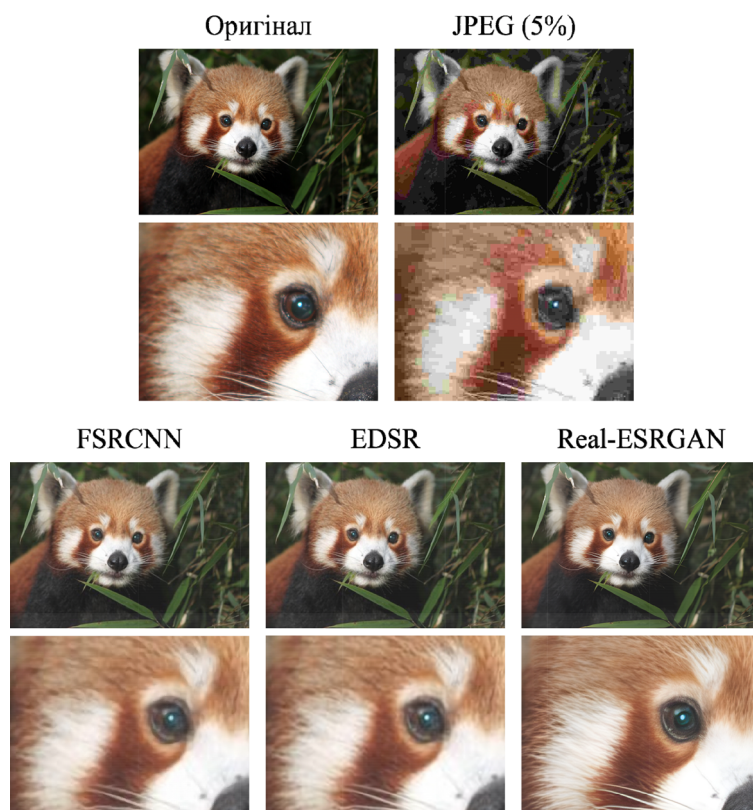


Рис. 1. Порівняльний аналіз методів стиснення та різних мереж суперроздільної здатності

Таблиця 2

FID значення для наведених методів стиснення

Метод стиснення	FID
JPEG (5%)	50.1796
FSRCNN	34.0294
EDSR	31.2186
Real-ESRGAN	27.0468

Якісний аналіз, з іншого боку, передбачав візуальне порівняння відновлених і традиційно стиснутих зображень. Цей аналіз зосереджувався на оцінці суб'єктивної якості, а саме зверталася увага на збереження дрібних деталей, текстур і загальної візуальної привабливості зображень. Порівнюючи результати звичайного стиснення з алгоритмом суперроздільної здатності, можна побачити перевагу запропонованого алгоритму над традиційним: менше артефактів, збереження кольорів, чіткі деталі, візуально привабливе зображення (рис. 1). Найкраще себе зарекомендувала модель Real-ESRGAN, яка належить до генеративних змагальних мереж.

## Висновки

Із проведеного дослідження можна зробити висновок, що інтеграція алгоритмів суперроздільної здатності в методи стиснення може суттєво покращити якість зображень і підвищити ефективність стиснення. Переваги алгоритму можна побачити в отриманих кількісних та якісних показниках. Однак варто враховувати такі ризики, як складність обчислень і універсальність. Використання суперроздільної здатності може стати перспективним напрямом для подальшого вдосконалення процесів стиснення зображень, але варто приділити додаткову увагу оптимізації та адаптації під різні вимоги та сценарії застосування.

Цілком імовірно, що, використовуючи алгоритми суперроздільної здатності, можна запропонувати більш ефективне та дієве рішення, що дасть змогу вдосконалити спосіб зберігання та обмін цифровими зображеннями.

## Список літератури

- Choi J.-S. A Deep Convolutional Neural Network with Selection Units for Super-Resolution [Electronic resource] / Jae-Seok Choi, Munchurl Kim // 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017. — [S. l.], 2017. — <https://doi.org/10.1109/cvprw.2017.153>.
- Dahl R. Pixel Recursive Super Resolution [Electronic resource] / Ryan Dahl, Mohammad Norouzi, Jonathon Shlens //

- 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 22–29 October 2017. — [S. 1.], 2017. — <https://doi.org/10.1109/iccv.2017.581>.
3. DIV2K Dataset [Electronic resource]. — Mode of access: <https://data.vision.ee.ethz.ch/cv/DIV2K/>.
  4. Dong C. Accelerating the Super-Resolution Convolutional Neural Network [Electronic resource] / Chao Dong, Chen Change Loy, Xiaoou Tang // Computer Vision — ECCV 2016. — Cham, 2016. — Pp. 391–407. — [https://doi.org/10.1007/978-3-319-46475-6\\_25](https://doi.org/10.1007/978-3-319-46475-6_25).
  5. Enhanced Deep Residual Networks for Single Image Super-Resolution [Electronic resource] / Bee Lim [et al.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017. — [S. 1.], 2017. — <https://doi.org/10.1109/cvprw.2017.151>.
  6. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks [Electronic resource] / Xintao Wang [et al.] // Lecture Notes in Computer Science. — Cham, 2019. — Pp. 63–79. — [https://doi.org/10.1007/978-3-030-11021-5\\_5](https://doi.org/10.1007/978-3-030-11021-5_5).
  7. Generative adversarial networks [Electronic resource] / Ian Goodfellow [et al.] // Communications of the ACM. — 2020. — Vol. 63, no. 11. — Pp. 139–144. — <https://doi.org/10.1145/3422622>.
  8. Hore A. Image Quality Metrics: PSNR vs. SSIM [Electronic resource] / Alain Hore, Djemel Ziou // 2010 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey, 23–26 August 2010. — [S. 1.], 2010. — <https://doi.org/10.1109/icpr.2010.579>.
  9. Image Super-Resolution Using Deep Convolutional Networks [Electronic resource] / Chao Dong [et al.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2016. — Vol. 38, no. 2. — Pp. 295–307. <https://doi.org/10.1109/tpami.2015.2439281>.
  10. Image Super-Resolution Using Very Deep Residual Channel Attention Networks [Electronic resource] / Yulun Zhang [et al.] // Computer Vision — ECCV 2018. — Cham, 2018. — Pp. 294–310. — [https://doi.org/10.1007/978-3-030-01234-2\\_18](https://doi.org/10.1007/978-3-030-01234-2_18).
  11. Kim J. Accurate Image Super-Resolution Using Very Deep Convolutional Networks [Electronic resource] / Jiwon Kim, Jung Kwon Lee, Kyoung Mu Lee // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. — [S. 1.], 2016. <https://doi.org/10.1109/cvpr.2016.182>.
  12. Obukhov A. Quality Assessment Method for GAN Based on Modified Metrics Inception Score and Fréchet Inception Distance [Electronic resource] / Artem Obukhov, Mikhail Krasnyanskiy // Software Engineering Perspectives in Intelligent Systems. — Cham, 2020. — Pp. 102–114. — [https://doi.org/10.1007/978-3-030-63322-6\\_8](https://doi.org/10.1007/978-3-030-63322-6_8).
  13. On the Computation of PSNR for a Set of Images or Video [Electronic resource] / Onur Keles [et al.] // 2021 Picture Coding Symposium (PCS), Bristol, United Kingdom, 29 June – 2 July 2021. — [S. 1.], 2021. — <https://doi.org/10.1109/pcs50896.2021.9477470>.
  14. Overview of the High Efficiency Video Coding (HEVC) Standard [Electronic resource] / Gary J. Sullivan [et al.] // IEEE Transactions on Circuits and Systems for Video Technology. — 2012. — Vol. 22, no. 12. — Pp. 1649–1668. — <https://doi.org/10.1109/tcsvt.2012.2221191>.
  15. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network [Electronic resource] / Christian Ledig [et al.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 21–26 July 2017. — [S. 1.], 2017. — <https://doi.org/10.1109/cvpr.2017.19>.
  16. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network [Electronic resource] / Wenzhe Shi [et al.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. — [S. 1.], 2016. — <https://doi.org/10.1109/cvpr.2016.207>.
  17. Sajjadi M. S. M. EnhanceNet: Single Image Super-Resolution Through Automated Texture Synthesis [Electronic resource] / Mehdi S. M. Sajjadi, Bernhard Scholkopf, Michael Hirsch // 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 22–29 October 2017. — [S. 1.], 2017. — <https://doi.org/10.1109/iccv.2017.481>.
  18. Strejil R. C. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives [Electronic resource] / Robert C. Strejil, Stefan Winkler, David S. Hands // Multimedia Systems. — 2014. — Vol. 22, no. 2. — Pp. 213–227. — <https://doi.org/10.1007/s00530-014-0446-1>.
  19. Tai Y. Image Super-Resolution via Deep Recursive Residual Network [Electronic resource] / Ying Tai, Jian Yang, Xiaoming Liu // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 21–26 July 2017. — [S. 1.], 2017. — <https://doi.org/10.1109/cvpr.2017.298>.
  20. Wallace G. K. The JPEG still picture compression standard [Electronic resource] / G. K. Wallace // IEEE Transactions on Consumer Electronics. — 1992. — Vol. 38, no. 1. — Pp. xviii–xxxiv. — <https://doi.org/10.1109/30.125072>.

## References

- Choi, J.-S., & Kim, M. (2017). A Deep Convolutional Neural Network with Selection Units for Super-Resolution. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE. <https://doi.org/10.1109/cvprw.2017.153>.
- Dahl, R., Norouzi, M., & Shlens, J. (2017). Pixel Recursive Super Resolution. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE. <https://doi.org/10.1109/iccv.2017.581>.
- DIV2K Dataset*. (n.d.). <https://data.vision.ee.ethz.ch/cv/DIV2K/>.
- Dong, C., Loy, C. C., He, K., & Tang, X. (2016). Image Super-Resolution Using Deep Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2), 295–307. <https://doi.org/10.1109/tpami.2015.2439281>.
- Dong, C., Loy, C. C., & Tang, X. (2016). Accelerating the Super-Resolution Convolutional Neural Network. In *Computer Vision – ECCV 2016* (pp. 391–407). Springer International Publishing. [https://doi.org/10.1007/978-3-319-46475-6\\_25](https://doi.org/10.1007/978-3-319-46475-6_25).
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139–144. <https://doi.org/10.1145/3422622>.
- Hore, A., & Ziou, D. (2010). Image Quality Metrics: PSNR vs. SSIM. In *2010 20th International Conference on Pattern Recognition (ICPR)*. IEEE. <https://doi.org/10.1109/icpr.2010.579>.
- Keles, O., Yilmaz, M. A., Tekalp, A. M., Korkmaz, C., & Dogan, Z. (2021). On the Computation of PSNR for a Set of Images or Video. In *2021 Picture Coding Symposium (PCS)*. IEEE. <https://doi.org/10.1109/pcs50896.2021.9477470>.
- Kim, J., Lee, J. K., & Lee, K. M. (2016). Accurate Image Super-Resolution Using Very Deep Convolutional Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. <https://doi.org/10.1109/cvpr.2016.182>.
- Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., & Shi, W. (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. <https://doi.org/10.1109/cvpr.2017.19>.

- Lim, B., Son, S., Kim, H., Nah, S., & Lee, K. M. (2017). Enhanced Deep Residual Networks for Single Image Super-Resolution. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE. <https://doi.org/10.1109/cvprw.2017.151>.
- Obukhov, A., & Krasnyanskiy, M. (2020). Quality Assessment Method for GAN Based on Modified Metrics Inception Score and Fréchet Inception Distance. In *Software Engineering Perspectives in Intelligent Systems* (pp. 102–114). Springer International Publishing. [https://doi.org/10.1007/978-3-030-63322-6\\_8](https://doi.org/10.1007/978-3-030-63322-6_8)
- Sajjadi, M. S. M., Scholkopf, B., & Hirsch, M. (2017). EnhanceNet: Single Image Super-Resolution Through Automated Texture Synthesis. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE. <https://doi.org/10.1109/iccv.2017.481>.
- Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., & Wang, Z. (2016). Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. <https://doi.org/10.1109/cvpr.2016.207>.
- Strejl, R. C., Winkler, S., & Hands, D. S. (2014). Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22 (2), 213–227. <https://doi.org/10.1007/s00530-014-0446-1>.
- Sullivan, G. J., Ohm, J.-R., Han, W.-J., & Wiegand, T. (2012). Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22 (12), 1649–1668. <https://doi.org/10.1109/tesvt.2012.2221191>.
- Tai, Y., Yang, J., & Liu, X. (2017). Image Super-Resolution via Deep Recursive Residual Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. <https://doi.org/10.1109/cvpr.2017.298>.
- Wallace, G. K. (1992). The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38 (1), xviii–xxxiv. <https://doi.org/10.1109/30.125072>.
- Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., & Loy, C. C. (2019). ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *Lecture Notes in Computer Science* (pp. 63–79). Springer International Publishing. [https://doi.org/10.1007/978-3-030-11021-5\\_5](https://doi.org/10.1007/978-3-030-11021-5_5).
- Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., & Fu, Y. (2018). Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In *Computer Vision – ECCV 2018* (pp. 294–310). Springer International Publishing. [https://doi.org/10.1007/978-3-030-01234-2\\_18](https://doi.org/10.1007/978-3-030-01234-2_18).

O. Buchko, S. B. Nhuien

## COMPARATIVE ANALYSIS OF SUPER-RESOLUTION ALGORITHMS FOR IMAGE COMPRESSION

*Image compression is essential in today's digital age when sharing and storing high-quality images is becoming increasingly important. With the growing demand for visually appealing content, there is also a growing need for efficient image compression methods that help to store images without losing visual details.*

*The main disadvantage of traditional compression methods is that they often degrade image quality, lead to artefacts, and cause loss of texture and colour. This problem can be significant in areas where high image quality is crucial, such as medical imaging, satellite imagery, and professional photography.*

*This paper studies the types of super-resolution algorithms and their role in improving the image compression process. The basic principles of these methods, their application in image compression, and the current state of research in this area are analyzed. The goal was to show the potential of super-resolution algorithms in the context of information compression, images and videos, and to give impetus to further research in this area.*

*The proposed approach to image compression based on super-resolution methods can improve the visual quality of image reproduction and provide more efficient compression results than traditional compression methods. The paper uses quantitative and qualitative indicators to present an experimental evaluation of the proposed algorithm based on FSRCNN, EDSR, and Real-ESRGAN super-resolution networks.*

*The experimental results confirm the high efficiency of the proposed approach. The use of FSRCNN, EDSR, and Real-ESRGAN allows us to improve the reproduction of images with high accuracy and detail.*

*This article generally reveals the importance of super-resolution methods in image compression and provides a basis for further research. Using these methods can bring significant benefits in various fields where image quality is crucial.*

**Keywords:** image compression, super-resolution, neural networks, computer vision.

Матеріал надійшов 15.05.2022



Ліп'яніна-Гончаренко Х. В.

## МЕТОД ФОРМУВАННЯ НАВЧАЛЬНОЇ ВИБІРКИ ДЛЯ МАСИВІВ ДАНИХ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

У цій роботі запропоновано новий метод формування навчальної вибірки на базі машинного навчання, що об'єднує дані з RFM-аналізу та кластерного аналізу. Метод застосовано до даних, отриманих з аукціонів українського сайту ProZorro Продажі. Запропонована вибірка охоплює 92 638 аукціонів, 29 164 унікальні аукціони та 39 747 унікальних організаторів. У процесі RFM-аналізу дані розбито на групи: «Найкращі організатори тендерів», «Вірні організатори тендерів» та ін. Далі, методом K-means, дані були поділено на кластери, що дало змогу відокремити різні категорії організаторів. Результати тестування, проведеного з використанням Logistic Regression і Naïve Bayes, засвідчили високу точність для обох методів. Продемонстровано, що вибірка та групування за допомогою запропонованого методу допомагають відрізнити організаторів тендерів за їхніми характеристиками та результатами. Подальші дослідження мають бути у напрямі розроблення автоматизованої системи для вибору організаторів тендерів на основі машинного навчання, що сприятиме оптимізації участі у тендерних процедурах.

**Ключові слова:** навчальна вибірка, машинне навчання, RFM-аналіз, кластерний аналіз, тендери.

### Вступ

Нині постає необхідність розроблення ефективного методу формування навчальної вибірки для задач машинного навчання, зокрема в контексті аналізу даних RFM-аналізу та кластерного аналізу. Наявні методи формування вибірок часто не забезпечують оптимальний відбір репрезентативних та інформативних зразків для навчання моделей.

Ключові аспекти проблеми такі:

- Недостатня репрезентативність. Традиційні методи формування навчальних вибірок подеколи не враховують різноманітність даних і їх динамічні зміни. Внаслідок цього моделі можуть бути погано навчені на нових, раніше не відомих даних.
- Незначущі дані. Велика кількість шумових або незначущих даних може негативно впливати на якість навчання моделей, призводячи до перенавчання або недонавчання.
- Обмежені ресурси. Для ефективного навчання моделей важливо вибрати найінформативніші дані, особливо якщо обчислювальні або часові ресурси є обмеженими.
- Змінність даних. Динаміка зміни даних може потребувати адаптивності методу формування вибірки, щоби було відображено актуальну структуру даних.

Отже, постановка проблеми передбачає розроблення нового методу формування навчальної вибірки, який би поєднував підходи RFM-аналізу та кластерного аналізу для досягнення оптимальної репрезентативності, враховував значущість даних і був адаптивним до змін. Такий метод має покращити ефективність навчання моделей машинного навчання та забезпечити точніші результати прогнозування на нових даних.

### Огляд літератури

У статті [6] розглянуто методи формування класифікованої навчальної вибірки, що генерується лише за допомогою активних перешкод, для адаптації вагових коефіцієнтів просторових фільтрів за умов поєднаної наявності перешкод. У статті [7] запропоновано адаптивний метод формування класифікованої навчальної вибірки на основі використання порогової оцінки коефіцієнта кореляції міжканалів комбінованих перешкод. У статті [4] представлено новий непараметричний лінійний метод вилучення ознак для класифікації гіперспектральних зображень, який використовує ідеї вікон Парзена для визначення локального середнього значення сусідніх зразків і нові вагові функції для формування матриць розкиду міжкласової та внутрішньокласової варіації. У статті [3] запропоновано алгоритм роз-

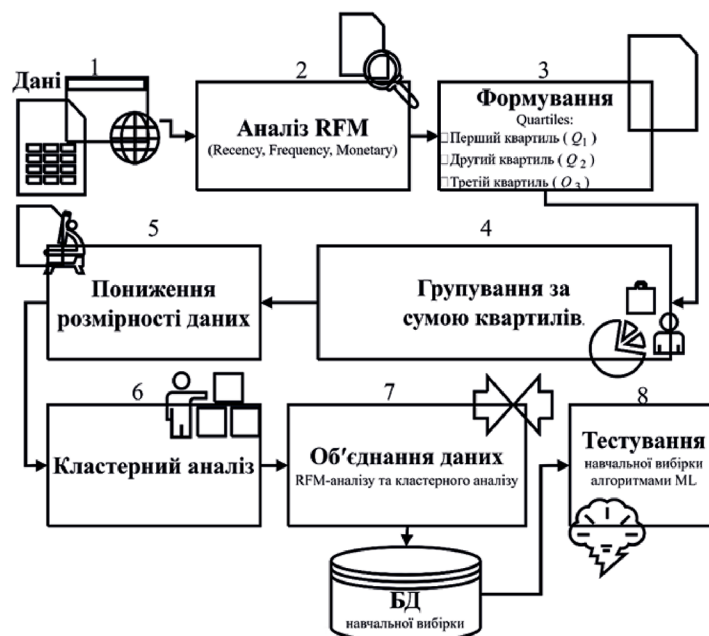


Рис. 1. Структура формування навчальної вибірки для сегментації організаторів тендерів на основі машинного навчання

роблення навчального набору, який найкраще описує об'єкти розпізнавання.

Метою цієї статті є розроблення методу формування навчальної вибірки на основі машинного навчання.

На відміну від аналогів, розроблений метод формування навчальної вибірки на основі машинного навчання дасть змогу автоматизовано та ефективно вибирати найбільш репрезентативні та інформативні зразки для побудови моделей машинного навчання. Основна ідея полягає в тому, щоб використовувати здобуті знання зі сфери RFM-аналізу та кластерного аналізу, групуючи схожі об'єкти разом. Це дає можливість підібрати найбільш репрезентативних представників різних груп даних і їхні характеристики для формування навчальної вибірки.

Застосування такого методу в контексті аналізу та оброблення даних може значно покращити результати навчання моделей, зменшити вплив шуму та незначущих даних, а також зробити процес навчання швидшим і ефективнішим. Характерною особливістю цього підходу є його адаптивність, тобто здатність адекватно реагувати на зміни в навчальних даних і змінювати склад навчальної вибірки відповідно до нової інформації.

### Метод

Авторка розробила метод формування навчальної вибірки на основі машинного навчання. Цей метод (рис. 1) передбачає такі кроки:

Крок 1. Ввід вхідних даних (Блок 1).

Крок 2. RFM-аналіз (Блок 2). Аналіз RFM [1] (Recency, Frequency, Monetary) — це техніка сегментації даних.

Крок 3. Формування Quartiles (Блок 3). Quartiles ділять число точок даних на чотири частини, або чверті, більш-менш однакового розміру. Дані повинні бути впорядковані від найменшого до найбільшого для обчислення кuartилів. Три основні кuartилі такі:

- Перший кuartиль ( $Q_1$ ) — це середнє число між найменшим числом (мінімальним) і медіаною набору даних. Він також відомий як *нижчий* або *25-й емпіричний* кuartиль, оскільки 25 % даних лежать нижче цієї точки.
- Другий кuartиль ( $Q_2$ ) є медіаною набору даних, 50 % даних лежать нижче цієї точки.
- Третій кuartиль ( $Q_3$ ) — це середнє значення між медіаною та найвищим значенням (максимумом) набору даних. Він відомий як *верхній* або *75-й емпіричний* кuartиль, оскільки 75 % даних лежать нижче цієї точки.

Крок 4. Групування за сумою кuartилів (Блок 4).

Крок 5. Пониження розмірності даних (Блок 5). Зниження розмірності є процесом скорочення кількості випадкових змінних шляхом отримання множини головних змінних. Виділення ознак і зниження розмірності можна об'єднати в один етап за допомогою методу головних компонент (МГК), лінійного розділювального аналізу (ЛРА), канонічного кореляційного аналізу (ККА) або розкладення невід'ємних матриць (РНМ).

Крок 6. Кластеризація (Блок 6). Кластерний аналіз [5] — задача розбиття заданої вибірки об'єктів (ситуацій) на підмножини, які називаються кластерами, так, щоб кожен кластер складався зі схожих об'єктів, а об'єкти різних кластерів істотно відрізнялися. Цей аналіз поділяють на такі етапи:

- проведення дослідження;
- підготовка даних до кластерного аналізу;
- вибір методу кластерного аналізу;
- вибір міри відстані між об'єктами та її обчислення;
- вибір стратегії кластеризації;
- застосування вибраної стратегії для утворення кластерів;
- перевірка результатів кластерного аналізу на осмисленість і їх інтерпретацію.

Крок 7. Об'єднання (Блок 7) даних RFM-аналізу та кластерного аналізу та внесення до БД. На основі цих даних можна провести навчання класифікації.

Крок 8. Тестування (Блок 8) навчальної вибірки на основі алгоритмів класифікації машинного навчання.

Для підтвердження розробленого методу формування навчальної вибірки на основі машинного навчання в наступному розділі проведено його реалізацію для сегментації організаторів тендерів.

**Реалізація.** Для формування навчальної вибірки для сегментації організаторів тендерів на основі машинного навчання вибрано мову Python. При цьому використано такі бібліотеки: pandas, numpy, train\_test\_split, KMeans, PCA.

Як вхідні дані використано завершені угоди учасників тендерів в Україні з сайту ProZorro Продажі [8]. Вибірка після очищення становить 93 336 значень відносно 10 параметрів. У процесі оцінювання кількісних показників виявлено 92 638 аукціонів, 29 164 унікальних аукціонів та 39 747 унікальних організаторів.

Далі проведено RFM-аналіз, який допоміг розділити організаторів на різні категорії або кластери, щоб визначити організаторів, які ча-

стіше проводять аукціони з найбільшими сумами. Це три атрибути клієнта для кожного організатора (рис. 2).

ID_Orzanizer	Recency	Frequency	Monetary
1	952	2	1134546
2	412	1	23414
3	154	1	7
4	421	14	8625230
5	1214	1	24225
6	122	9	27834555
7	1	1	16368
8	1086	17	5746360
9	436	4	307177
10	647	4	209575

Рис. 2. RFM-аналіз (head(10))

Щоб розрахувати Recency, потрібно вибрати дату, з якої буде проведено оцінювання.

Frequency угод дасть можливість дізнатися, скільки разів організатор провів угоди. Для цього проведено перевірку, скільки рахунків зареєстровано тим самим організатором.

Monetary визначає, скільки грошей зароблено на угодах організатором.

Найпростіший спосіб просегментувати організаторів — це використати Quartiles, а саме надати оцінки від 1 до 4 Recency, Frequency and Monetary (рис. 3, чотири — найвище значення, одиниця — найнижче значення).

Із рисунку 3 видно, що організатор з ID 408 і 1632 отримали найвищу оцінку, тобто: R\_Quartile = 4, нещодавно проведена угода; F\_Quartile = 4, проведено найбільшу кількість угод; M\_Quartile = 4, зароблено найбільше коштів. Відповідно у цих організаторів тендерів RFMScore = 444.

Оцінку загальної вибірки (рис. 4) проведено за такими критеріями: The best organizers of tenders (RFMScore = 444), Loyal organizers of tenders (F\_Quartile = 4), Large consumers (M\_Quartile = 4), Tenders are seldom held, but for a

ID_Orzanizer	Recency	Frequency	Monetary	Rank	R_Quartile	F_Quartile	M_Quartile	RFMScore
1745	374	21	15920922384	1.0	3	4	4	344
453	353	52	9056844144	2.0	3	4	4	344
408	275	33	5848400325	3.0	4	4	4	444
1290	426	48	4465808177	4.0	2	4	4	244
1632	31	24	3856853376	5.0	4	4	4	444

Рис. 3. RFM-Quartiles



large sum (RFMScore = 114) і Weak tender organizers (RFMScore = 111).

The best organizers of tenders: 119
Loyal organizers of tenders: 385
Large consumers: 455
Tenders are seldom held, but for a large sum: 15
Weak tender organizers: 48

Рис. 4. Оцінка RFM організаторів тендерів

Тепер, коли є сегментація наших організаторів тендерів, можна оцінити кожну групу окремо та проаналізувати, як витрачаються кошти і які організатори найчастіше проводять тендери.

Щоби більше зрозуміти поведінку організатора тендерів, треба глибше вивчити взаємозв'язок між змінними RFM. Тому потрібно поєднати отримані результати з певними прогнозуючими моделями, як-от кластеризація K-means clustering, логістична регресія або рекомендаційна система, для отримання кращих інформативних результатів щодо поведінки організаторів тендерів.

Для групування обрано K-means clustering, оскільки цей простий метод широко використовують для сегментації ринку.

Перед кластеризацією зменшено розмірність даних методом PCA з двома вимірними векторами (компонентами).

За методом ліктя визначено кількість значень на другому кластері. Оцінка Silhouette також є найвищою для другого кластеру. Також спостерігається значне зменшення помилки кластеру з 2 до 5, а після 6 зменшення не є великим. Відповідно обрано  $n\_clusters = 5$ , щоб правильно сегментувати організаторів тендерів.

На рис. 5 подано кластеризацію K-means організаторів тендерів, де кількість кластерів до-

рівнює 5. Графік представлено відносно 2-компонентного PCA методу. На boxplot діаграмі зображено викиди до кожного кластеру, теж у розрізі двокомпонентного PCA методу. До кожного кластеру віднесено таку кількість значень: кластер за номером 0 — 494; кластер за номером 3 — 475; кластер за номером 2 — 352; кластер за номером 1 — 345; кластер за номером 4 — 155.

Аналіз boxplot показав, що перший компонент (кластер 0) має найменше викидів, що вказує на детальний розподіл. Проведено докладний аналіз цього кластеру: мінімум — 0,6, Q1 — 1,2, медіана — 1,5, Q3 — 2, максимум — 3,2; є декілька викидів. Другий кластер (1): мінімум — -2,5, Q1 — -1,5, медіана — -1,4, Q3 — -0,9, максимум — -0,1; викидів немає. Третій кластер (2): мінімум — 0, Q1 — 1,3, медіана — 1,8, Q3 — 2,4, максимум — 3,8; викидів немає. Четвертий кластер (3): мінімум — -2,5, Q1 — -1,3, медіана — -1, Q3 — -0,5, максимум — 0,6; один викид. П'ятий кластер (4): мінімум — -0,6, Q1 — -0,2, медіана — 0,3, Q3 — 0,5, максимум — 1,2; викидів немає.

При порівнянні оцінок RTF та груп K-means з організаторами трендів, виявлено збіг між групою, що активно проводить тендери і залучає значні кошти, та групою, яка мало активна у цьому напрямі і не досягає великих обсягів фінансування. Інші групи організаторів тендерів демонструють лише частковий збіг.

На основі цих даних можна проводити передбачення груп за допомогою методів машинного навчання. Для цього використано метод Logistic Regression [9] та Naive Bayes [2], адже ці методи мають найпростішу логіку кваліфікації та хороші результати оцінки моделювання.

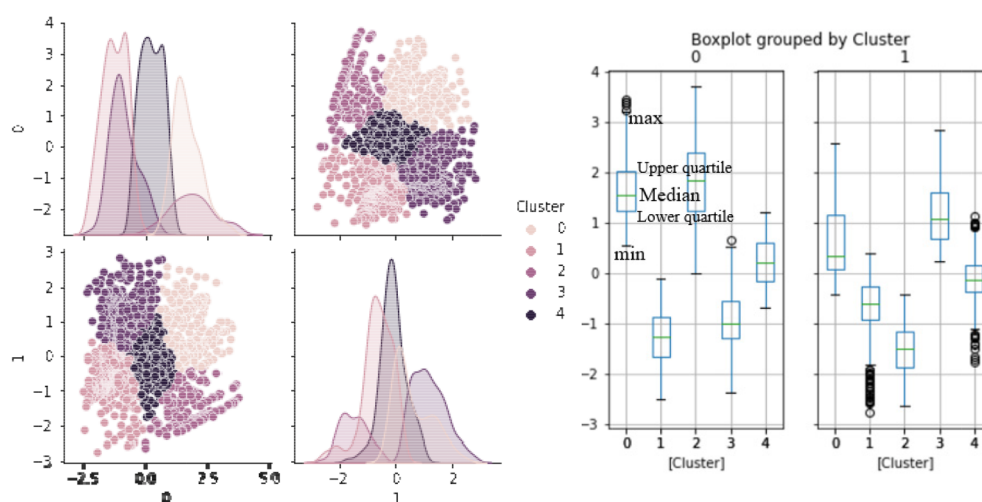


Рис. 5. Кластеризація K-means організаторів тендерів

ID_Ornizer	Recency	Frequency	Monetary	Cluster	RFMScore
1551	30.0	1.0	3060000.0	3	413
695	309.0	1.0	2598260.0	3	413
954	1.0	1.0	1401975.0	4	413
598	309.0	1.0	1140000.0	3	413
579	30.0	1.0	1070856.0	3	413
...	...	...	...	...	...
1349	317.0	1.0	43.0	0	411
1754	30.0	1.0	36.0	0	411
1214	317.0	1.0	32.0	0	411
300	290.0	1.0	24.0	0	411
3	154.0	1.0	7.0	0	411

Рис. 6. Результат кластеризації RTF оцінки та K-means

Для навчання взято 70 % вибірки. Проведено навчання алгоритмами Logistic Regression і Naive Bayes. Після проведення тестування результатами оцінювання є, для обох методів:

Train Set Accuracy for Power Transformed Data: 100.0 %  
Test Set Accuracy for Power Transformed Data: 100.0 %

Bias Error: 0.0  
Variance Error: 0.0

Результати моделювання свідчать, що RTF оцінки та K-means дають стовідсоткову точність групування, відповідно за цими даними у подальшому можна проводити класифікацію організаторів тендерних проєктів, що дає змогу визначати більш привабливих організаторів тендерів.

Отже, вибірка містить 92 638 аукціонів, 29 164 унікальні аукціони та 39 747 унікальних організаторів. На основі RFM-аналізу сформовано такі групи: The best organizers of tenders — 119; Loyal organizers of tenders — 385; Large consumers — 455; Tenders are seldom held, but for a large sum — 15; Weak tender organizers — 48. На основі кластеризації за методом K-means віднесено таку кількість значень: кластер за номером 0 — 494; кластер за номером 3 — 475; кластер за номером 2 — 352; кластер за номером

1 — 345; кластер за номером 4 — 155. Після проведення тестування алгоритмами Logistic Regression та Naive Bayes, результатами оцінювання є для обох методів: Train Set Accuracy for Power Transformed Data — 100 %; Test Set Accuracy for Power Transformed Data — 100 %.

**Висновок.** Авторка розробила метод формування навчальної вибірки на основі машинного навчання, що дає змогу сформувати вибірку на основі об'єднання даних RFM-аналізу та кластерного аналізу. Метод реалізований на основі вхідних даних за завершеними угодами учасників тендерів в Україні з сайту ProZorro Продажі. Вибірка містить 92 638 аукціонів, 29 164 унікальні аукціони та 39 747 унікальних організаторів. На основі RFM-аналізу сформовано такі групи: The best organizers of tenders — 119; Loyal organizers of tenders — 385; Large consumers — 455; Tenders are seldom held, but for a large sum — 15; Weak tender organizers — 48. На основі кластеризації методом K-means віднесено таку кількість значень: кластер за номером 0 — 494; кластер за номером 3 — 475; кластер за номером 2 — 352; кластер за номером 1 — 345; кластер за номером 4 — 155. Порівнюючи оцінки RTF та групи K-means з організаторами трендів, виявлено збіг між групою, що активно організовує тендери та залучає значні кошти, і групою, яка мало залучена до проведення тендерів та не отримує значних сум. Тестування даних проведено алгоритмами Logistic Regression і Naive Bayes. Після проведення тестування результатами оцінювання є, для обох методів: Train Set Accuracy for Power Transformed Data — 100 %; Test Set Accuracy for Power Transformed Data — 100 %.

До напрямів подальших наукових досліджень слід віднести розроблення автоматизованої системи для вибору організаторів тендерів на основі машинного навчання, що також дасть можливість автоматизувати процес участі у тендері.

#### Список літератури

1. Anitha P. RFM model for customer purchase behavior using K-Means algorithm [Electronic resource] / P. Anitha, Malini M. Patil // Journal of King Saud University — Computer and Information Sciences. — 2019. — <https://doi.org/10.1016/j.jksuci.2019.12.011>.
2. Classification Method of Fictitious Enterprises Based on Gaussian Naive Bayes [Electronic resource] / Andriy Krysovaty [et al.] // 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 22–25 September 2021. — [S. l.], 2021. — <https://doi.org/10.1109/csit52700.2021.9648584>.
3. Kamilov M. Algorithm for the Development of a Training Set that Best Describes the Objects of Recognition [Electronic resource] / M. Kamilov, M. Hudayberdiev, A. Khamroev // Procedia Computer Science. — 2019. — Vol. 150. — Pp. 116–122. — <https://doi.org/10.1016/j.procs.2019.02.024> (date of access: 18.08.2023).
4. Kianisarkaleh A. Nonparametric feature extraction for classification of hyperspectral images with limited training samples [Electronic resource] / Azadeh Kianisarkaleh, Hassan Ghassemian // ISPRS Journal of Photogrammetry and Remote Sensing. — 2016. — Vol. 119. — Pp. 64–78. — <https://doi.org/10.1016/j.isprsjprs.2016.05.009>.
5. Li G. Application of Improved K-Means Clustering Algorithm in Customer Segmentation [Electronic resource] / Gang Li // Applied Mechanics and Materials. — 2013. — Vol. 411–414. — Pp. 1081–1084. — <https://doi.org/10.4028/www.scientific.net/amm.411-414.1081>.
6. Method of Forming Classified Training Sample in Case of Spatial Signal Processing under Influence of Combined Interference

- [Electronic resource] / D. M. Piza [et al.] // *Radioelectronics and Communications Systems*. — 2018. — Vol. 61, no. 7. — Pp. 325–331. — <https://doi.org/10.3103/s0735272718070051>.
7. Piza D. M. Methods of Forming Classified Training Sample for Adaptation of Weight Coefficient of Automatic Interference Compensator [Electronic resource] / D. M. Piza, G. V. Moroz // *Radioelectronics and Communications Systems*. — 2018. — Vol. 61, no. 1. — Pp. 32–37. — <https://doi.org/10.3103/s0735272718010041>.
  8. ProZorro Продажі. — <https://bi.prozorro.sale/#/participants-Card>.
  9. Recognizing the Fictitious Business Entity on Logistic Regression Base [Electronic resource] / Andriy Krysovaty, [et al.] // *Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security Khmelnytskyi*. — 2022. — Vol. 3156. — Pp. 218–227. — <https://ceur-ws.org/Vol-3156/paper15.pdf>.

### References

- Anitha, P., & Patil, M. M. (2019). RFM model for customer purchase behavior using K-Means algorithm. *Journal of King Saud University — Computer and Information Sciences*. <https://doi.org/10.1016/j.jksuci.2019.12.011>
- Kamilov, M., Hudayberdiev, M., & Khamroev, A. (2019b). Algorithm for the Development of a Training Set that Best Describes the Objects of Recognition. *Procedia Computer Science*, 150, 116–122. <https://doi.org/10.1016/j.procs.2019.02.024>.
- Kianisarkaleh, A., & Ghassemian, H. (2016). Nonparametric feature extraction for classification of hyperspectral images with limited training samples. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119, 64–78. <https://doi.org/10.1016/j.isprsjprs.2016.05.009>.
- Krysovaty, A., Lipianina-Honcharenko, H., Sachenko, S., Desyatnyuk, O., Banasik, A., & Lukasevych-Krutnyk, I. (2022). Recognizing the Fictitious Business Entity on Logistic Regression Base. *Proceedings of the 3rd International Workshop on Intelligent Information Technologies & Systems of Information Security Khmelnytskyi*, 3156, 218–227. <https://ceur-ws.org/Vol-3156/paper15.pdf>.
- Krysovaty, A., Lipianina-Goncharenko, H., Desyatnyuk, O., Sachenko, S., Lukasevych-Krutnyk, I., & Butrin-Boka, N. (2021). Classification Method of Fictitious Enterprises Based on Gaussian Naive Bayes. In *2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT)*. IEEE. <https://doi.org/10.1109/csit52700.2021.9648584>.
- Li, G. (2013). Application of Improved K-Means Clustering Algorithm in Customer Segmentation. *Applied Mechanics and Materials*, 411–414, 1081–1084. <https://doi.org/10.4028/www.scientific.net/amm.411-414.1081>.
- Piza, D. M., & Moroz, G. V. (2018). Methods of Forming Classified Training Sample for Adaptation of Weight Coefficient of Automatic Interference Compensator. *Radioelectronics and Communications Systems*, 61 (1), 32–37. <https://doi.org/10.3103/s0735272718010041>.
- Piza, D. M., Bugrova, T. I., Lavrentiev, V. N., & Semenov, D. S. (2018). Method of Forming Classified Training Sample in Case of Spacial Signal Processing under Influence of Combined Interference. *Radioelectronics and Communications Systems*, 61 (7), 325–331. <https://doi.org/10.3103/s0735272718070051>.
- ProZorro. <https://bi.prozorro.sale/#/participantsCard>.

Kh. Lipianina-Honcharenko

## METHOD FOR FORMING TRAINING SAMPLES FOR DATA ARRAYS BASED ON MACHINE LEARNING

The study introduces an innovative methodology for crafting training samples through the integration of machine learning techniques. This method encompasses a fusion of RFM (Recency, Frequency, Monetary) analysis and cluster analysis, offering a comprehensive approach to sample formation. The application of this approach is demonstrated on a dataset derived from concluded tender agreements by participants in Ukraine, sourced from the ProZorro Sales platform. The compiled dataset encompasses an impressive volume, encompassing a total of 92,638 auctions, which further breaks down into 29,164 distinct auctions and an assemblage of 39,747 unique organizers.

The utilization of RFM analysis within this framework yields the categorization of the dataset into distinct groups, each characterized by its own distinct attributes. These groupings include designations such as “The Best Organizers of Tenders,” “Loyal Organizers of Tenders,” “Large Consumers,” “Tenders Held Infrequently but with Substantial Sums,” and “Weak Tender Organizers.” Following the RFM analysis, the K-means clustering methodology is implemented, resulting in the division of the data into five clusters, each contributing to a nuanced differentiation of diverse organizer profiles.

Intriguingly, a comparative analysis involving RTF (Relative Total Frequency) scores and the K-means groupings reveals congruence between clusters representing organizers who actively orchestrate numerous tenders with significant monetary value, as well as clusters characterized by minimal tender activity with less substantial monetary implications. To validate the efficacy of the proposed method, rigorous testing is conducted employing Logistic Regression and Naive Bayes algorithms. Encouragingly, the results consistently showcase impressive accuracy for both methods, highlighting their robustness.

An outlook towards future research endeavors suggests a promising avenue of developing an automated system for the selection of tender organizers, underpinned by machine learning principles. Such a system would undoubtedly revolutionize the optimization of participation strategies within the domain of tender processes, fostering efficiency and accuracy in decision-making.

**Keywords:** training sample, machine learning, RFM analysis, cluster analysis, tenders.



Царинюк О. В.

## ЗАСТОСУВАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ КЛАСИФІКАЦІЇ СУПУТНИКОВИХ ЗОБРАЖЕНЬ: ОГЛЯД ЛІТЕРАТУРИ ТА КЛЮЧОВИХ ФРЕЙМВОРКІВ

*У розвитку цифрових технологій і космічної галузі значущу роль відведено ефективному аналізу супутникових зображень. У цій статті докладно розглянуто методи машинного навчання, які застосовують для класифікації таких зображень, а також для вивчення ключових інструментів і фреймворків, які активно використовують у цьому напрямі. В основі дослідження — аналіз сучасних підходів до класифікації, зокрема за допомогою глибоких нейронних мереж. Окремий розділ присвячено популярним фреймворкам, як-от PyTorch, TensorFlow тощо, їхнім особливостям під час роботи з геоданими. Для об'єктивності подано літературний огляд відомих досліджень і робіт, що вплинули на розвиток цієї галузі.*

**Ключові слова:** машинне навчання, класифікація зображень, супутникові знімки, глибоке навчання, згорткові нейронні мережі, PyTorch, TensorFlow, ГІС.

### Вступ

В епоху цифрової трансформації актуальність і точність геоданих стають вирішальними для безлічі сфер, від містобудування до екологічного моніторингу. Цифрові карти, які використовують у найрізноманітніших галузях, потрібно постійно оновлювати й уточнювати. Географічні інформаційні системи (ГІС) є ключовим інструментом у цьому процесі, а швидкість та якість оброблення геоданих визначають їхню конкурентоспроможність.

Розвиток космічних технологій і методів дистанційного зондування надає можливість отримувати дедалі більше високодеталізованих супутникових зображень Землі, що стають основою для формування геоданих і створення цифрових карт. Проте перетворення цих супутникових зображень на векторизовані геодані залишається трудомістким завданням, що потребує значних ресурсів.

Машинне навчання відкриває нові горизонти для автоматизації процесів розпізнавання та векторизації супутникових зображень. Застосування сучасних алгоритмів і фреймворків машинного навчання може в рази зменшити витрати часу та людських ресурсів на виробництво геоданих, а також забезпечити більшу точність і масштабованість оброблення.

У цій статті проведено комплексний огляд методів машинного навчання, які застосовують для розпізнавання супутникових зображень,

а також проаналізовано основні фреймворки, що використовують у цій галузі.

### Методи машинного навчання

Класифікація об'єктів на супутникових зображеннях за допомогою методів машинного навчання — це комплексний процес, який складається з кількох етапів: навчання класифікатора та власне класифікації зображення. Ці етапи відіграють важливу роль у розпізнаванні об'єктів і можуть бути реалізовані різними методами.

Класифікація полягає у наданні кожному пікселю растрового зображення певної категорії або класу на основі його спектральних характеристик.

Нижче розглянуто основні підходи з використанням технологій машинного навчання.

*Histogram of Oriented Gradients* (HOG, гістограма орієнтованих градієнтів) — це техніка визначення ознак (features extraction) на зображеннях, яку часто використовують у завданнях комп'ютерного зору.

Основна ідея HOG полягає в отриманні інформації про градієнти (напрямок зміни інтенсивності) в різних частинах зображення. Надалі інформація компонується у формі гістограми.

HOG є дуже ефективним для виявлення об'єктів, наприклад людських облич або пішохідів. Першу пропозицію щодо використання ознаки HOG як методу розпізнавання об'єктів зробили Dalal і Triggs у 2005 р. [4, р. 886].

Цю методику часто використовують у поєднанні з класифікаторами, як-от SVM (машини опорних векторів), для виявлення об'єктів на зображеннях.

*Support Vector Machine* (SVM, машина опорних векторів) — це лінійний бінарний класифікатор, який може бути також використаний для регресії та виявлення змін класів. Основна ідея SVM — знаходження «оптимальної гіперплощини», яка найкраще розділяє два класи даних. Цей метод було запропоновано у статті «An Overview of Statistical Learning Theory» [14, p. 988], зараз його широко використовують для виявлення доріг [5, p. 3906; 6, p. 1977; 12, p. 1365].

*Artificial Neural Network* (ANN, штучна нейронна мережа) було запропоновано на початку 1980-х рр. Вона має здатність розпізнавати патерни, складність яких робить їх важкими для аналізу за допомогою інших звичайних підходів [2, p. 11].

Деякі типові архітектури нейронних мереж, наприклад згорткові нейронні мережі (Convolutional Neural Network, CNN) [7, p. 1422], з успіхом використовують у багатьох додатках дистанційного зондування, зокрема для виявлення доріг [10, p. 32; 15, p. 3144].

### Згорткові нейронні мережі

Великі обсяги супутникових даних потребують ефективних методів оброблення, щоб перетворити ці дані на корисну інформацію. Згорткові нейронні мережі (ЗНМ) демонструють високу ефективність у низці завдань зорового розпізнавання і, як показали дослідження, можуть бути особливо корисними для розпізнавання растрових зображень. На сьогодні існує велика кількість наукових публікацій, у яких досліджено й підтверджено ефективність згорткових нейронних мереж для задач розпізнавання супутникових знімків.

У статті «Imagenet classification with deep convolutional neural networks» [8, p. 84] було представлено архітектуру AlexNet, яка суттєво випередила інші методи на конкурсі ImageNet у 2012 р. й стала однією з основних робіт, що популяризувала використання ЗНМ для задач розпізнавання зображень.

У статті «Remote Sensing Image Scene Classification: Benchmark and State of the Art» зроблено огляд методів класифікації сцен супутникових знімків, з акцентом на глибокі методи навчання, зокрема ЗНМ [3, p. 1865].

Однією з найбільш упізнаваних архітектур у контексті семантичної сегментації є U-net [11, p. 1]. Початково розроблена для задач мікроско-

пії клітин, ця архітектура показала свою ефективність у вирішенні завдань, у яких потрібно виконувати високоточну локалізацію. Специфіка архітектури U-net, що містить етапи стиснення та розширення, робить її особливо придатною для задач, де важлива просторова інформація, що є характерним для супутникових знімків.

Серед прикладів успішного використання U-net для розпізнавання супутникових зображень можна згадати такі публікації. У статті «M-UNet: Modified U-Net Segmentation Framework with Satellite Imagery» [13, p. 47] було запропоновано модифіковану архітектуру U-Net, яка використовує отримання контекстної інформації з супутникових зображень. У статті «Deep learning for classification of hyperspectral data: A comparative review» [1, p. 159] подано огляд глибокого навчання для класифікації гіперспектральних даних з акцентом на архітектури, наприклад U-net. У статті «Application of UNet Fully Convolutional Neural Network to Impervious Surface Segmentation in Urban Environment from High Resolution Satellite Imagery» [9, p. 3915] описано застосування архітектури U-Net для автоматичного розпізнавання різних типів поверхонь на супутникових знімках високої роздільної здатності.

Високу ефективність ЗНМ у розв'язанні задач розпізнавання і класифікації супутникових знімків зумовлено такими факторами:

- *ієрархічне представлення даних*: ЗНМ автоматично навчаються ієрархічним способом, виявляючи низькорівневі ознаки (як-от краї або текстури) в початкових шарах і складніші абстракції в глибших шарах;
- *локальність і неперервність*: завдяки операції згортки ЗНМ здатні концентруватися на локальних ознаках зображення, що є особливо корисним для аналізу супутникових знімків, де об'єкти можуть мати різний розмір та орієнтацію;
- *масштабованість*: ЗНМ можуть ефективно обробляти великі зображення, які часто трапляються в супутникових знімках;
- *Transfer Learning*: ЗНМ дають змогу використовувати попередньо навчені моделі на великих наборах даних (наприклад, ImageNet) для специфічних задач супутникового розпізнавання з меншою кількістю даних.

Однак твердження, що ЗНМ — це найефективніший метод для всіх задач розпізнавання супутникових знімків, може бути занадто загальним. Залежно від конкретної задачі, роздільної здатності зображення, доступних даних для

навчання та інших факторів, інші методи машинного навчання або комбінації методів можуть бути також доречними.

### Огляд фреймворків для машинного навчання

Для сучасного моделювання та реалізації алгоритмів машинного навчання дослідники та розробники мають використовувати продуктивні й адаптивні інструменти. У цьому розділі розглянуто провідні фреймворки у сфері машинного навчання.

*PyTorch* — це відкрита платформа для глибокого навчання, розроблена командою Facebook's AI Research (FAIR). Вона надає динамічний граф обчислень, що робить його дуже корисним для дослідження.

Особливості:

- динамічний граф обчислень;
- інтуїтивний API для створення та навчання моделей;
- ефективна інтеграція з CUDA для прискореного обчислення на GPU;
- широкий спектр вбудованих функцій для глибокого навчання.

*TensorFlow* — це відкритий фреймворк для машинного навчання, розроблений командою Google Brain. Його можна використовувати для різних задач, від класичних алгоритмів машинного навчання до глибокого навчання.

Особливості:

- статичний граф обчислень, хоча з інтродукцією TensorFlow 2.x було додано підтримку динамічних графів за допомогою tf.eager;
- розширена інтеграція з TensorBoard для візуалізації та моніторингу;
- підтримка для розподілених обчислень і робота на мобільних пристроях.

*Keras* — високорівневий інтерфейс для створення моделей глибокого навчання, який можна виконувати на основі TensorFlow, Theano або CNTK.

*Caffe* — відкритий фреймворк, який в основному використовують для задач, пов'язаних із зображеннями. Розроблений у Каліфорнійському університеті в Берклі.

*MXNet* — гнучкий фреймворк для глибокого навчання, що підтримує різні мови програмування.

*Theano* — один із перших фреймворків для глибокого навчання, нині не дуже активно розробляється, але відіграв важливу роль у становленні екосистеми глибокого навчання.

З-поміж численних доступних фреймворків машинного навчання вибір оптимального інструменту часто визначається специфічними потребами та вимогами до вашого проекту. Попри велике різноманіття опцій, PyTorch і TensorFlow залишаються на вершині популярності. Це можна пояснити їхньою гнучкістю, високою продуктивністю та активними спільнотами розробників і користувачів.

Розглядаючи питання вибору фреймворку з погляду оброблення геоданих, PyTorch вирізняється завдяки інтеграції з системою ArcGIS. Така комбінація надає дослідникам і фахівцям можливість використовувати єдину платформу для всього процесу: від оброблення геоданих до тренування та впровадження моделей. Особливо важливо зазначити, що ArcGIS надає засоби для роботи з просторовими зображеннями у відповідних координатах, у TensorFlow такої можливості немає.

### Висновки

Процес оброблення супутникових знімків завжди супроводжується викликами, зумовленими великою кількістю даних, неоднорідністю ландшафту та змінами у земних покриттях. Традиційні методи часто виявляються часомісткими й потребують великої кількості ручної роботи. З появою та розвитком методів машинного навчання, зокрема глибокого навчання, стало можливим значно автоматизувати процеси розпізнавання та класифікації.

Аналіз, представлений у цій статті, демонструє, що сучасні алгоритми, як-от ANN і ЗНМ, можуть значно підвищити ефективність оброблення зображень. Однак, як і у всіх інших галузях, важливо вибрати правильний інструмент для конкретної задачі.

PyTorch, завдяки своїй інтеграції з ArcGIS, є відмінним інструментом для роботи з геоданими. Однак TensorFlow і інші фреймворки також можуть виявитися корисними залежно від специфіки задачі та вимог.

Враховуючи швидкий розвиток технологій і постійні наукові дослідження у сфері машинного навчання, можна очікувати появи нових, ще більш удосконалених методів і технік у найближчому майбутньому. Незалежно від вибраного методу або фреймворку, головною метою залишається покращення якості та швидкості аналізу геоданих для різноманітних застосувань, від моніторингу навколишнього середовища до міського планування.

## Список літератури

1. Audebert Nicolas. Deep Learning for Classification of Hyperspectral Data: A Comparative Review / Nicolas Audebert, Bertrand Le Saux, Sebastien Lefevre // *IEEE Geoscience and Remote Sensing Magazine*. — 2019. — Vol. 7 (2). — Pp. 159–173.
2. Cheng Gong. A Survey on Object Detection in Optical Remote Sensing Images / Gong Cheng, Junwei Han // *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing*. — 2016. — Vol. 117 (July). — Pp. 11–28.
3. Cheng Gong. Remote Sensing Image Scene Classification: Benchmark and State of the Art / Gong Cheng, Han Junwei, Xiaoqiang Lu // *Proceedings of the IEEE*. — 2017. — Vol. 105 (10). — Pp. 1865–1883.
4. Dalal N. Histograms of Oriented Gradients for Human Detection / N. Dalal, B. Triggs // *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. — 2005. — Vol. 1. — Pp. 886–893.
5. Das Sukhendu. Use of Salient Features for the Design of a Multistage Framework to Extract Roads From High-Resolution Multispectral Satellite Images / Sukhendu Das, T. T. Mirmalinee, Koshy Varghese // *IEEE Transactions on Geoscience and Remote Sensing: A Publication of the IEEE Geoscience and Remote Sensing Society*. — 2011. — Vol. 49 (10). — Pp. 3906–3931.
6. Huang Xin. Road Centreline Extraction from High-resolution Imagery Based on Multiscale Structural Features and Support Vector Machines / Xin Huang, Liangpei Zhang // *International Journal of Remote Sensing*. — 2009. — Vol. 30 (8). — Pp. 1977–1987.
7. Jin Xiaoying. Vehicle Detection from High-Resolution Satellite Imagery Using Morphological Shared-Weight Neural Networks / Xiaoying Jin, Curt H. Davis // *Image and Vision Computing*. — 2007. — Vol. 25 (9). — Pp. 1422–1431.
8. Krizhevsky Alex. Imagenet Classification with Deep Convolutional Neural Networks [Electronic resource] / Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton // *Advances in Neural Information Processing Systems*. — 2012. — Vol. 25. — Mode of access: <https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html>.
9. McGlinchy. Application of UNet Fully Convolutional Neural Network to Impervious Surface Segmentation in Urban Environment from High Resolution Satellite Imagery / Joe McGlinchy, Brian Johnson, Brian Muller, Maxwell Joseph, Jeremy Diaz // *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*. — IEEE, 2019. — Pp. 3915–3918.
10. Mokhtarzade M. Road Detection from High-Resolution Satellite Images Using Artificial Neural Networks / M. Mokhtarzade, M. J. Valadan Zojj // *International Journal of Applied Earth Observation and Geoinformation*. — 2007. — Vol. 9 (1). — Pp. 32–40.
11. Ronneberger Olaf. U-Net: Convolutional Networks for Biomedical Image Segmentation [Electronic resource] / Olaf Ronneberger, Philipp Fischer, Thomas Brox // *arXiv [cs.CV]*. 2015. — Mode of access: <http://arxiv.org/abs/1505.04597>.
12. Song Ming Jun. Road Extraction Using SVM and Image Segmentation / Ming Jun Song, Daniel Civco. — *Photogrammetric Engineering & Remote Sensing*. — 2004. — No. 12. — Pp. 1365–1371.
13. Soni Ashish. M-Net: Modified U-Net Segmentation Framework with Satellite Imagery / Ashish Soni, Radhakanta Koner, Vasant Govind Kumar Villuri // *Proceedings of the Global AI Congress 2019*. — Springer Singapore, 2020. — Pp. 47–59.
14. Vapnik V. N. An Overview of Statistical Learning Theory / V. N. Vapnik // *IEEE Transactions on Neural Networks*. — 1999. — Vol. 10 (5). — Pp. 988–999.
15. Wang Jun. Road Network Extraction: A Neural-Dynamic Framework Based on Deep Learning and a Finite State Machine / Jun Wang, Jingwei Song, Mingquan Chen, Zhi Yang // *International Journal of Remote Sensing*. — 2015. — Vol. 36 (12). — Pp. 3144–3169.

## References

- Audebert, N., Le Saux, B., & Lefevre, S. (2019). Deep Learning for Classification of Hyperspectral Data: A Comparative Review. *IEEE Geoscience and Remote Sensing Magazine*, 7 (2), 159–173.
- Cheng, G., & Han, J. (2016). A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing: Official Publication of the International Society for Photogrammetry and Remote Sensing*, 117, 11–28.
- Cheng, G., Han, J., & Lu, X. (2017). Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 105 (10), 1865–1883.
- Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 1, 886–893.
- Das, S., Mirmalinee, T. T., & Varghese, K. (2011). Use of Salient Features for the Design of a Multistage Framework to Extract Roads From High-Resolution Multispectral Satellite Images. *IEEE Transactions on Geoscience and Remote Sensing: A Publication of the IEEE Geoscience and Remote Sensing Society*, 49 (10), 3906–3931.
- Huang, X., & Zhang, L. (2009). Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines. *International Journal of Remote Sensing*, 30 (8), 1977–1987.
- Jin, X., & Davis, C. H. (2007). Vehicle detection from high-resolution satellite imagery using morphological shared-weight neural networks. *Image and Vision Computing*, 25 (9), 1422–1431.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25. <https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html>.
- McGlinchy, J., Johnson, B., Muller, B., Joseph, M., & Diaz, J. (2019). Application of UNet Fully Convolutional Neural Network to Impervious Surface Segmentation in Urban Environment from High Resolution Satellite Imagery. *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 3915–3918.
- Mokhtarzade, M., & Zojj, M. J. V. (2007). Road detection from high-resolution satellite images using artificial neural networks. *International Journal of Applied Earth Observation and Geoinformation*, 9 (1), 32–40.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In *arXiv [cs.CV]*. arXiv. <http://arxiv.org/abs/1505.04597>.
- Song, M. J., & Civco, D. (2004). Road Extraction Using SVM and Image Segmentation. *Photogrammetric Engineering & Remote Sensing*, 12, 1365–1371.
- Soni, A., Koner, R., & Villuri, V. G. K. (2020). M-UNet: Modified U-Net Segmentation Framework with Satellite Imagery. *Proceedings of the Global AI Congress 2019*, 47–59.
- Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE Transactions on Neural Networks / a Publication of the IEEE Neural Networks Council*, 10 (5), 988–999.
- Wang, J., Song, J., Chen, M., & Yang, Z. (2015). Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine. *International Journal of Remote Sensing*, 36 (12), 3144–3169.

O. Tsaryniuk

## APPLICATION OF MACHINE LEARNING METHODS FOR SATELLITE IMAGE CLASSIFICATION: A LITERATURE REVIEW AND OVERVIEW OF KEY FRAMEWORKS

*In the era of digital technology and space exploration, efficient analysis of satellite imagery has become increasingly crucial. This article provides a comprehensive examination of machine learning methods applied to the classification of satellite images, alongside an exploration of key tools and frameworks utilized in this field. Central to this research is the analysis of contemporary classification approaches, particularly through the use of deep neural networks.*

*The article delves into the significance of satellite imagery in various applications, ranging from urban planning and environmental monitoring to agricultural and disaster management. It emphasizes the challenges posed by the vast volume and high resolution of satellite data, underscoring the necessity for sophisticated analytical tools.*

*A substantial focus is placed on convolutional neural networks (CNNs), renowned for their efficacy in image recognition tasks. The article reviews the evolution of CNNs and their pivotal role in advancing satellite image analysis, highlighted by case studies and successful applications.*

*Additionally, this study offers a look at PyTorch and TensorFlow, two of the most prominent machine learning frameworks. Their integration, features, and suitability for geospatial data analysis are critically evaluated. The article discusses PyTorch's notable integration with ArcGIS, providing a unique advantage for geospatial applications. TensorFlow's extensive ecosystem and its adaptability for various computational environments are also examined.*

*To ensure a balanced perspective, the article includes a literature review of seminal studies and papers that have shaped the domain of satellite imagery analysis. This includes an overview of groundbreaking works and recent advancements, providing readers with a contextual understanding of the field's evolution.*

*In conclusion, the article not only serves as a guide to current methodologies and tools in satellite image analysis but also as a window into the potential future developments in this rapidly evolving field.*

**Keywords:** machine learning, image classification, satellite imagery, deep learning, convolutional neural networks, PyTorch, TensorFlow, GIS.

*Матеріал надійшов 15.11.2023*



Creative Commons Attribution 4.0 International License (CC BY 4.0)



Давиденко А. М., Глибовець М. М.

## ВЕББАЗОВАНА СИСТЕМА ГРУПОВОГО РОЗПОДІЛЕНОГО РОЗРОБЛЕННЯ ПРОГРАМ

У статті описано архітектуру, використані технології створеної веббазованої системи розподіленого розроблення програм з акцентом на ефективне й просте одночасне групове створення тексту програми та її редагування і відлагодження. Впровадження модульної архітектури та механізму збереження вмісту сесій редагування забезпечило злагоджену функціональність компонент системи та ефективну підтримку узгодженості, управління неблокуючим паралелізмом, а також підтримку застосування сторонніх компіляторів та їх інтеграції з вебсистемою для колаборативного редагування.

Для зручності спілкування користувачів у системі реалізовано чат. Вагомим доробком є підтримка окремих сесій редагування. Також імплементовано механізм прототипування клієнтських вебзастосунків, що дає змогу розробникам вебзастосунків швидко перевірити коректність програмного коду або верстки вебсторінки і поділитися цим з іншими розробниками.

Коротко представлено сфери застосування системи і подальшого вдосконалення.

**Ключові слова:** операційні перетворення, розподілені системи, веброзроблення, програмування, колаборативні середовища.

### Вступ

Операційні перетворення (ОП) — це технологія для підтримки низки функціональних можливостей для спільної роботи в розподілених програмних системах. ОП було винайдено для підтримки узгодженості й управління неблокуючим паралелізмом під час розподіленого редагування (РР) текстових документів.

Уперше концепцію ОП розробили К. Елліс та С. Гіббс у системі GROVE (GRoup Outtie Viewing Edit) у 1989 р. [3]. Декілька років потому було виявлено деякі проблеми коректності, в результаті незалежно запропоновано кілька підходів щодо вирішення цих питань. Для розвитку взаємодії дослідників ОП і РР у 1998 р. було створено спеціальну групу за інтересами SIGCE (Special Interest Group of Collaborative Editing). Відтоді SIGCE проводить щорічні семінари в рамках основних конференцій CSCW (Computer Supported Cooperative Work).

Кожен учасник розподіленої системи з використанням операційних перетворень має підтримувати таку функціональність [4]:

- 1) генерування операцій: кожна дія користувача продукує операцію і має бути розіслана всім іншим учасникам;
- 2) прийом операцій: учасник повинен отримувати операції, що були згенеровані іншими учасниками;

- 3) виконання (застосування) операцій: учасник редагування повинен застосовувати отримані операції щодо локального контексту.

Колаборативні розподілені системи, що побудовані за ідеологією ОП, здебільшого використовують архітектуру, що базується на реплікації документів, які перебувають у сумісному розподіленому доступі. Це дає змогу досягти прийнятної швидкості відклику в середовищах із високою затримкою (таких, як Інтернет). Документи зі спільним доступом копіюються у локальне сховище кожного учасника РР. Таким чином, операції редагування можуть бути виконані локально відразу, а після цього поширені віддаленим учасникам. Віддалені операції редагування, що надходять учаснику, зазвичай спершу перетворюються і потім виконуються. Перетворення дає змогу гарантувати, що критеріїв узгодженості (які насамперед залежать від застосунку), буде досягнуто у всіх учасників РР. Оскільки ОП має неблокуючу властивість, локальний час відгуку є нечутливим до мережевих затримок. З урахуванням вищезазначених особливостей і переваг, ОП особливо підходить для реалізації колаборативних функцій, як-от групове редагування у Вебі та Інтернеті, що і було реалізовано у нашій системі.

## 1. Опис веббазованої системи для розподіленого розроблення програм

Основну ідею операційних перетворень можна проілюструвати на прикладі сценарію РР тексту. Дано текстовий документ з рядком тексту "abc", який репліковано у двох учасників редагування, а також дві паралельні операції:

1.  $O_1 = Insert [0, "x"]$  (вставити символ "x" на позицію "0")
2.  $O_2 = Delete [2, "c"]$  (видалити символ "c" з позиції "2").

Ці операції згенеровано двома учасниками РР (рис. 1).

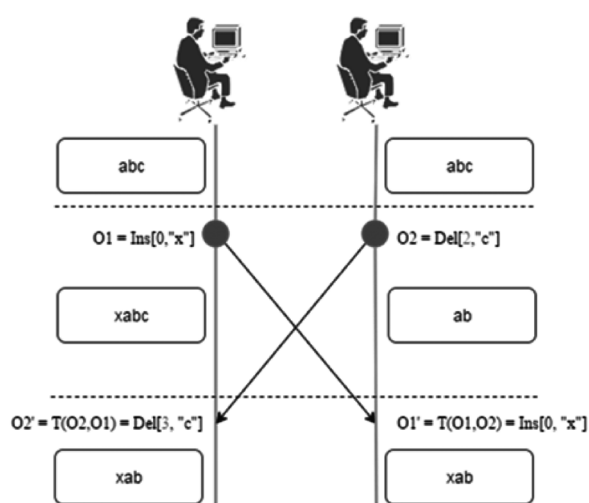


Рис. 1. Головний принцип операційного перетворення

Припустимо, що ці дві операції виконуються у такому порядку: спочатку  $O_1$ , потім  $O_2$ . Після виконання  $O_1$  рядок у документі міститиме значення "xabc". Щоб виконати  $O_2$  після  $O_1$ ,  $O_2$  має бути перетворено відносно  $O_1$  і відповідно операція матиме такий вигляд:  $O_2' = Delete [3, "c"]$ . Як можна побачити, перший параметр операції *Delete* було інкрементовано на одиницю у зв'язку зі вставкою символу "x" операцією  $O_1$ . Виконання  $O_2'$  на рядку "xabc" має видалити коректний символ "c", і в результаті значення стане "xab". Варто зазначити, що у більшості систем для визначення порядку операцій використовується механізм векторного годинника [5].

У разі виконання операції  $O_2$  без перетворення буде некоректно видалено символ "b" замість символу "c". Основною ідеєю ОП є перетворення (або налаштування) параметрів операцій редагування відповідно до ефектів попередньо виконаних паралельних операцій так, щоб перетворена операція досягала бажаного ефекту і зберігала узгодженість документа.

Розглянемо реалізацію поширеної функції будь-якого текстового редактора — Undo (відміна попередньої дії користувача). Маємо двох розподілених учасників редагування — Site 1 та Site 2. Також дано текстовий документ із рядком тексту "12" (рис. 2).

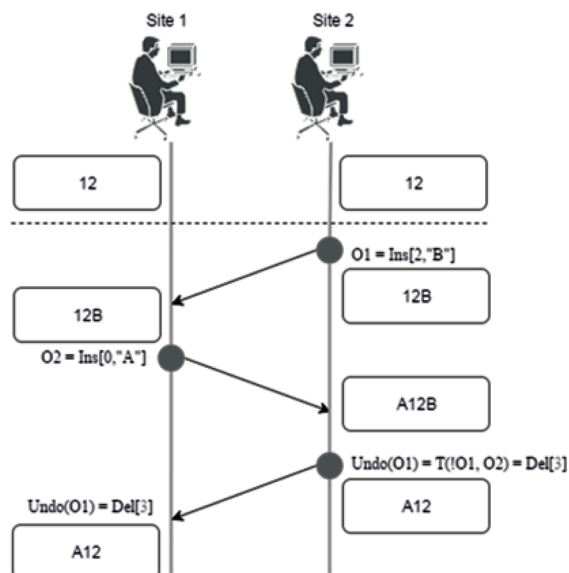


Рис. 2. Візуалізація відміни останньої дії (Undo)

Спершу користувач з Site 2 виконує вставку символу "B" на позицію 2, генеруючи наступну операцію:  $O_1 = Insert [2, "B"]$ . Ця операція також поширюється на Site 1 і застосовується без перетворень. У результаті маємо текст "12B" у обох учасників РР.

Далі користувач з Site 1 виконує вставку символу "A" на позицію 0, генеруючи наступну операцію:  $O_2 = Insert [0, "A"]$ . Ця операція також поширюється на Site 2 і застосовується без перетворень. У результаті маємо текст "A12B" у обох учасників РР.

Після виконання  $O_1$  і  $O_2$  користувач на Site 2 виконує відміну своєї попередньої дії, генеруючи команду *Undo* ( $O_1$ ) яка власне не є останньою виконаною операцією. Система, що базується на підході ОП, спершу створить наступну інвертовану операцію:

$$!O_1' = Inverse (O_1 = Insert [2, "B"]) = Delete [2].$$

Потім буде трансформовано  $!O_1'$  проти  $O_2$ :  $!O_1' = T(!O_1', O_2) = Delete [3]$ . Врешті-решт, після застосування  $!O_1'$ , буде отримано рядок "A12", що відповідає очікуваному ефекту операції Undo.

Підсумовуючи, зазначимо, що основною ідеєю операційних перетворень для операції відміни останньої дії користувача є інвертування операції  $O$  з урахуванням операцій, що були виконані після  $O$  задля досягнення ефекту відміни

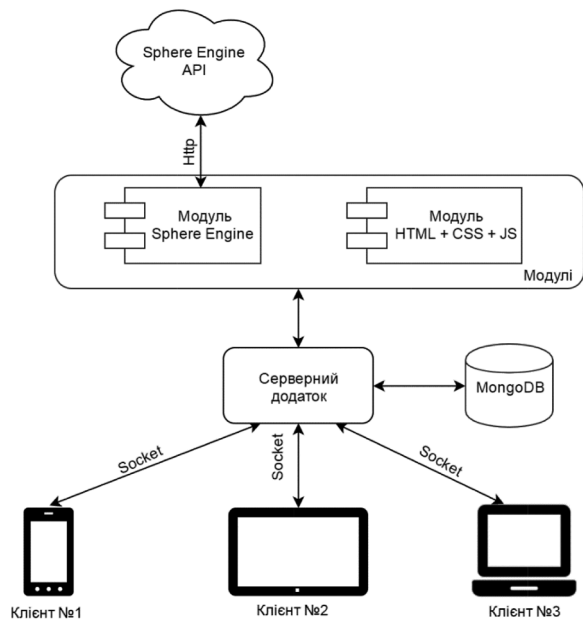


Рис. 3. Компонентна архітектура веббазованої системи для розподіленого розроблення програм

(Undo effect). Коректний ефект відміни досягається виключенням ефекту операції *O*, проте зі збереженням ефектів інших операцій.

## 2. Опис веббазованої системи для розподіленого розроблення програм

Порівняно з попередньо реалізованим прототипом [1], представлена тут система РР отримала нову та покращену функціональність. Також у зв'язку зі впровадженням модульної архітекту-

ри та механізму збереження вмісту кімнат було оновлено компонентну архітектуру застосунку (рис. 3).

Насамперед, вагомим доробком є підтримка окремих сесій редагування. Після відкриття головної сторінки вебзастосунку користувача буде перенаправлено на сторінку зі згенерованим випадковим чином префіксом. Цей префікс і відповідатиме id кімнати, у якій перебуває користувач. Одночасно до кімнати може бути підключено багато учасників РР. Для того щоб запросити до кімнати іншого учасника, достатньо скопіювати поточну адресу в браузері та поділитися нею, що робить процес РР найбільш простим та інтуїтивним. Після завантаження сторінки необхідно ввести ім'я користувача і натиснути кнопку "Join". Після входу інтерфейс системи має такий вигляд (рис. 4).

Для збереження внесених змін програмного коду та можливості відновлення втрачених даних у разі технічного збою сервера було використано нереляційну базу даних MongoDB [6]. Окрім збереження самого документа, що редагується у колаборативному розподіленому середовищі, зберігається вся історія правок документа, що дасть змогу за необхідності відтворити зміни.

Для зручності користувачів у правій частині середовища розміщено чат, де користувачі можуть спілкуватися між собою у поточній кімнаті.

Також було імплементовано механізм для прототипування клієнтських вебзастосунків.

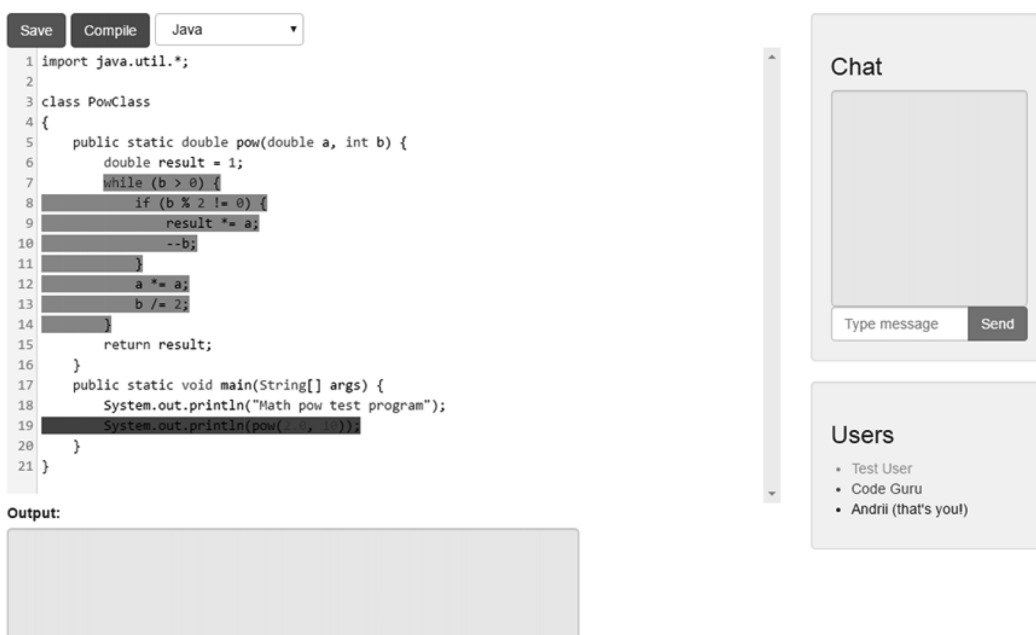


Рис. 4. Інтерфейс веббазованої системи для розподіленого розроблення програм

Часто у розробників вебзастосунків виникає ситуація, коли необхідно швидко перевірити коректність програмного коду або верстки вебсторінки і поділитися цим з іншими розробниками. Саме для таких випадків новий механізм стане у пригоді. Після вибору режиму “HTML+CSS+JS” перед учасниками редагування з’являється чотири секції: HTML, JavaScript, CSS та Result (результат компіляції).

Структура скомпільованого вебзастосунку матиме такий вигляд:

```
<html>
  <head>
    <style>
      <!-- Вміст секції CSS -->
    </style>
  </head>
  <body>
    <!-- Вміст секції HTML -->
    <script>
      <!-- Вміст секції
              JavaScript -->
    </script>
  </body>
</html>
```

Після компіляції вебзастосунку скомпоновану сторінку буде відображено в розділі Result, а JavaScript скрипти буде виконано (рис. 5).

В оновленій версії системи RP було розроблено гнучку модульну архітектуру для підтримки використання сторонніх компіляторів та їх інтеграції з вебсистемою для колаборативного редагування.

За програмним контрактом кожен створений адаптер компілятора повинен мати такі методи:

- `beforeCompile(obj)` — викликається перед виконанням компіляції та призначений для попереднього оброблення даних (наприклад, видалення спецсимволів, попередня валідація тощо). Як вхідний параметр приймає масив документів колаборативного редагування;
- `compile(obj)` — виконує компіляцію програмного коду. Як вхідний параметр приймає результат методу `beforeCompile`. У середині цього методу має бути здійснено виклик до стороннього компілятора або ж проведено компіляцію «на місці». Цей метод має бути асинхронним і повертати об’єкт `promise` одразу, а після виконання компіляції — результат компіляції;
- `processResponse(obj)` — готує результат компіляції до відображення у вікні виведення графічного інтерфейсу користувача. Як вхідний параметр приймає результат методу `compile`. Повертає рядок символів.

Конструктор компілятора може приймати необхідну кількість параметрів (наприклад, токен для компіляції тощо).

Для того щоб створити власний модуль адаптера до компілятора, необхідно виконати такі кроки:

1. Створити файл із розширенням `*.js`, що має міститися у папці `compilers` на тому самому рівні, що й запущений вебзастосунок.

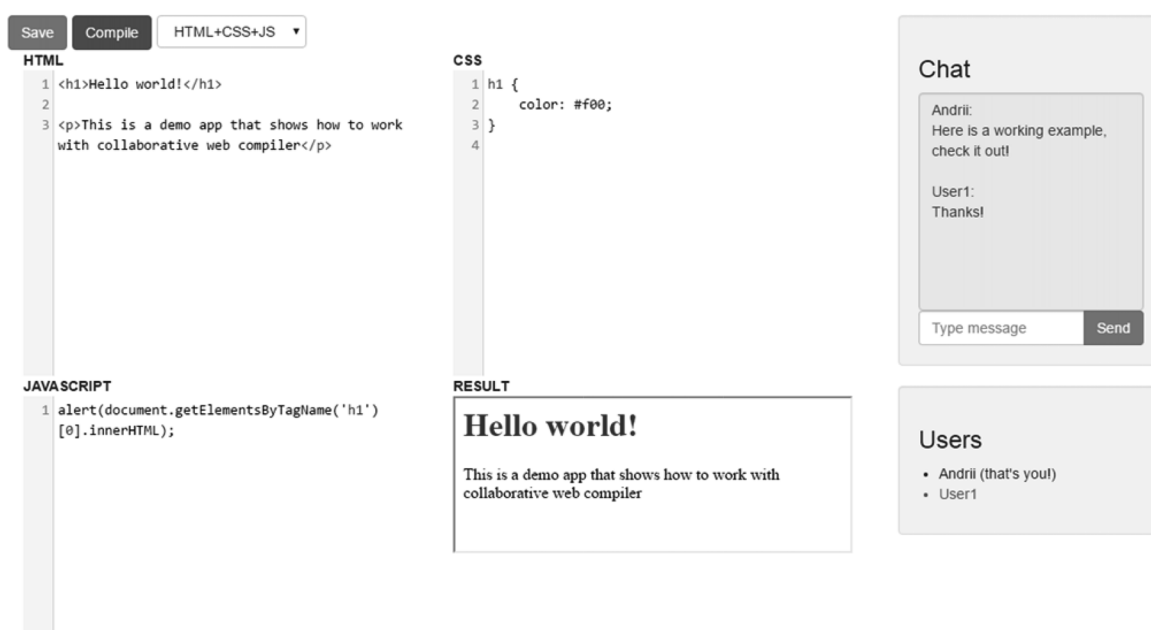


Рис. 5. Режим “HTML+CSS+JS” у веббазованій системі для розподіленого розроблення програм

2. Створити модуль адаптера до компілятора. У загальному випадку модуль має таку структуру:

```
var q = require('q');

function compilerModule(param1,
param2) {
  this.output = «compose»;
  this.compile = function(obj) {
    var deferred = q.defer();
    deferred.resolve(obj);
    return deferred.promise;
  },

  this.processResponse =
    function(response) {
      return response;
    },

  this.beforeCompile =
    function(obj) {
      return obj;
    }
};
module.exports = function(module_
holder) {
  module_holder['compiler_module']
= compilerModule;
};
```

3. Зареєструвати компілятор у конфігураційному файлі. Для цього необхідно відкрити конфігураційний файл `config.js` та у масив `config.compilers` додати запис про новостворений адаптер у такому форматі:

```
{
  id: 'compiler_module', // назва
                        компілятора, що була
                        // вказана
                        у експорті модуля
  params: ['param', 40] // масив
                        параметрів для
                        //ініціалі-
                        зації об'єкту компілятора
}
```

4. У конфігураційному файлі у масиві `config.langs` додати або змінити компілятор, що використовується для певної мови (поле `compilerId`), наприклад:

```
{
id: 3,
name: «Java»,
compilerId: «compiler_name»,
style: «text/x-java»,
nested: «false»,
view: «partial-single-document.
html»,
```

```
placeholder: «code-area»,
output: «compile-output»,
visible: «true»
}
```

Опис значень параметрів конфігурації:

- **id** — унікальний ідентифікатор мови;
- **name** — назва мови (режиму), яка виводиться у списку графічного інтерфейсу користувача;
- **compilerId** — назва компілятора, що використовується для цієї мови;
- **style** — стиль текстового редактора CodeMirror для стилізації програмного коду;
- **nested** — позначає, чи підтримує ця мова (режим) багатодокументне редагування. Якщо значення цього параметра дорівнює `true`, то необхідно вказати ідентифікатори мов у масиві `inner`;
- **view** — назва `html`-документа, який буде відображено для поточної мови. Має містити секції для введення коду і для виведення результату;
- **placeholder** — ідентифікатор `div` для розміщення коду;
- **output** — ідентифікатор `div` для розміщення результату;
- **visible** — чи показувати цю мову у списку мов на клієнті.

5. Перезапустити серверний застосунок. Під час запуску сервер динамічно підвантажить усі наявні адаптери з папки `compilers`.

Така архітектура є досить гнучкою і дає змогу використовувати різноманітні сторонні компілятори. За необхідності нові адаптери до компіляторів можна додавати без правки коду серверного застосунку, що є якісною ознакою гарної архітектури вебсистеми.

## Висновки

У рамках цієї роботи представлено створену веббазовану систему, що базується на операційному підході до розподіленого редагування, а також значно розширено гнучкість модульної архітектури та функціональність попередньо розробленого прототипу [1].

У статті описано архітектуру, використані технології створеної веббазованої системи розподіленого розроблення програм з акцентом на ефективне й просте одночасне групове створення тексту програми та її редагування і відлагодження. Впровадження модульної архітектури та механізму збереження вмісту забезпечило злагоджену функціональність компонент систе-

ми й ефективну підтримку узгодженості, управління неблокуючим паралелізмом. Оскільки система має неблокуючу властивість, локальний час відгуку є нечутливим до мережових затримок. Для збереження внесених змін програмного коду та можливості відновлення втрачених даних у випадку технічного збою сервера використано нереляційну базу даних MongoDB.

Важливою особливістю системи є використання гнучкої модульної архітектури для підтримки ефективного застосування сторонніх компіляторів та їх інтеграції з вебсистемою за колаборативного редагування.

Для зручності спілкування користувачів у системі реалізовано чат. Вагомим доробком є підтримка окремих сесій редагування. Також імплементовано механізм прототипування клієнтських вебзастосунків, що дає змогу розроб-

никам вебзастосунків швидко перевірити коректність програмного коду чи верстки вебсторінок і поділитися цим з іншими розробниками. Саме для таких випадків новий механізм стане у пригоді.

Прикладом застосування розробленої системи може бути використання у електронному навчанні. З огляду на стрімке поширення засобів електронного навчання у вищій школі [2], а також дистанційний характер навчання протягом останніх років, запропонована система може сприяти покращенню рівня співпраці учасників навчального процесу.

Подальшим розвитком цієї роботи може бути дослідження неконфліктуючих реплікованих типів даних (CRDT) як альтернативи операційним перетворенням для забезпечення синхронізації даних за колаборативного редагування.

#### Список літератури

1. Бублик В. В. Колаборативні методи в електронному навчанні програмування / В. В. Бублик, А. М. Давиденко // Наукові записки НаУКМА. — 2016. — Т. 190. Комп'ютерні науки. — С. 41–45.
2. Глибовець М. М. Застосування Semantic WEB до створення колаборативного освітнього простору / М. М. Глибовець // Збірник праць П'ятої Міжнародної конференції «Нові інформаційні технології для всіх». — Київ : Академперіодика, 2010. — С. 179–192.
3. Ellis C. A. Concurrency control in groupware systems / C. A. Ellis, S. J. Gibbs // Proceedings of the 1989 ACM SIGMOD international conference on Management of data / C. A. Ellis, S. J. Gibbs. — New York, NY, USA : ACM, 1989. — Pp. 399–407.
4. Copies Convergence in a Distributed Realtime Collaborative Environment / N. Vidot, M. Cart, J. Ferrie, M. Suleiman // Proceedings of the ACM Conference on Computer-Supported Cooperative Work / N. Vidot, M. Cart, J. Ferrie, M. Suleiman., 2000. — Pp. 171–180.
5. Lamport L. Time, Clocks, and the Ordering of Events in a Distributed System / L. Lamport // Communications of the ACM 21. — 1978. — No. 21. — Pp. 558–565.
6. MongoDB Manual [Electronic resource]. — Mode of access: <https://docs.mongodb.com/manual/>.

#### References

- Bublyk, V. V., & Davydenko, A. M. (2016). Kolaboratyvni metody v elektronnomu navchanni prohramuvannia. *Naukovi zapysky NaUKMA. Kompiuterni nauky*, 190, 41–45 [in Ukrainian].
- Hlybovets, M. M. (2010). Zastosuvannia Semantic WEB do stvorennia kolaboratyvnoho osvithnoho prostoru. In *Zbirnyk Prats Piatoi Mizhnarodnoi Konferentsii "Novi Informatsiini Tekhnolohii Dlia Vsikh"* (pp. 179–192). Akademperiodyka [in Ukrainian].
- Ellis, C. A., & Gibbs, S. (1989). Concurrency control in groupware systems. *Sigmod Record*, 18 (2), 399–407. <https://doi.org/10.1145/66926.66963>.
- Vidot, N., Cart, M., Ferrie, J., & Suleiman, M. (1999). Copies Convergence in a Distributed Realtime Collaborative Environment. *Proc. ACM Conf. Computer-Supported Cooperative Work (CSCW '00)*, 171–180.
- Lamport, L. (1978). Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21 (7), 558–565. <https://doi.org/10.1145/359545.359563>.
- What is MongoDB?* — *MongoDB Manual*. (n.d.). <https://docs.mongodb.com/manual/>.

A. Davydenko, M. Glybovets

## WEB-BASED SYSTEM FOR DISTRIBUTED GROUPWARE SOFTWARE DEVELOPMENT

*The article describes the architecture and technologies used to create a web-based distributed software development system with an emphasis on efficient and simple simultaneous grouped creation of program text, its editing, and debugging. The introduction of a modular architecture and a content storage mechanism ensured the coordinated functionality of the system components and effective support for consistency and non-blocking parallelism management. Since the system has a non-blocking property,*

*the local response time is insensitive to network delays. The non-relational MongoDB database is used to save the changes made to the program code and to recover lost data in the event of a technical server failure.*

*An important feature of the system is the use of a flexible modular architecture to support the effective use of third-party compilers and their integration with the web-based system for collaborative editing.*

*To facilitate user communication, the system has a chat feature. A significant improvement is the support of separate editing sessions. A mechanism for prototyping client web applications has also been implemented, which allows web application developers to quickly check the correctness of the program code or web page layout and share it with other developers. It is for such cases that the new mechanism will come in handy.*

*The areas of its application and further improvement are briefly presented. Further development of the system may include the study of the implementation of nonconflicting replicated data types (CRDTs) as an alternative to operational transformations to ensure data synchronization during collaborative editing.*

**Keywords:** operational transformations, distributed systems, groupware, web-development, programming, collaborative environments

*Матеріал надійшов 27.11.2023*



Creative Commons Attribution 4.0 International License (CC BY 4.0)

Глибовець А. М., Хмель С. М., Печкурова О. М.

## ВИКОРИСТАННЯ НЕЙРОКОМП'ЮТЕРНИХ ІНТЕРФЕЙСІВ ДЛЯ ЗБИРАННЯ ТА НАКОПИЧЕННЯ ІНФОРМАЦІЇ ПРО КОРИСТУВАЧІВ ВЕБРЕСУРСІВ

У статті описано дослідження можливостей використання нейрокомп'ютерних інтерфейсів (НІ) для збирання та накопичення інформації про користувачів вебресурсів шляхом проведення експериментів на електроенцефалографічному приладі Emotiv INSIGHT на 5 каналів і браузером Chrome, проаналізовано стан сучасних НІ та їх застосування, представлено розроблену методику збирання даних за допомогою НІ та її реалізацію у вигляді програмного застосунку. Розроблений прототип дає змогу керувати сторінками Chrome за допомогою ментальних команд і збирати дані про досвід користування вебресурсами, що доповнені даними про емоційний і ментальний стан користувача.

**Ключові слова:** нейрокомп'ютерні інтерфейси, ВСІ, збирання і зберігання інформації про користувачів вебресурсів, взаємодія з вебресурсом, безпека.

### Вступ

Використання інтернету та вебресурсів у наші дні набуває все більшої популярності, а отже, розуміння потреб користувачів, збільшення їхньої зацікавленості та покращення досвіду користувача та безпеки користування вебресурсами є важливим завданням для розробників і бізнесу. Відстеження та аналіз цієї інформації може бути важливим інструментом для покращення якості та ефективності вебресурсів.

Одним із потенційних методів для збирання та аналізу даних є використання нейрокомп'ютерних інтерфейсів (НІ, англ. brain-computer interface, ВСІ), які здатні перетворювати сигнали мозку на корисну інформацію. Використання НІ може допомогти збирати дані про користувачів вебресурсів, як-от зацікавленість, ставлення до певних продуктів і послуг, а також емоційний стан.

Розроблений прототип допомагає розширити можливості взаємодії користувача з вебресурсами, додає можливість керування роботою браузера за допомогою ментальних команд та імпульсів, що передає ВСІ-прилад. Інформація, що збирається упродовж користування вебресурсами за наявності ВСІ-приладу, дасть змогу краще зрозуміти поведінку та потреби користувача і в подальшому покращити взаємодію з вебресурсом та безпеку користування.

Нейрокомп'ютерний інтерфейс — це зв'язок між електричною активністю мозку і комп'ютером або іншими зовнішніми пристроями.

Сигнали, отримані від мозку, можуть мати електрофізіологічний, магнітний або метаболічний характер. Перетворення цих сигналів на команди, якщо враховано різні потенціали, дає можливість керувати приладами та взаємодіяти з іншими програмними інтерфейсами [7]. Загальну схему інтерфейсу мозок-комп'ютер (brain-computer) подано на рис. 1.

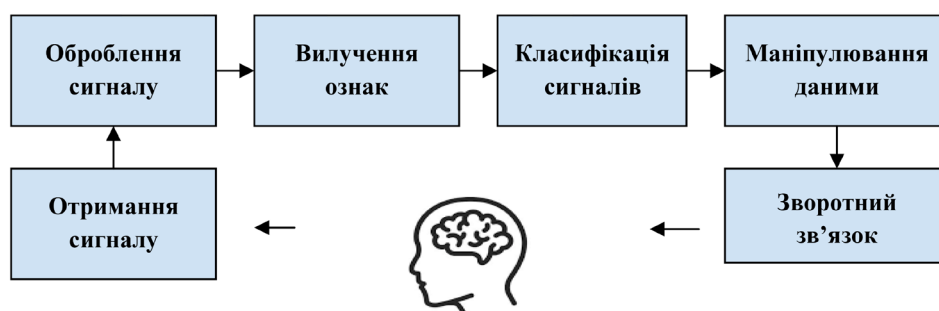


Рис. 1. Загальна схема інтерфейсу мозок-комп'ютер (brain-computer)



У інвазивного ВСІ отримання сигналів мозку реалізується за допомогою датчиків, що імплантовані безпосередньо в сіру речовину мозку шляхом нейрохірургії. Цей метод дає змогу отримати найбільш якісний і точніший сигнал, що містить менше шумів. Однак він більш ризикований, тому дослідження інвазивного ВСІ проводили тільки з тваринами [7]. За частково інвазивного ВСІ електроди імплантують усередину черепа, але за межами мозку. Сила сигналу нижча, однак ризик для пацієнта також менший.

Неінвазивний ВСІ не потребує внутрішньочерепної хірургії та імплантації пристрою в мозок. Це найбезпечніший спосіб, однак сила сигналу в такому разі нижча за інші способи. Неінвазивний ВСІ використовує функціональну магнітно-резонансну томографію (МРТ), позитронно-електронну томографію (ПЕТ), функціональну спектроскопію ближнього інфрачервоного діапазону (fNIRS), магнітоенцефалографію (МЕГ), електроенцефалографію (ЕЕГ) та однофотонну емісійну комп'ютерну томографію (SPECT) для захоплення сигналів мозку.

Ми будемо розглядати ЕЕГ-прилади, оскільки їх вважають найбільш перспективними. Сигнали ЕЕГ легко фіксувати й аналізувати. Прилади на основі ЕЕГ порівняно дешеві, їх використання не передбачає хірургічного втручання, що є важливим фактором. ЕЕГ реєструє коливання напруги внаслідок потоку іонного струму під час синаптичних збуджень у нейронах мозку. За цього способу електроди прикріплюють до шкіри голови для отримання сигналів мозку. Номер електрода варіюється від 1 до 256 для різних ЕЕГ-гарнітур. Вимірний сигнал ЕЕГ — це різниця напруги між активним електродом і електродом порівняння протягом часу з амплітудою в мікрвольтах (мкВ). Зазвичай амплітуда ЕЕГ коливається від -100 до +100 мікрвольт. Сигнали ЕЕГ можна класифікувати (табл. 1) відповідно до діапазонів частот, і кожен із цих діапазонів має певне біологічне значення.

Є два основні способи отримання сигналів ЕЕГ: бездротовий і дротовий. У нашому дослі-

дженні використано бездротовий Emotiv INSIGHT на 5 каналів. Цей прилад має напівсухі полімерні датчики й датчики руху, що дає змогу виявляти рухи голови [1].

Виявлення базуються на кількох різних експериментальних наборах даних, зібраних від добровольців під час власних експериментів Emotiv [15]. Наприклад, виявлення фрустрації було проведено з більш ніж 30 суб'єктами змішаної статі, які носили гарнітури Emotiv, ЕКГ, респіратор, GSR і сфігмоманометр під час певної діяльності. Їх супроводжував психолог, вони знімали на відео та повідомляли дані про себе до та після події, а базові показники всього обладнання було зроблено також до та після.

Експерименти склалися з серії онлайн-ігор, які мали викликати розчарування, наприклад, гра в стилі PacMan, де елементи керування випадково виходили з ладу або змінювалися щораз частіше, або гра FPS, де суб'єкт поступово переважувався, а його обладнання було ненадійним.

Крім того, деякі з виявлень не покладаються на попередні дані, вони збирають дані з тегами й тренують нові сигнатури на основі даних ЕЕГ кожної особи (так працює набір розумових команд).

### Розроблення архітектури системи взаємодії з браузером та збирання інформації користувача

Система містить кілька компонент, які працюють разом, щоб забезпечити збирання, оброблення і відображення даних, зібраних з ЕЕГ-приладу (рис. 2).

Взаємодію компонент забезпечує Chrome-застосунок (рис. 3).

Розширення збирає дані з ЕЕГ-приладу й надсилає їх на сервер. Для написання розширення використовують JavaScript, для взаємодії з приладом — Cortex API, який надала компанія — виробник приладу Emotiv, для відправки даних на сервер застосовують REST. Важливо,

Таблиця 1

Смуги частот ЕЕГ із властивостями

	Частота (Hz)	Амплітуда (mV)	Розміщення	Активність
Delta	0,5–4 Hz	100–200	Фронтальна зона голови	Глибокий сон
Theta	4–8 Hz	5–10	Різні ділянки	Сонливість, легкий сон
Alpha	8–13 Hz	20–80	Задня ділянка голови	Розслаблений
Beta	13–30 Hz	1–5	Ліва та права частина, симетричний розподіл, найбільш виражений фронтально	Активне мислення, пильність
Gamma	> 30Hz	0,5–2	Соматосенсорна кора	Гіперактивність

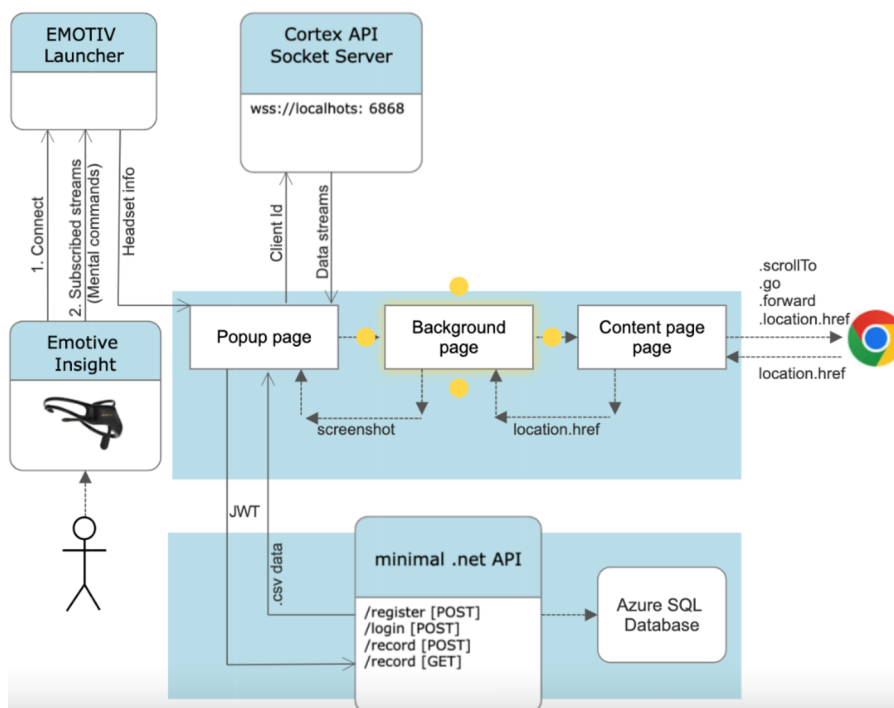


Рис. 2. Загальне архітектурне рішення

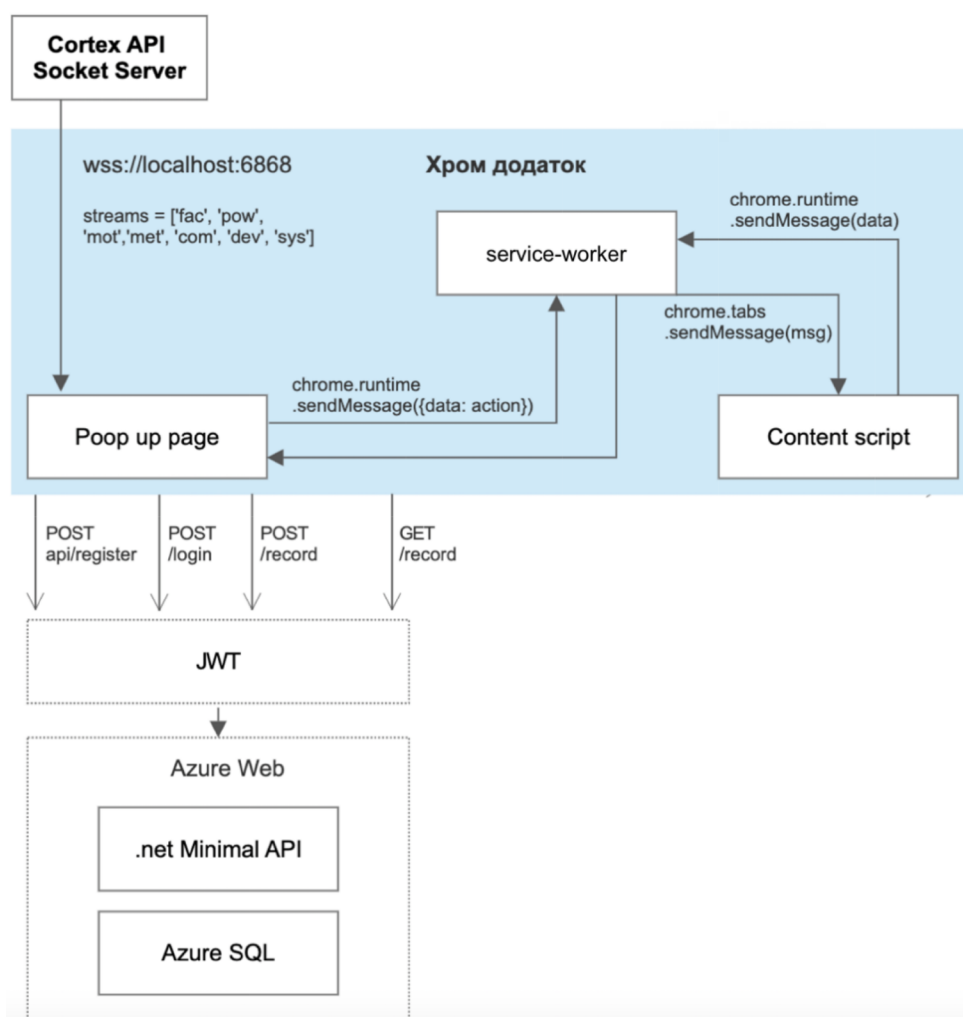


Рис. 3. Взаємодія між компонентами

щоб розширення було легким і не уповільнювало роботу браузера, тому для прототипу не використовували фреймворки, хоча в майбутньому можливо було б задіяти React для пришвидшення рендерингу.

Для стилізації вжито Bootstrap, але вибір бібліотеки не є принциповим. Графічну візуалізацію показників виконує Apexcharts, але це також не є принциповим рішенням. Розширення зібрано з використанням Webpack, а побудовано за принципами Chrome Manifest v3 [8].

Сервер приймає дані від розширення та зберігає їх у базі даних. Для оброблення даних та аналізу результатів можна використовувати спеціалізовані бібліотеки для роботи з ЕЕГ-даними. Для забезпечення безпеки та конфіденційності даних сервер має відповідні заходи захисту, як-от шифрування та автентифікація.

### Розроблення API з використанням minimal.NET API для зберігання та обміну даними

Для зберігання отриманих даних було розроблено API з використанням minimal .NET API [11]. Обмін даними використовує CRUD, авторизація — JWT.

Рівень API обробляє вхідні HTTPS-запити та надсилає HTTPS-відповіді. Він складається з кінцевих точок для різних методів HTTP, таких як GET, POST, PUT і DELETE. Ці контролери взаємодіють із сервісним рівнем для виконання операцій CRUD над даними.

Сервісний рівень містить бізнес-логіку програми. Він відповідає за реалізацію шаблону репозиторію для взаємодії з базою даних і виконання операцій CRUD, а також реалізує авторизацію JWT для перевірки запитів користувачів.

Рівень сховища взаємодіє з базою даних для виконання операцій CRUD. Він містить реалізацію DTO для передавання даних між різними рівнями програми.

Рівень бази даних містить базу даних SQL, яка зберігає дані програми.

Структуру проєкту відображено на рис. 4.

У цій архітектурі елементи AuthEndpoints і RecordEndpoints визначають кінцеві точки для різних методів HTTP. Клас RecordEndpoints викликає інтерфейс IRecordRepository, який реалізований класом RecordRepository. Клас AuthEndpoints викликає інтерфейс IAuthRepository, який реалізований класом AuthRepository. Класи RecordRepository та AuthRepository взаємодіють із базою даних SQL для виконання операцій CRUD над моделями Record і LocalUser відповідно.

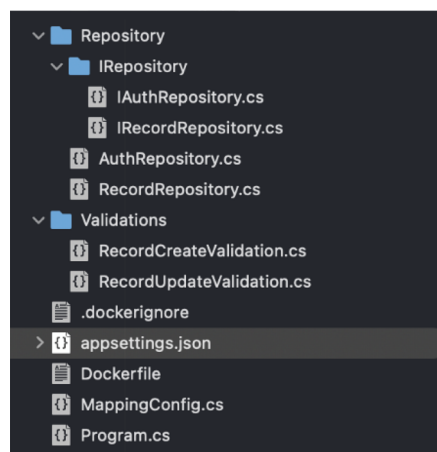
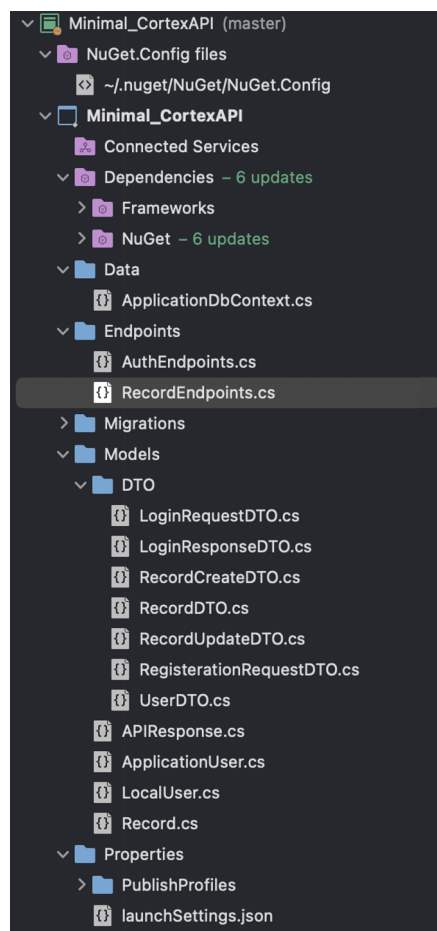


Рис. 4. Структура проєкту

Для аутентифікації використовується Microsoft.AspNetCore.Authentication.AuthenticationMiddleware. Авторизацію виконує Microsoft.AspNetCore.Authentication.AuthorizationMiddleware.

UseRouting налаштовує маршрутизацію для оброблення вхідних запитів. Потім UseAuthorization авторизує запит на основі схем автентифікації та політик, налаштованих у програмі. UseEndpoints відображає кінцеві точки на відповідні запити.

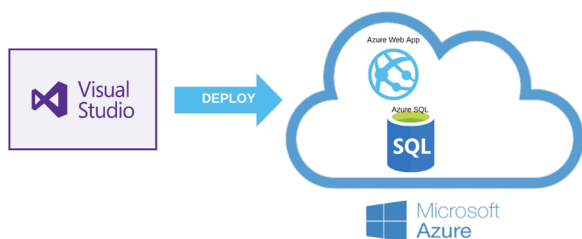


Рис. 5. Публікація застосунку

Для JWT для автентифікації користувача використовується `Microsoft.AspNetCore.Authentication.JwtBearer`. Це пакет у структурі ASP.NET Core, який забезпечує перевірку вебтокенів JSON (JWT) у вхідних запитах. Керування автентифікацією та авторизацією користувачів забезпечує `Microsoft.AspNetCore.Identity`.

Файл `appsettings.json` містить налаштування конфігурації програми, наприклад рядок підключення до бази даних SQL. Файл `Program.cs` відповідає за налаштування та запуск програми .NET minimal API.

## Публікація та взаємодія з Azure

Публікація (рис. 5) наразі відбувається за використання Visual Studio, але в подальшому розвитку проекту можливо було б налаштувати CI/CD іншим чином.

## Архітектура бази даних

Як базу даних було вибрано SQL Azure database. На рис. 6 показано зв'язок між користувачем і створеними ним записами.

Із цією структурою бази даних можна використовувати Entity Framework або іншу ORM, щоб зіставляти моделі з таблицями бази даних і взаємодіяти з ними у своїй програмі.

## Реалізація прототипу збирання та накопичення інформації

Для збирання даних користувача було розроблено Chrome-застосунок, який використовує Cortex API для взаємодії з гарнітурою і в якому

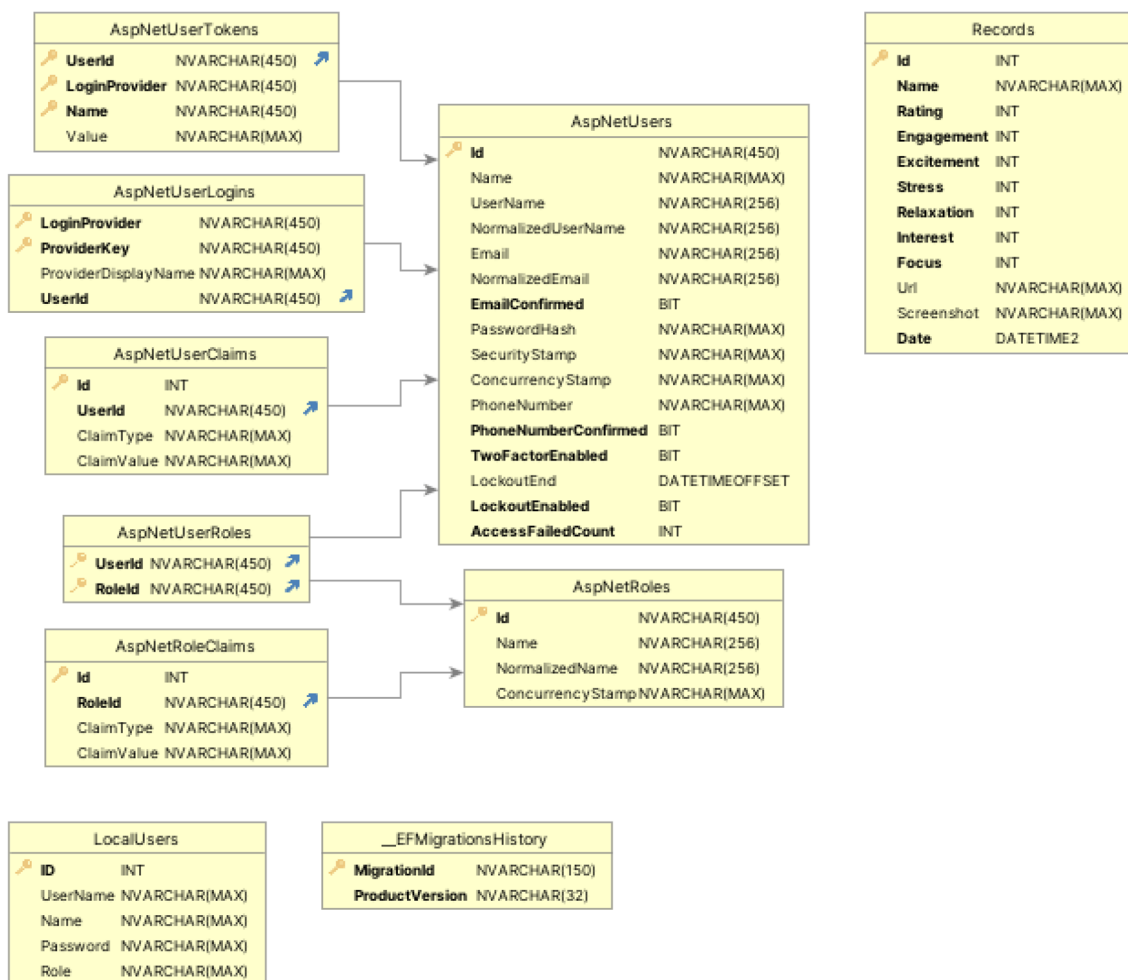


Рис. 6. Структура бази даних

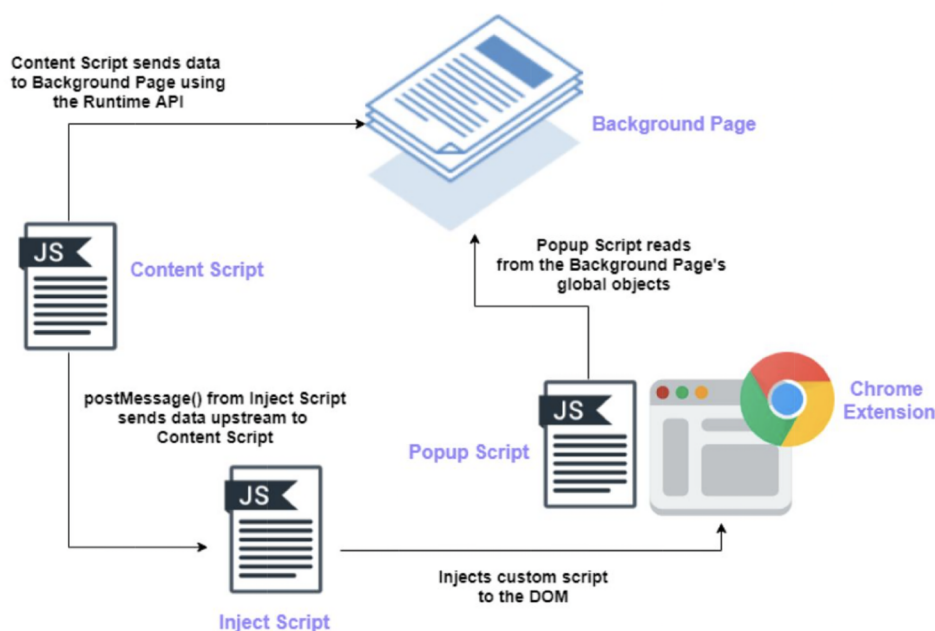


Рис. 7. Передавання повідомлень у Chrome-застосунку

реалізовано робочий процес API, створено підписку на отримання даних і графічний інтерфейс для відображення отриманих даних, побудовано інтерфейс для реєстрації та входу користувача (дані з гарнітури отримуються за допомогою Websocket і відправляються на сервер за допомогою REST API). Схему передавання повідомлень у застосунку відображено на рис. 7.

Взаємодія з браузером відбувається шляхом передавання команди у content-script, якій має безпосередній доступ до сторінки браузера. Тобто відбувається JavaScript inject, з можливостями, що надає JavaScript [8].

У прототипі було використано такі команди (але в подальшому потрібно провести аналіз найбільш потрібних користувачеві дій):

```
chrome.runtime.onMessage.addListener((msg, sender, sendResponse) => {
  console.log("content got message: ", msg);
  if (msg.data === 'scrollUp') {
    window.scrollTo({ top: document.documentElement.scrollTop -100,
behavior: 'smooth' })
  }
  if (msg.data === 'scrollDown') {
    window.scrollTo({ top: document.documentElement.scrollTop +100,
behavior: 'smooth' })
  }
  if (msg.data === 'scrollToTop') {
    window.scrollTo({ top: 0, behavior: 'smooth' })
  }
  if (msg.data === 'scrollToBottom') {
    window.scrollTo(0, document.body.scrollHeight);
  }
  if (msg.data === 'goBack') {
    window.history.go(-1);
  }
  if (msg.data === 'goForward') {
    window.history.forward();
  }
});
```

Взаємодія з гарнітурою і збирання даних користувача відбувається у рорир сторінці. Таке рішення було пов'язане з частою втратою сигналу з гарнітури, тож у рорир сторінці цю проблему простіше відстежити й усунути.

Для взаємодії з гарнітурою використовується Cortex API, тож коли надходить сигнал із гарнітури, з рорир відправляється запит до service worker. Якщо в момент запиту один з Chrome tab є активним, то service worker надсилає повідомлення до рорир page з такими даними: Screenshot сторінки, URL сторінки. Ці дані було вибрано, щоб продемонструвати можливості взаємодії додатку з сторінкою браузера. В подальшому розвитку проекту можливо передавати інші дані, за переліком — усі, які можна отримати з сторінки браузера за допомогою JavaScript.

Зібрані дані надсилаються до сервера для збереження у базі даних. Приклад зібраних даних наведено на рис. 8. Зібрані дані можна завантажити у вигляді .csv файлу.

## Висновки

У цьому дослідженні проаналізовано можливості взаємодії між VCI-приладом і браузером Chrome. Розроблений прототип дає змогу керувати сторінками Chrome за допомогою ментальних команд і збирати дані про досвід користу-

вання вебресурсами, що доповнені даними про емоційний та ментальний стан користувача.

Розроблений прототип можна використовувати для покращення навчального процесу або маркетингових досліджень, тобто він передбачає модель B2B (business-to-business). Подальший напрям розвитку додатка передбачає підтримку найпоширеніших гарнітур VCI.

Серед недоліків застосунку зазначимо таке. Під час тестування не вдалося отримати 100 % з'єднання, максимальна отримана якість була 80 %. Відповідно користуватися ментальними командами було неможливо. Щоб усунути недолік, треба замінити прилад. Для отримання ЕЕГ-даних потрібна окрема ліцензія, а для отримання ментальних команд — окремий запит до служби підтримки. Користуватися застосунком можна тільки з clientId.

Проте збирання даних, які поєднують інформацію про досвід користувача, разом із даними про ментальні стани та управління за допомогою ментальних команд відкриває широкі перспективи і можливості. Для цього необхідно провести додаткові дослідження та вдосконалити прототип.

Загалом використання VCI є цікавим і перспективним напрямом, але існує ще багато питань, які можуть бути перешкодою для щоденного використання VCI користувачами, що не спеціалізуються в цій галузі.

Id	Name	Rating	Engagement	Excitement	Stress	Relaxation	Interest	Focus	Url	Screenshot	Date
9	user2	0	0,169666	0,144721	0,231963	0,11751	0,424111	0,146369			0001-4
10	user2	0	0,137832	0,167417	0,233406	0,132889	0,43631	0,07329			0001-4
11	user2	0	0	0	0	0	0	0	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
12	user2	0	0	0	0	0	0	0	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
13	user2	0	0,715838	0,205028	0,365784	0,421557	0,494936	0,516555	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
14	user2	0	0,469385	0,498245	0,493113	0,551166	0,550124	0,37729	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
15	user2	0	0,646979	0,713939	0,41293	0,434064	0,495224	0,5617	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
16	user2	0	0,630468	0,393141	0,38384	0,408921	0,460409	0,591261	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
17	user2	0	0,592997	0,235211	0,39806	0,412705	0,471805	0,467372	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
18	user2	0	0,504471	0,332451	0,300691	0,346026	0,540614	0,309841	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
19	user2	0	0,517188	0,397777	0,328276	0,33638	0,480305	0,294002	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
20	user2	0	0,500879	0,203085	0,274869	0,304545	0,468501	0,314883	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
21	user2	0	0,50077	0,417278	0,383585	0,452125	0,554597	0,309707	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
22	user2	0	0,457239	0,296883	0,332148	0,377111	0,50687	0,244384	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
23	user2	0	0,403107	0,302896	0,291229	0,313338	0,448016	0,202878	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4
24	user2	0	0,385461	0,302975	0,311744	0,311488	0,48506	0,248093	https://en.wikipedia.org/...	data: image/jpeg;base64,/9...	0001-4

Рис. 8. Приклад зібраних даних

## Список літератури

1. A Beginner's Guide to Brain-Computer Interface and Convolutional Neural Networks by Alexandre Gonfalonieri [Electronic resource]. — 2018. — Mode of access: <https://towardsdatascience.com/a-beginners-guide-to-brain-computer-interface-and-convolutional-neural-networks-9f35bd4af948>.
2. Are Emotiv Products Medical Devices? [Electronic resource] // Emotiv. — 2020. — Mode of access: <https://www.emotiv.com/knowledge-base/are-emotiv-products-medical-devices/>.
3. Basics of Neural Oscillations [Electronic resource] // Emotiv. — Mode of access: <https://www.emotiv.com/tutorials/basics-of-neural-oscillations/>.
4. BCI Guide [Electronic resource] // Emotiv. — Mode of access: <https://www.emotiv.com/bci-guide/>.
5. Brain-computer interface [Electronic resource]. — Mode of access: [https://en.wikipedia.org/wiki/Brain%E2%80%93computer\\_interface](https://en.wikipedia.org/wiki/Brain%E2%80%93computer_interface).
6. Cortex API Documentation [Electronic resource] // Emotiv. — Mode of access: <https://emotiv.gitbook.io/cortex-api/>.
7. Create a web API with ASP.NET Core. Microsoft Docs [Electronic resource]. — Mode of access: <https://learn.microsoft.com/en-us/aspnet/core/tutorials/min-web-api?view=aspnetcore-7.0&tabs=visual-studio>.
8. Extensions 101 - Chrome Developers. Google Chrome Developer Documentation [Electronic resource] / Emotiv. — Mode of access: <https://developer.chrome.com/docs/extensions/mv3/getstarted/extensions-101/>.
9. Insight 2 User Manual: Insight Sensor Coverage [Electronic resource] // Emotiv. — Mode of access: <https://emotiv.gitbook.io/insight-2-user-manual/introduction/insight-sensor-coverage>.
10. Insight [Electronic resource] // Emotiv. — Mode of access: <https://www.emotiv.com/insight/>.
11. Microsoft Azure SQL Database: Platform-as-a-Service (PaaS) Overview [Electronic resource]. — Mode of access: <https://learn.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview?view=azuresql>.
12. Neuroscience Guide [Electronic resource] // Emotiv. — Mode of access: <https://www.emotiv.com/neuroscience-guide/>.
13. Rashid M. Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review. *Frontiers in Neurorobotics*. [Electronic resource] / M. Rashid, N. Sulaiman, A. P. P. Abdul Majeed, R. M. Musa, A. F. Ab. Nasir, B. S. Bari, S. Khatun. — 2020. — Mode of access: <https://www.frontiersin.org/articles/10.3389/fnbot.2020.00025/full>.
14. TajDini M. Wireless Sensors for Brain Activity — A Survey [Electronic resource] / Mahyar TajDini, Volodymyr Sokolov, Ievgeniia Kuzminykh, Stavros Shiaeles, and Bogdan Ghita. — 2020. — Mode of access: [https://www.researchgate.net/publication/346714810\\_Wireless\\_Sensors\\_for\\_Brain\\_Activity-A\\_Survey](https://www.researchgate.net/publication/346714810_Wireless_Sensors_for_Brain_Activity-A_Survey).
15. What are the Detections Based On? How Were the Algorithms Created? [Electronic resource] // Emotiv. — 2019. — Mode of access: <https://www.emotiv.com/knowledge-base/what-are-the-detections-based-on-how-were-the-algorithms-created/>.
16. What are the Performance Metrics? Detection Suite [Electronic resource] // Emotiv. — 2019. — Mode of access: <https://www.emotiv.com/knowledge-base/what-are-the-performance-metrics-detection-suite/>.
17. Where Can I Get Access to Raw EEG API? [Electronic resource] // Emotiv. — Mode of access: <https://www.emotiv.com/knowledge-base/where-can-i-get-access-to-raw-eeeg-api/>.

## References

- A Beginner's Guide to Brain-Computer Interface and Convolutional Neural Networks by Alexandre Gonfalonieri. (2018, Nov. 25). <https://towardsdatascience.com/a-beginners-guide-to-brain-computer-interface-and-convolutional-neural-networks-9f35bd4af948>.
- Emotiv. *Neuroscience Guide*. <https://www.emotiv.com/neuroscience-guide/>.
- Emotiv. (2019, May 31). *What are the Detections Based On? How Were the Algorithms Created?* <https://www.emotiv.com/knowledge-base/what-are-the-detections-based-on-how-were-the-algorithms-created/>.
- Emotiv. (2021, Sept. 11). *What are the Performance Metrics? Detection Suite*. <https://www.emotiv.com/knowledge-base/what-are-the-performance-metrics-detection-suite/>.
- Emotiv. *Are Emotiv Products Medical Devices?* <https://www.emotiv.com/knowledge-base/are-emotiv-products-medical-devices/>.
- Emotiv. *Basics of Neural Oscillations*. <https://www.emotiv.com/tutorials/basics-of-neural-oscillations/>.
- Emotiv. *BCI Guide*. <https://www.emotiv.com/bci-guide/>.
- Emotiv. *Cortex API Documentation*. <https://emotiv.gitbook.io/cortex-api/>.
- Emotiv. *Insight* <https://www.emotiv.com/insight/>.
- Emotiv. *Insight 2 User Manual: Insight Sensor Coverage*. <https://emotiv.gitbook.io/insight-2-user-manual/introduction/insight-sensor-coverage>.
- Emotiv. *Where Can I Get Access to Raw EEG API?* <https://www.emotiv.com/knowledge-base/where-can-i-get-access-to-raw-eeeg-api/>.
- Google. (2023, Sept. 8). *Extensions 101 – Chrome Developers. Google Chrome Developer Documentation*. <https://developer.chrome.com/docs/extensions/mv3/getstarted/extensions-101/>.
- Microsoft Azure SQL Database: Platform-as-a-Service (PaaS) Overview. (2023, March. 3). <https://learn.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview?view=azuresql>.
- Microsoft. (2023, Aug. 22). *Create a web API with ASP.NET Core. Microsoft Docs*. <https://learn.microsoft.com/en-us/aspnet/core/tutorials/min-web-api?view=aspnetcore-7.0&tabs=visual-studio>.
- Rashid, M., Sulaiman, N., Abdul Majeed, A. P. P., Musa, R. M., Ab. Nasir, A. F., Bari, B. S., & Khatun, S. (2020, June 3). *Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review. Frontiers in Neurorobotics*. <https://www.frontiersin.org/articles/10.3389/fnbot.2020.00025/full>.
- TajDini, M., Sokolov, V., Kuzminykh, I., Shiaeles, S., & Ghita, B. (2020, Dec.). *Wireless Sensors for Brain Activity — A Survey*. [https://www.researchgate.net/publication/346714810\\_Wireless\\_Sensors\\_for\\_Brain\\_Activity-A\\_Survey](https://www.researchgate.net/publication/346714810_Wireless_Sensors_for_Brain_Activity-A_Survey).
- Wikipedia contributors. *Brain-computer interface*. [https://en.wikipedia.org/wiki/Brain%E2%80%93computer\\_interface](https://en.wikipedia.org/wiki/Brain%E2%80%93computer_interface).

A. Hlybovets, S. Khmel, O. Pyechkurova

## USAGE OF NEURO-COMPUTER INTERFACES FOR COLLECTING AND STORING INFORMATION ABOUT WEB RESOURCE USERS

*Brain-computer interface (BCI) is a link between the electrical activity of the brain and a computer or other external devices. Signals obtained from the brain can be electrophysiological, magnetic, or metabolic. Conversion of these signals into commands allows for controlling devices and interacting with other software interfaces.*

*The aim of this paper is to describe the development of a prototype that demonstrates the capabilities of using neuro-computer interfaces for collecting and storing information about web resource users. By utilizing the developed prototype, the interaction between users and web resources can be enhanced, allowing for browser control through mental commands and impulses transmitted by a BCI device. The gathered information during the use of web resources with the help of a BCI device will aid in better understanding user behavior and needs, improving the interaction with web resources, and ensuring user safety.*

*To collect user data, a Chrome extension was developed using JavaScript. Bootstrap is used for styling the extension, although the choice of library is not essential. The graphical visualization of indicators is implemented using Apexcharts, but this is also not a mandatory solution. Webpack is used for building the extension, following the principles of Chrome Manifest v3.*

*The extension interacts with the headset through the Cortex API. A graphical user interface was created to display the obtained data, as well as interfaces for user registration and login. Data from the headset is obtained through Websockets. An API was developed using minimal .NET API to store the collected data. CRUD is used for data exchange, and JWT is employed for authentication.*

*The developed prototype allows users to control Chrome pages using mental commands and collect data about the user experience of web resources, supplemented by data about the user's emotional and mental state.*

**Keywords:** neuro-computer interfaces, BCI, collection and storage of information about web resource users, interaction with web resources, security.

*Матеріал надійшов 20.09.2023*



Creative Commons Attribution 4.0 International License (CC BY 4.0)



Федорченко В. М.

## ДЕКЛАРАТИВНЕ КОНФІГУРУВАННЯ ІНВЕРСІЇ КОНТРОЛЮ В СУЧАСНИХ .NET8 ЗАСТОСУНКАХ

З огляду на постійне підвищення складності сучасних програмних систем стає дедалі важливішим значення впровадження компонентно-орієнтованих архітектурних парадигм. Шаблони інверсії контролю (IoC) та ін'єкції залежностей (DI) відіграють ключову роль у керуванні залежностями об'єктно-орієнтованих (ОО) компонентів, зменшенні зв'язності та забезпеченні безперешкодної інтеграції компонентів. Ця стаття досліджує теоретичні основи IoC та DI, розкриваючи їх практичну реалізацію в сучасних додатках .NET 8.

**Ключові слова:** інверсія контролю, ін'єкція залежностей, компонент-орієнтована розробка, .NET Core, .NET8.

Абстрагування залежностей між ОО-компонентами є одним із ключових факторів, що визначають важливі характеристики програмного коду: гнучкість, ефективність супроводу та можливість повторного використання. Низький рівень зв'язності зменшує кількість помилок, дає змогу легше змінювати компоненти, проводити їх незалежне тестування і, як наслідок, підвищує загальну якість програмного забезпечення [2]. Для забезпечення низького рівня зв'язності використовують такі архітектурні шаблони, як інверсія контролю (Inversion-of-Control) та ін'єкція залежностей (Dependency Injection) [2; 6]. Сутність шаблону «інверсія керування» полягає в абстрагуванні створення та ініціалізації компонентів через делегування цих дій особливому компоненту (DI-контейнеру). Створення компонентів і визначення залежностей (dependencies resolution) у вигляді конкретних екземплярів (об'єктів) виконує такий контейнер; компонентам більше не потрібно знати, як створювати чи налаштовувати свої залежності, що робить їх більш незалежними та здатними до адаптації. Це приводить до коду, який легше тестувати, підтримувати та розширювати.

У сучасних об'єктно-орієнтованих програмних платформах подібні архітектурні рішення стали невід'ємною частиною, і Microsoft .NET не є винятком. Починаючи з .NET Core 1.0 (2016) всі стандартні компоненти платформи розроблено з урахуванням того, що життєвий цикл і залежності цих компонентів (сервісів) контролюються вбудованим DI-контейнером [4], а зв'язування та визначення реалізацій інтерфейсів повністю абстраговано від самих компонентів.

Розглянемо компонент як блок, що має здатність до композиції з визначеними інтерфейсами взаємодії та явно описаним контекстом залежностей. Такий програмний компонент може бути використаний незалежно і передбачає можливість композиції з іншими компонентами. Одна з проблем компонентної розробки полягає в тому, що вона потребує від розробників відстеження залежностей, щоб визначити компоненти, які мають стосунок до задачі, і вирішити, які компоненти та як слід повторно використовувати [7]. В контексті об'єктно-орієнтованої програми компоненти зазвичай визначаються у вигляді класу (або групи пов'язаних класів) з відповідно визначеним інтерфейсом та чітко окресленими залежностями (C#):

```
public class AComponent :  
    IMyComponent  
{  
    TextWriter Output;  
    public AComponent(TextWriter  
output) {  
        Output = output;  
    }  
    public void WriteMessage(string  
message) {  
        output.WriteLine($"Message:  
{message}");  
    }  
}
```

Декларація цього компонента (сервісу) в стандартному DI-контейнері виконується шляхом виклику `IServiceCollection.Add (ServiceDescriptor)` або відповідного методу-розширення (код в методі `Startup.ConfigureServices`):

```
services.AddScoped<IMyComponent, MyComponent>();
```

Якщо в декларації компонента немає експліцитного визначення необхідних залежностей (параметрів конструктора `MyComponent`), контейнер імпліцитно визначає відповідні посилання за типами цих параметрів.

Варто зауважити, що ця вбудована реалізація DI-контейнера має суттєві обмеження, які перешкоджають застосуванню деяких аспектів техніки інверсії контролю:

- не підтримується можливість визначати декілька компонентів, які мають той самий тип, що значно обмежує здатність компонентів до композиції;
- не підтримується імпліцитна ін'єкція залежностей, які визначені у вигляді публічної властивості класу (`property injection`);
- немає підтримки декларативного визначення компонентів та їх залежностей у вигляді JSON або XML конфігурації.

У найновішій версії платформи .NET8 (2023) з'явилась підтримка іменованих сервісів (`keyed services`), що дає змогу визначити декілька компонентів одного типу:

```
services.AddKeyedScoped<AComponent>("component1");
services.AddKeyedScoped<AComponent>("component2");
```

Щоб отримати посилання саме на конкретний компонент, декларація залежності має виглядати так:

```
public BComponent([FromKeyedServices("component1")] MyComponent c)
```

Нескладно помітити суттєвий недолік: подібна декларація порушує принцип інверсії контролю, тому що реалізація (компонент) містить посилання на ідентифікатор компонента. Цьому можна запобігти через експліцитне визначення залежностей:

```
services.
AddKeyedScoped<BComponent>(srv =>
    new BComponent(srv.GetRequiredKeyedService<AComponent>("component1")));
```

Так само можна зробити ін'єкцію властивостей, але такий опис має форму імперативного коду для визначення всіх залежностей (яких у реальних компонентів буває багато), що само по собі підвищує зв'язність в застосунку та прово-

кує появу значної кількості шаблонного (і надлишкового) коду, який конфігурує DI-контейнер. Ці особливості стандартного контейнера сприяють ситуації, коли в реальних .NET програмах лише обмежена кількість компонентів «верхнього рівня» реєструється в контейнері, самі компоненти мають надлишкову зв'язність, а багато більш низькорівневих компонентів залишаються зв'язані жорстко і для них інверсія контролю не застосовується.

Для вирішення цих недоліків стандартного DI-контейнера пропонується реалізувати розширення у вигляді фабрики компонентів, яка буде реєструвати компоненти в контейнері та визначати їх залежності відповідно до декларативної конфігурації, що може зберігатися в стандартному (для .NET Core застосунків) файлі `appsettings.json`, або завантажуватись з окремого JSON файлу (або взагалі якимось чином генеруватись динамічно):

```
[
  {Type: "AComponent", Name:
"component1", Lifetime: "Scoped"},
  {Type: "AComponent", Name:
"component2"},
  {
    Type": "BComponent",
    Constructor: [{$ref: "component1"}],
    Properties: {
      ValueProp: 10,
      DependencyProp: {$ref:
"component1"}
    }
  },
]
```

Розглянемо основні технічні аспекти реалізації такої фабрики компонентів.

Для внутрішнього представлення опису компонентів та їх залежностей (які декларуються як параметри конструктора класу та/або публічні `set`-властивості) визначимо таку об'єктну модель.

Відповідно до цієї моделі весь граф залежностей компонентів може бути визначений як множина `ComponentDescriptor`. Залежності конкретного компонента визначаються через списки параметрів конструктора та/або ін'єкції властивостей класу компонента. Кожна така залежність може бути посиланням на інший компонент (`RefDescriptor`), або значенням простого типу даних (`ValueDescriptor`), або списком чи словником (`ListDescriptor` та `DictionaryDescriptor`), значення якого своєю чергою можуть бути описані як посилання на компоненти чи прості значення. Посилання на інший компонент може

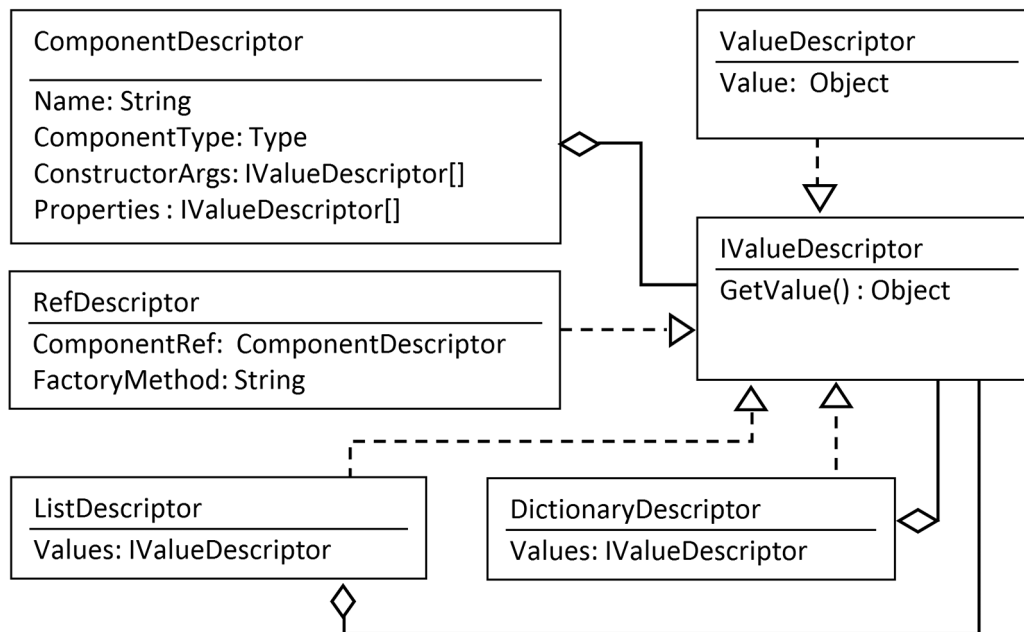


Рис. 1. Об'єктна модель опису компонентів

бути за назвою, за типом або як inline-визначення. Подібне внутрішнє представлення дає змогу абстрагувати фабрику компонентів від конкретного формату конфігурації, який може бути не тільки JSON, а й XML або YAML.

Функціональна декомпозиція розширення стандартного DI-контейнера, яке реалізує підтримку декларативної конфігурації залежностей, складатиметься з таких елементів:

- **JsonLoader.** Оскільки файл конфігурації сучасних .NET застосунків (appsettings.json) має JSON формат, для парсингу конфігурації залежностей використовуємо стандартний пакет System.Text.Json. Ітеруючи елементи JsonDocument, створимо об'єктну модель графу залежностей компонентів. На цьому етапі виконуються валідація та зв'язування всіх типів компонентів, вказаних у конфігурації. Для цього призначена типова реалізація ITypeResolver, яка використовує механізм відображень (Reflection) для визначення типів (System.Type) за їх повною або частковою назвою. При цьому .NET застосунок має можливість визначити свою реалізацію ITypeResolver для підтримки скорочених назв типів; наприклад, це можуть бути списки пріоритетних областей імен (namespaces), в яких буде проводитись пошук таких скорочених назв типів.
- **ComponentFactory.** Реалізація фабрики конкретного компонента, яка створює екземпляри відповідно до опису у вигляді ComponentDescriptor. Ця процедура передба-

чає пошук найбільш відповідного конструктора класу компонента (якщо їх декілька), визначення експліцитних (за посиланням у конфігурації) та/або імпліцитних посилань (за типом параметра конструктора) на екземпляри інших компонентів. Оскільки життєвий цикл компонентів контролюється стандартним DI-контейнером, для отримання цих посилань на інші компоненти фабрика визначає залежність від IComponentContainer, який своєю чергою реалізується через механізм IServiceProvider стандартного DI-контейнера (ServiceProviderComponentFactory). Варто зазначити, що декларативна конфігурація конкретної ін'єкції може визначатись як inline-визначення компонента; такі компоненти не реєструються в DI-контейнері, а отже, їх екземпляри створюються безпосередньо (без використання механізму IServiceProvider).

- **IServiceCollection.LoadComponentsFromJson.** Інтеграція фабрики компонентів зі стандартним DI-контейнером імплементується через метод розширення (extension method), який використовує JsonLoader для завантаження конфігурації у вигляді об'єктної моделі (множини ComponentDescriptor), реєструє компоненти в DI-контейнері, і як фабричний метод (який створює екземпляр компонента) використовує ComponentFactory:

```

foreach (ComponentDescriptor c in
components) {
    var factory = new
  
```

```

ComponentFactory(c);
Func<IServiceProvider, object,
object> create = (srvPrv, key) =>
{
    return factory.Create(
        new ServiceProviderComponentF
actory(srvPrv),
c.ImplementationType);
};
var descriptor =
ServiceDescriptor.DescribeKeyed(
c.ServiceType ?? c.
ImplementationType,
c.Name, create, c.Lifetime);
services.Add(descriptor);
}

```

Отже, для підключення декларативної конфігурації компонентів у будь-якому сучасному .NET застосунку достатньо додати лише один рядок коду в `Startup.ConfigureServices`:

```
services.LoadComponentsFromJson("com
ponents.json");
```

Повна реалізація фабрики компонентів розміщена на GitHub у вигляді бібліотеки: <https://github.com/nreco/dependencyinjection>.

За допомогою такого розширення шаблону «інверсія контролю» може бути застосований не тільки до прикладних компонентів, а й до самого DI-контейнера. Оскільки конфігурування залежностей повністю абстраговане від програмного коду застосунку, зміна цієї конфігурації не потребує перекомпіляції застосунку, що своєю чергою значно підвищує можливості адаптації програми до конкретного середовища виконання —

наприклад, версії для різних операційних систем або різних провайдерів хмарного хостингу. Наявність такого альтернативного способу декларації компонентів створює сприятливі технічні можливості для активного використання функціональних інтерфейсів при декомпозиції [5; 8] та підтримку ін'єкцій таких функціональних залежностей з автоматичним узгодженням [1].

Сама наявність декларативної конфігурації DI-контейнера відкриває широкі можливості для генеративного програмування. Стає технічно можливим впровадження модель-орієнтованої розробки, зокрема підходу, який використовує конфігурацію DI-контейнера як кінцеву модель рівня виконання в ланцюжку трансформацій предметно-специфічних моделей [3].

Компонент-орієнтована розробка є загальноприйнятим ефективним методом зниження витрат виробництва програмних продуктів, скорочення часу виходу на ринок та підвищення якості [7]. Сучасна платформа .NET8 передбачає, що програмне забезпечення збирається з бібліотек уже написаних компонентів та надає стандартизований механізм середовища часу виконання для цих компонентів. Розглянуто практичні аспекти застосування шаблону «інверсія контролю» за умови використання стандартної реалізації DI-контейнера, визначено наявні технологічні обмеження, а також запропоновано рішення для декларативного визначення залежностей між компонентами, яке сприяє низькому рівню зв'язності між компонентами та декомпозиції програми на більш гранулярні та більш абстрактні компоненти, що також підвищує імовірність їх повторного використання.

### Список літератури

1. Глибовець М. М. Ін'єкція функціональних залежностей у контейнері інверсії керування / М. М. Глибовець, В. М. Федорченко // Проблеми програмування. — 2014. — № 4. — С. 33–39.
2. Fowler M. Reducing Coupling / Martin Fowler // IEEE Software. — 2001. — Vol. 18, no. 4. — Pp. 102–104.
3. Glibovets N. N. Simplified infrastructure for the transformation of XML models / N. N. Glibovets, V. M. Fedorchenko // Cybernetics and Sys. Anal. — 2010. — Vol. 46, no. 1. — Pp. 93–97. <https://doi.org/10.1007/s10559-010-9187-0>.
4. Larkin K. Dependency injection in ASP.NET Core [Electronic resource] / K. Larkin, S. Smith, B. Dahler. — Mode of access: <https://learn.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection>.
5. Lecessi R. Functional Interfaces in Java / Ralph Lecessi. — Apress, 2019.
6. Martin R. The Dependency Inversion Principle / Martin Robert C. // C++ Report. — 1996. — Vol. 8.
7. Szyperski C. Component Software: Beyond Object-Oriented Programming / Clemens Szyperski. — Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 2002.
8. Velasco Elizondo P. Deriving functional interface specifications for composite components / P. Velasco Elizondo, M. K. C. Ndjatchi // Springer eBooks. — 2011. — Pp. 1–17. [https://doi.org/10.1007/978-3-642-22045-6\\_1](https://doi.org/10.1007/978-3-642-22045-6_1).

### References

- Fowler, M. (2001). Reducing coupling. *IEEE Software*, 18 (4), 102–104. <https://doi.org/10.1109/ms.2001.936226>.
- Glibovets, N. N., & Fedorchenko, V. M. (2010). Simplified infrastructure for the transformation of XML models. *Cybernetics and Systems Analysis*, 46 (1), 93–97. <https://doi.org/10.1007/s10559-010-9187-0>.
- Hlybovets, M. M., & Fedorchenko, V. M. (2014). Iniektsiia funktsionalnykh zalezhnosti u konteineri inversii keruvannia. *Problemy prohramuvannia*, 4, 33–39.
- Larkin, K., S. Smith, S., & B. Dahler, B. — *Dependency injection in ASP.NET Core*. <https://learn.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection>.

- Lecessi, R. (2019). *Functional interfaces in Java: Fundamentals and Examples*. Apress.
- Martin, R. (1996). The Dependency Inversion Principle. C++ Report, 8.
- Szyperski, C. (2002). *Component software: Beyond Object-Oriented programming*.
- Velasco-Elizondo, P., & Ndjatchi, C. (2011). Deriving functional interface specifications for composite components. In *Springer eBooks* (pp. 1–17). [https://doi.org/10.1007/978-3-642-22045-6\\_1](https://doi.org/10.1007/978-3-642-22045-6_1).

V. Fedorchenko

## DECLARATIVE INVERSION OF CONTROL CONFIGURATION IN MODERN .NET8 APPLICATIONS

*As software systems continue to evolve in complexity and scale, the importance of adopting component-based architectural paradigms becomes increasingly evident. Inversion of Control (IoC) and Dependency Injection (DI) patterns play important role in managing OO-components dependencies, reduce coupling and enable seamless integration of components. This article explores the theoretical foundations of IoC and DI, shedding light on their practical implementation in modern .NET applications.*

*Standard .NET DI-container has a number of technical limitations: properties injection is not supported, dependencies definitions from keyed services partially breaks IoC principle, and finally lack of declarative way to define components and their dependencies. To address these limitations, it is proposed to implement a special extension for standard DI-container in the form of a component factory. This factory would register components within the container and define their dependencies based on declarative configuration. The configuration can be stored either in a standard (for .NET Core apps) appsettings.json file or loaded from a separate JSON file. In fact, this JSON could be even dynamically generated in specific usage scenarios.*

*The capability to use a declarative configuration for standard DI-container opens up broad possibilities for generative programming. It becomes technically feasible to implement a model-driven development, particularly an approach that leverages the DI-container configuration as the final (execution) model in a chain of transformations of domain-specific models.*

*Modern .NET8 platform assumes that software is assembled from libraries of pre-existing components that are hosted in a standard DI-container that provides suitable runtime environment for them. Practical aspects of applying the Inversion of Control pattern are examined, considering the use of a standard Dependency Injection (DI) container implementation. Existing technological limitations are defined, and a solution is proposed: implementation of declarative configuration of dependencies between components. This approach contributes to a reduced level of coupling between components and facilitates the decomposition of the program into more granular and abstract components – which increases their reusability in consequence.*

**Keywords:** inversion of control, dependency injection, component based development, declarative IoC configuration, .net core, .NET8.

Матеріал надійшов 01.12.2023



Creative Commons Attribution 4.0 International License (CC BY 4.0)

Пухальський Б. М., Кирієнко О. В.

## СИСТЕМИ ОБРОБЛЕННЯ НЕСТРУКТУРОВАНОЇ МУЛЬТИМЕДІЙНОЇ ІНФОРМАЦІЇ

*У роботі представлено огляд систем пошуку зображень і методів, які використовують для вирішення проблем комп'ютерного зору. У першій частині увагу зосереджено на потребах користувачів і вимогах до систем пошуку зображень, після чого розглянуто наявні системи, як-от Google Photos, Apple Photos, Amazon Photos, Microsoft Photos і Flickr; оцінено їхні переваги та недоліки та обговорено запропоновані шляхи вирішення виявлених недоліків. Детально проаналізовано запропоновані методи комп'ютерного зору, зокрема виявлення об'єктів (облич) і текстовий опис зображень.*

**Ключові слова:** системи пошуку зображень, комп'ютерний зір, виявлення об'єктів, текстовий опис зображень.

### Вступ

Швидке зростання цифрових зображень зумовило збільшення потреби в ефективних системах пошуку зображень, які могли б ефективно задовольнити потреби користувачів. У цій статті зроблено докладний огляд систем пошуку зображень і методів, які використовують для вирішення проблем комп'ютерного зору шляхом покращення можливостей пошуку та індексування.

Основною метою цього дослідження є аналіз потреб користувачів і вимог до систем пошуку зображень, визначення ключових аспектів, які сприяють ефективній та зручній роботі. Розуміючи ці вимоги, ми можемо оцінити системи, доступні на ринку, і вивчити їхні сильні та слабкі сторони. Зокрема, ми зосередимося на добре відомих системах пошуку зображень: Google Photos, Apple Photos, Amazon Photos, Microsoft Photos і Flickr. Ці платформи пропонують різноманітні підходи до пошуку зображень і цінну інформацію про поточний стан галузі. Окрім того, наше дослідження сприятиме знаходженню нових способів застосування комп'ютерного зору у виявленні об'єктів.

Результати цієї роботи можуть слугувати цінним ресурсом для дослідників, розробників і професіоналів галузі, які займаються розробленням і впровадженням систем пошуку зображень. Запропоновані рішення та висновки, отримані в результаті цього дослідження, можуть керувати майбутніми досягненнями в галузі.

### 1. Системи пошуку зображень

Мультимедіа — це форма комунікації, що поєднує такі види інформації, як текст, аудіо, зображення, анімація та відео [8]. На відміну від «традиційних» засобів подання інформації, мультимедіа надає користувачеві засоби взаємодії. Мультимедійна інформація репрезентується, обробляється та передається через численні сенсорні канали, як-от камери спостереження, датчики звуку тощо. Поєднання різних типів медіа підвищує насиченість і ефективність спілкування, створюючи захопливий та інтерактивний досвід.

У контексті систем пошуку зображень мультимедіа відіграє вирішальну роль у захопленні та представленні візуальної інформації. Мультимедіа охоплює широкий діапазон типів медіа: статичні (текст, зображення), динамічні (аудіо, відео) та багатовимірні (віртуальна реальність) [10] (рис. 1).

Текстова інформація, як-от підписи, описи та метадані, надає контекст і допомагає зрозуміти вміст мультимедійних об'єктів [13]. Зображення — це візуальні уявлення, які можуть передавати інформацію, викликати емоції і бути засобом спілкування [3]. Аудіокомпоненти містять звукові доріжки, голос за кадром, фонову музику або будь-яку форму аудіозаписів, які супроводжують мультимедійні презентації [6]. Відео поєднує в собі рухомі зображення та аудіо, що дає змогу знімати та відтворювати динамічні сцени, події або оповідання [3]. Анімація перед-

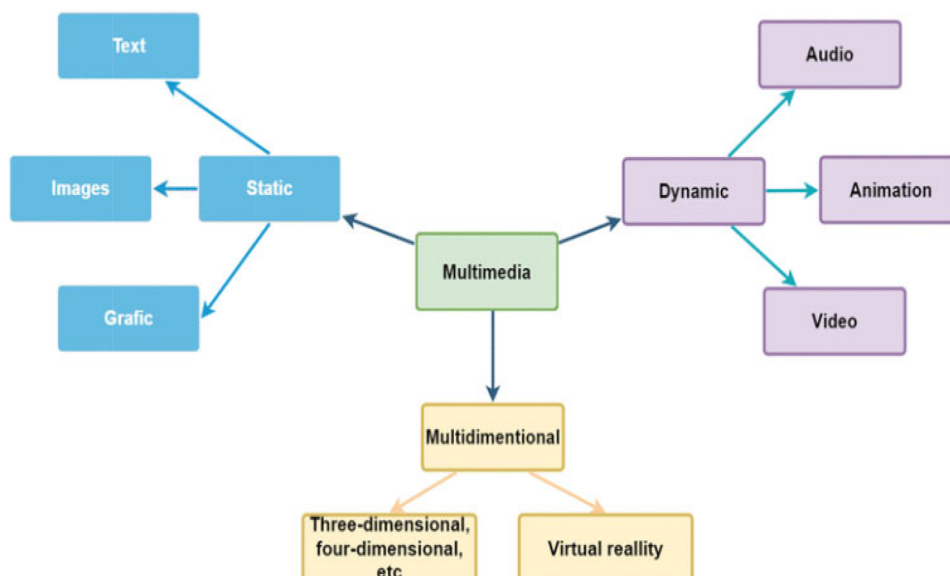


Рис. 1. Схема класифікації мультимедійної інформації [10]

бачає послідовне відображення зображень або візуальних елементів для створення ілюзії руху або зміни [13].

Кожен із цих типів потребує спеціальних методів і алгоритмів для ефективного зберігання, індексування та пошуку.

Фотографії посідають важливе місце в мультимедіа, і їх ефективний пошук є ключовою метою систем пошуку зображень. Для отримання відповідних фотографій із великих колекцій потрібні надійні механізми індексування [12].

Індексація фотографій передбачає вилучення значущих характеристик із зображень, таких як колір, текстура, форма та просторова інформація [3]. Потім ці функції використовуються для створення компактних представлень або дескрипторів зображень, які можна ефективно зберігати і здійснювати пошук [13].

Щоб увімкнути подальший пошук та індексацію, ці дескриптори зображень організуються та індексуються за допомогою відповідних структур даних, наприклад інвертованих файлів, хеш-таблиць або деревоподібних структур [12]. Цей процес індексування дає змогу здійснювати ефективний пошук на основі різних критеріїв, як-от схожість, релевантність або специфічні візуальні атрибути [3]. Пошук за дескрипторами не є новим, він широко використовується у різноманітних пошукових системах. Ці системи спрямовані на подолання семантичного розриву між низькорівневими візуальними функціями та високорівневими запитами користувача, забезпечуючи ефективний пошук і перегляд мультимедійного вмісту [6]. Вони відіграють життєво

важливу роль у різних сферах, зокрема у цифрових бібліотеках, базах даних зображень, платформах соціальних медіа та особистих колекціях фотографій.

Зазвичай користувачам систем пошуку мультимедійної інформації потрібно знайти зображення: за датою, за геолокацією, за об'єктами, зображеними на фото, за особами, зображеними на фото, за описом дій, що виконуються на фото.

Перші дві згадані функції реалізовано в більшості систем пошуку. Така інформація міститься у метаданих зображення, а перед тим наповнюється програмним забезпеченням, встановленим на пристрої, із якого роблять фотографію. Решта функцій здебільшого реалізовані в інформаційних системах із використанням нейромереж.

У таблиці 1 відображено реалізацію цих функцій у відомих системах.

Таблиця 1

#### Порівняльна характеристика функціоналу розглянутих систем

	Пошук за об'єктами	Пошук за обличчями	Пошук за діями
Google Photos	+	+	-
Apple Photos	+	+	-
Amazon Photos	+	+	-
Microsoft Photos	+	+	-
Flickr	+	+	-

Як бачимо, у кожному продукті реалізовано функціонал розпізнавання об'єктів, і облич також. Однак за допомогою жодного розглянутого застосунку не можна знайти фото за описом.

Продуктивність є важливим атрибутом якості інформаційних систем. Надмірне очікування

є неприпустимим для користувача системи. Крім того, продуктивність тісно пов'язана із масштабованістю, оскільки такі показники, як пропускна здатність і час очікування відповіді, визначають момент для розширення обчислювальних потужностей системи.

Реалізацію зазначеного функціоналу можна розглядати як розв'язання типових задач комп'ютерного зору. Є чотири основні класи задач комп'ютерного зору: класифікація зображень, виявлення об'єктів, сегментація зображень і текстовий опис зображень [9].

## 2. Методи розв'язання задач комп'ютерного зору

Найпоширенішими задачами комп'ютерного зору є класифікація зображень, визначення об'єктів (зокрема облич), сегментація зображень, текстовий опис зображень. Особливий інтерес становлять задачі з виявлення об'єктів (облич) і текстового опису зображень.

Виявлення об'єктів — це важливе завдання комп'ютерного зору, яке має справу з виявленням екземплярів візуальних об'єктів певного класу (як-от люди, тварини чи автомобілі) у цифрових зображеннях. Метою виявлення об'єктів є розроблення обчислювальних моделей і методів, які надають одну з основних частин інформації, необхідної для програм комп'ютерного зору, тобто дані про те, де які об'єкти розташовані [1; 11].

Застосування виявлення об'єктів у сфері охорони здоров'я зумовило революцію в медичній візуалізації та діагностиці, давши змогу автоматизовано виявляти та локалізувати анатомічні структури, ураження і аномалії. Ці програми використовують методи комп'ютерного зору, алгоритми глибокого навчання і передові технології візуалізації для підвищення точності та ефек-

тивності виявлення захворювань, діагностики і планування лікування. Завдяки автоматизації процесу виявлення об'єктів медичні працівники можуть заощадити час, покращити результати лікування пацієнтів і сприяти більш персоналізованому та цілеспрямованому втручанню [7].

Шлях розвитку рішень для задачі з виявлення об'єктів можна умовно поділити на два періоди. Перший період — це традиційні методи рішення задачі, розвиток яких закінчився у 2014 р., а саме алгоритми, що базуються на ознаках об'єкта, заданих вручну. Із розвитком нейронних мереж і глибокого навчання розпочався другий період, методи якого базуються на алгоритмах машинного навчання. Розвиток у цьому напрямі триває [16]. Хронологію розвитку рішень задачі з виявлення об'єктів зображено на рис. 2.

Традиційні методи розв'язання задачі з виявлення об'єктів поділяють на такі, що ґрунтуються на особливостях, такі, що ґрунтуються на шаблонах, і такі, що ґрунтуються на зовнішньому вигляді.

Відповідно до технік, які базуються на особливостях, визначаються риси, притаманні певному об'єктові. Для обличчя це очі, ніс, рот тощо. Наступним кроком є перевірка правдоподібного географічного розташування виявлених рис.

Техніки, що базуються на шаблонах, здатні виявляти об'єкт у різноманітних позиціях і поданнях. Однак для них потрібна належна ініціалізація, і їх швидкодія не є високою.

Згідно з технікою, що ґрунтується на зовнішньому вигляді, зображення поділяється на прямокутні зони, які можуть перекривати одна одну. Згодом у цих зонах буде знайдено кандидатів на об'єкт, який потрібно виявити. Після цього вибір можна уточнити, використовуючи каскад складніших, але селективних алгоритмів виявлення. Більшість підходів, основаних на зовнішньому вигляді, сильно покладаються на навчання кла-

### Object Detection Milestones

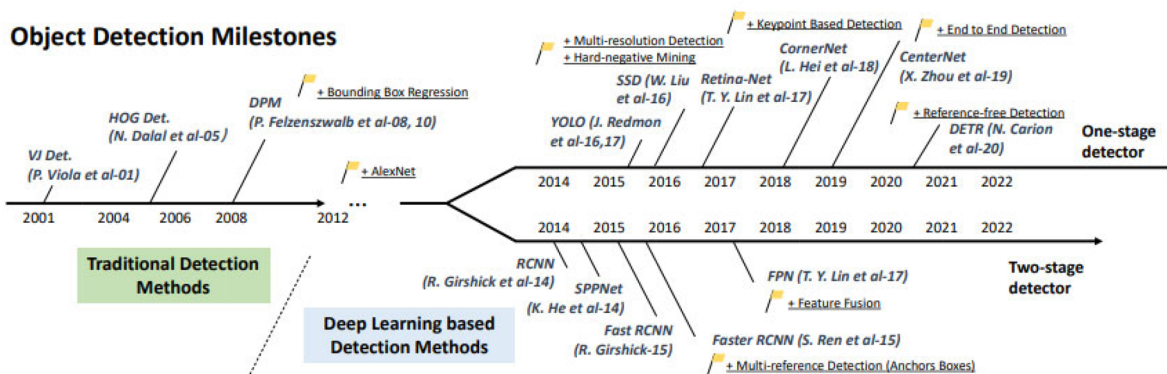


Рис. 2. Історія розвитку способів рішення задачі з виявлення об'єктів [16]



сифікаторів із використанням наборів мічених об'єктів і необ'єктних фрагментів. Серед «традиційних» детекторів найвідомішими є Viola Jones Detector, HOG Detector і Deformable Part-based Model (DPM).

Способи рішення періоду глибинного навчання поділяють на два типи, які розвивалися паралельно: виявлення у два етапи та виявлення в один етап. Перший спосіб визначає виявлення як процес «від грубого до тонкого», а другий є виявленням за один крок.

Найвідомішими двоетапними нейронними мережами є RCNN, SPPNet, FastRCNN, FasterRCNN. Короткий опис кожної із них подано нижче.

Першим етапом роботи нейромережі Regions with Convolutional Neural Networks (RCNN) є діставання пропозиції об'єктів. Потім кожна пропозиція масштабується до фіксованого розміру й передається до CNN моделі, натренованої на ImageNet, щоб дістати особливості. І наостанок, використовуються SVN класифікатори, щоб визначити наявність об'єкта у кожній області зображення та розпізнати категорії об'єкта.

Spatial Pyramid Pooling Networks (SPPNet) містить шар просторової піраміди, що дає змогу репрезентувати фіксовану довжину незалежно від розміру вхідного зображення або області без масштабування.

FastRCNN дає змогу одночасно тренувати детектор і регресор границь області за незмінної конфігурації мережі. Ця мережа є першим наскрізним і близьким до реального часу детектором. Головною особливістю є запровадження шару RPN, який забезпечує нересурсозатратні припущення областей. Крім того, окремі блоки системи виявлення об'єктів (виявлення припущення, вилучення особливостей, регресія границь області) поступово інтегрувалися в уніфікований наскрізний фреймворк навчання.

Серед одноетапних нейронних мереж для виявлення об'єктів варто згадати YOLO, SSD і Retina Net.

You Only Look Once (YOLO) — перший одноетапний детектор ери глибинного навчання. За принципом роботи ділить зображення на області, визначає границі та ймовірності для них. Основною відмінністю Single Shot Multibox Detector (SSD) є запровадження методів виявлення із декількома посиланнями та з різною роздільною здатністю, що значно підвищує точність виявлення одноетапного детектора, особливо для малих об'єктів. Оскільки одноетапні детектори не вирізняються точністю роботи, у Retina Net було введено нову функцію витрат

за назвою «фокальна витрата», яка змінює стандартні витрати перехресної ентропії, щоб під час навчання детектор більше зосереджувався на складних, неправильно класифікованих прикладах.

Визначено, що за принципом своєї роботи двоетапні нейронні мережі дають точніший результат, натомість одноетапні є швидшими. Вибираючи нейронну мережу, що буде застосовуватись у прикладному програмному забезпеченні, потрібно маневрувати між точністю та продуктивністю.

Окремо варто згадати про таку задачу комп'ютерного зору, як розпізнавання облич, що є окремим випадком задачі з виявлення об'єктів. Цю технологію застосовують у автоматизованому відстежуванні дотримання правил дорожнього руху, в розслідуванні кримінальних справ, у виявленні емоцій і накладанні масок у мобільних застосунках та в інших сферах. Хоч виявлення облич є підкласом більш загальної задачі, воно є дещо специфічним.

У процесі виявлення облич часто можна зіткнутися із такими складнощами, як варіативність, перекриття, виявлення у різних масштабах та виявлення у реальному часі [14]. Людське обличчя здатне виражати багато емоцій, причому кожна людина виражає їх по-своєму, до того ж існує велика варіативність кольору та відтінку шкіри, що ускладнює процес виявлення. Обличчя можуть бути частково закритими різноманітними об'єктами та аксесуарами, як-от волосся, головні убори, окуляри, макіяж, маска та інші. Масштаб фотографій облич також різниться. Лице може бути сфотографовано зблизька, здалеку або в натовпі. Для виявлення у реальному часі обличчя, яке містить досить багато характерних рис, потрібно залучати швидкі обчислювальні потужності, особливо коли йдеться про мобільні пристрої.

Згаданий детектор Viola Jones був однією з перших спроб рішення цієї проблеми і став основою для розвитку пізніших детекторів [2].

Текстовий опис зображень є порівняно новою проблемою у сфері комп'ютерного зору. Визначення проблеми полягає у генеруванні опису, який є найбільш доречним до заданого зображення. Опис містить об'єкти, які зображені на заданому мультимедійному файлі, та зв'язки між цими об'єктами. Серед зв'язків можуть бути дії, які виконують знайдені об'єкти, сполучники, прийменники та інші частини мови, що забезпечують цілісність опису. Не існує вимог щодо структури опису. Його можна надати в довільній формі [5].

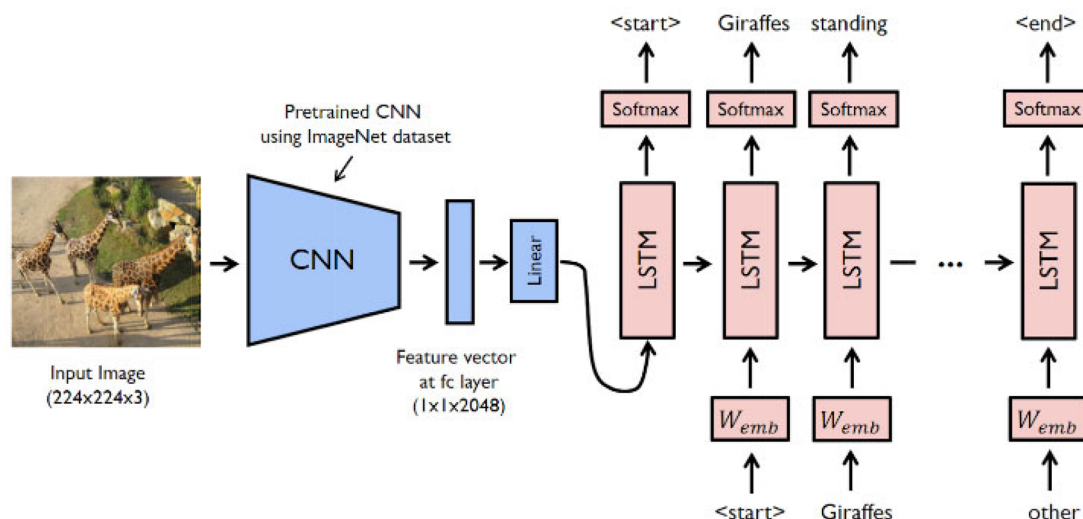


Рис. 3. Схема моделі Show and Tell [4]

Можна знайти багато застосувань вирішенням цієї проблеми. Найбільш очевидне — це генерація опису всього, що відбувається на вихідному зображенні, для підвищення ефективності пошуку самого зображення. Це дає нам змогу забезпечити посередництво між людиною з її високорівневими запитами та системою аналізу й пошуку зображень з її низькорівневими функціями, що суттєво підвищує зручність використання подібних систем.

Розглядувану функцію можна використовувати там, де є потреба забезпечити інклюзивність. Наприклад, люди із вадами зору, які не здатні чітко бачити мультимедійний вміст, отримують текстовий опис зображення зручним для них шрифтом.

По своїй суті текстовий опис зображень є комбінацією виявлення згаданих раніше об'єктів [9]. Протягом останніх майже десяти років виявлення виконують методами глибинного навчання, а опис — такими нейронними мережами, як RNN (Recurrent Neural Net) та LSTM (Long Short Term Memory), яка є розширеною версією RNN. Існує також модель Show and Tell, яка поєднує два згадані рішення з використанням LSTM [15]. На рис. 3 подано структуру цієї моделі.

Під час розроблення системи аналізу та пошуку зображень буде використано такий підхід із конфігураціями шарів для виявлення об'єктів.

## Висновки

Цю роботу було зосереджено на дослідженні систем пошуку зображень і використання методів комп'ютерного зору для ефективного пошуку та індексування. Проведено огляд систем пошуку, акцентовано на важливості пошуку зображень у ширшому мультимедійному контексті. Досліджено потреби та вимоги користувачів до систем пошуку зображень, що дало розуміння факторів, які сприяють ефективній взаємодії з користувачем.

Наведено порівняльну характеристику сучасних систем пошуку зображень, зокрема Google Photos, Apple Photos, Amazon Photos, Microsoft Photos і Flickr. Визначено переваги та недоліки цих систем.

У статті також описано основні методи вирішення проблем комп'ютерного зору. Акцент було зроблено на виявленні об'єктів, зокрема на розпізнаванні облич, і використанні текстового опису у пошуку зображень. В основному дано уявлення про методи, які використовують для розпізнавання обличчя та застосування текстового опису зображень, створення основи для подальшого розвитку системи пошуку та індексування зображень. Наведено порівняльну характеристику розглянутих способів.

## Список літератури

1. Як працює OSINT-розвідка? Від бізнес-аналізу до оборони України [Електронний ресурс] // ISSP Training Center. — 2022. — Режим доступу: [www.issp.training/post/yak-pratsyuye-osint-rozvidka-vid-biznes-analizu-do-oborony-ukrainy](http://www.issp.training/post/yak-pratsyuye-osint-rozvidka-vid-biznes-analizu-do-oborony-ukrainy).
2. Cheney J. Unconstrained face detection: State of the art baseline and challenges / Jordan Cheney, Ben Klein, Anil K. Jain, Brendan F. Klare // International Conference on Biometrics (ICB). — Phuket, Thailand, 2015. — Pp. 229–236. <https://doi.org/10.1109/ICB.2015.7139089>.

3. Datta R. Image retrieval: Ideas, influences, and trends of the new age / Ritendra Datta, Dhiraj Joshi, Jia Li, James Z. Wang // *ACM Computing Surveys*. — 2008. — Vol. 40, issue 2. — P. 60. <https://doi.org/10.1145/1348246.1348248>.
4. Image Captioning [Electronic resource]. — Mode of access: <http://shikib.com/captioning.html>.
5. Image Captioning [Electronic resource] // *Papers with Code*. — Mode of access: <https://paperswithcode.com/task/image-captioning>.
6. Lew M. S. Content-based multimedia information retrieval: State of the art and challenges / Michael S. Lew, Nicu Sebe, Chabane Djeraba, Ramesh Jain // *ACM Transactions on Multimedia Computing, Communications, and Applications*. — 2006. — Vol. 2, issue 1. — Pp. 1–19. <https://doi.org/10.1145/1126004.1126005>.
7. Litjens G. A survey on deep learning in medical image analysis / Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A. W. M. van der Laak, Bram van Ginneken, Clara I. Sánchez // *Medical Image Analysis*. — 2017. — Vol. 42. — Pp. 60–88. <https://doi.org/10.1016/j.media.2017.07.005>.
8. Multimedia [Electronic resource]. — Mode of access: <https://en.wikipedia.org/wiki/Multimedia>.
9. Murali S. An analysis on computer vision problems [Electronic resource] / Shravan Murali. — 2018. — Mode of access: <https://medium.com/deep-dimension/an-analysis-on-computer-vision-problems-6c68d56030c3>.
10. Nikulin D. What is Multimedia Data? [Electronic resource] / Dima Nikulin // DEV Community. — 2022. — Mode of access: <https://dev.to/dimanikulin/what-is-multimedia-data-111f>.
11. Raikote P. Object Detection — Part 1: Introduction [Electronic resource] / Pranav Raikote // *Applied Singularity*. — 2021. — Mode of access: <https://appliedsingularity.com/2021/05/18/object-detection/>.
12. Sivic J. Video Google: a text retrieval approach to object matching in videos / Josef Sivic, Andrew Zisserman // *Proceedings Ninth IEEE International Conference on Computer Vision, Nice, France*. — 2003. — Vol. 2. — Pp. 1470–1477. <https://doi.org/10.1109/ICCV.2003.1238663>.
13. Smeulders A. Content-based image retrieval at the end of the early years // A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. — 2000. — Vol. 22, no. 12. — Pp. 1349–1380. <https://doi.org/10.1109/34.895972>.
14. Varun. What is Face Detection? Ultimate Guide 2023 + Model Comparison [Electronic resource] / Varun // *LearnOpenCV*. — 2022. — Mode of access: <https://learnopencv.com/what-is-face-detection-the-ultimate-guide/>.
15. Vinyals O. Show and Tell: A Neural Image Caption Generator / Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. — June, 2015.
16. Zou Z. Object Detection in 20 Years: A Survey / Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, Jieping // *Proceedings of the IEEE*. — 2023. — Vol. 111, no. 3. — Pp. 257–276. <https://doi.org/10.1109/JPROC.2023.3238524>.

### References

- Cheney, J., Klein, B., Jain, A. K., & Klare, B. (2015). Unconstrained face detection: State of the art baseline and challenges. *2015 International Conference on Biometrics (ICB)*. <https://doi.org/10.1109/icb.2015.7139089>.
- Datta, R., Joshi, D., Li, J., & Wang, J. (2008). Image retrieval. *ACM Computing Surveys*, 40 (2), 1–60. <https://doi.org/10.1145/1348246.1348248>.
- Image Captioning. (n.d.). Captioning. <http://shikib.com/captioning.html>.
- Lew, M. S., Sebe, N., Djeraba, C., & Jain, R. (2006). Content-based multimedia information retrieval. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2 (1), 1–19. <https://doi.org/10.1145/1126004.1126005>.
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J., Van Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88. <https://doi.org/10.1016/j.media.2017.07.005>.
- Murali, S. (2018, June 17). An analysis on computer vision problems – Shravan’s Blog - Medium. *Medium*. <https://medium.com/deep-dimension/an-analysis-on-computer-vision-problems-6c68d56030c3>.
- Nikulin, D. (2022, July 7). *What is Multimedia Data?* DEV Community. <https://dev.to/dimanikulin/what-is-multimedia-data-111f>.
- Papers with Code - Image Captioning*. (n.d.). <https://paperswithcode.com/task/image-captioning>.
- Raikote, P. (2021, May 18). *Object Detection – Part 1: Introduction*. *Applied Singularity*. <https://appliedsingularity.com/2021/05/18/object-detection/>.
- Sivic, & Zisserman. (2003). Video Google: a text retrieval approach to object matching in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2. <https://doi.org/10.1109/iccv.2003.1238663>.
- Smeulders, A., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12), 1349–1380. <https://doi.org/10.1109/34.895972>.
- Varun. (2023, May 9). What is Face Detection? Ultimate Guide 2023 + Model Comparison. *LearnOpenCV – Learn OpenCV, PyTorch, Keras, Tensorflow with examples and tutorials*. <https://learnopencv.com/what-is-face-detection-the-ultimate-guide/>.
- Vinyals, O., Toshev, A., Bengio, S., Erhan, D., & Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2015). Show and Tell: a neural image caption generator. *These CVPR 2015 Papers Are the Open Access Versions, Provided by the Computer Vision Foundation*.
- Wikipedia contributors. (2023, November 6). *Multimedia*. Wikipedia. <https://en.wikipedia.org/wiki/Multimedia>.
- Yak pratsiuiie OSINT-rozvidka? Vid biznes-analizu do oborony Ukrainy*. (2022, September 20). ISSP Training. <https://www.issp.training/post/yak-pratsyuye-osint-rozvidka-vid-biznes-analizu-do-oborony-ukrainy>.
- Zou, Z., Chen, K., Shi, Z., Guo, Y., & Guo, Y. (2023). Object Detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3), 257–276. <https://doi.org/10.1109/jproc.2023.3238524>.

*B. Pukhalskyi, O. Kyriienko*

## SYSTEMS FOR PROCESSING UNSTRUCTURED MULTIMEDIA INFORMATION

*This paper presents an overview of image search systems and the methods used to solve computer vision problems. The first part focuses on the needs of users and the requirements for image search systems. Existing systems, such as Google Photos, Apple Photos, Amazon Photos, Microsoft Photos, and Flickr, are then considered. Their advantages and disadvantages are evaluated, and proposed solutions to the identified shortcomings are discussed. Proposed computer vision methods, including object detection (face detection) and image text description, are analyzed in detail.*

*The rapid growth of digital images has led to an increasing need for effective image search systems that can effectively meet the needs of users. The research presented in this paper aims to provide a detailed overview of image search systems and the methods used to solve computer vision problems by improving search and indexing capabilities.*

*The main purpose of this research is to analyze the needs of users and the requirements for image search systems, to identify key aspects that contribute to effective and convenient operation. By understanding these requirements, we can evaluate existing systems available on the market and study their strengths and weaknesses. Specifically, we will focus on well-known image search systems, including Google Photos, Apple Photos, Amazon Photos, Microsoft Photos, and Flickr. These platforms represent a variety of approaches to image search and offer valuable information about the current state of the industry. In addition, our research will contribute to the discovery of new ways to apply computer vision to object detection.*

*The results of this work can serve as a valuable resource for researchers, developers, and professionals in the field who are involved in the development and implementation of image search systems. The proposed solutions and conclusions obtained as a result of this research can guide future advancements in the field.*

**Keywords:** image search systems, computer vision, object detection, image text description.

*Матеріал надійшов 25.06.2023*



Creative Commons Attribution 4.0 International License (CC BY 4.0)

Нагнибіда А. А.

## СТАНДАРТИ ПОДАННЯ НАВЧАЛЬНИХ МАТЕРІАЛІВ У ПРОГРАМНИХ СИСТЕМАХ ПІДТРИМКИ ЕЛЕКТРОННОЇ ОСВІТИ

*У контексті стрімкого росту цифрової освіти стандарти електронного навчання відіграють ключову роль у структуризації, відстеженні та управлінні навчальним контентом. Ця стаття розглядає основні стандарти для представлення навчальних матеріалів у програмних системах електронного навчання, з акцентом на стандартах SCORM, xAPI, IMS Global Learning Consortium та IEEE Learning Technology Standards Committee (LTSC). Докладно розглянуто основні аспекти, як-от технічні специфікації, методи впровадження і підходи до структуризації метаданих, щоб висвітлити значущість і вплив цих стандартів на сферу електронного навчання.*

**Ключові слова:** цифрова освіта, стандарти електронного навчання, SCORM, xAPI, IMS Global Learning Consortium, IEEE Learning Technology Standards Committee (LTSC), технічні специфікації електронного навчання, програмні системи електронного навчання, структуризація навчального контенту.

### Вступ

Перехід до цифрової освіти стрімко активізується, оскільки все більше навчальних закладів упроваджують стандарти електронного навчання для ефективного та результативного поширення інформації та навчальних матеріалів. Стандарти електронного навчання полегшують доставлення, відстеження та управління навчальним контентом у різних системах, а отже, сприяють гнучкості та керованості навчального процесу.

За допомогою стандартів електронної освіти (e-learning) навчальні заклади можуть оптимізувати процес створення та доставлення контенту, заощаджуючи час і ресурси, а також підвищуючи залучення учнів. У цій статті ми розглянемо чотири ключові стандарти, які використовують для представлення навчальних матеріалів у програмних системах електронного навчання: SCORM, xAPI, IMS Global Learning Consortium та IEEE Learning Technology Standards Committee (LTSC).

### IEEE Learning Technology Standards Committee (LTSC)

IEEE Learning Technology Standards Committee (LTSC) — це підрозділ Асоціації стандартів IEEE, який відповідає за розроблення технічних стандартів для навчальних техноло-

гій. Метою IEEE LTSC є створення всеосяжного набору стандартів для підтримки впровадження технологій навчання, які є сумісними, довговічними та масштабованими. Стандарти охоплюють широкий спектр тем, зокрема метадані об'єктів контенту, інформацію про учнів, пакети навчального контенту та моделювання даних. Сфера застосування цих стандартів поширюється на різні аспекти електронного навчання, від створення і доставлення навчальних матеріалів до відстеження прогресу учнів [3].

Технічні стандарти, розроблені IEEE LTSC, стосуються різних аспектів електронного навчання. Наприклад, стандарт метаданих навчальних об'єктів (Learning Object Metadata, LOM) надає схему для опису навчального контенту, що полегшує зберігання, пошук і обмін навчальними матеріалами. Стандарт Reusable Competency Definitions (RCD) дає змогу описувати компетенції та обмінюватися ними між системами. Стандарти, як правило, орієнтовані на технологію, більше зосереджуючись на структурі, організації та описі навчального контенту і даних. Вони використовують різні технології для опису даних і обміну ними, зокрема XML і RDF [5].

Упровадження стандартів IEEE LTSC зазвичай передбачає адаптацію або розроблення систем чи контенту електронного навчання відповідно до специфікацій відповідного стандарту. Це може передбачати структурування метаданих

навчального контенту відповідно до стандарту LOM або моделювання даних про учнів відповідно до стандарту Learner Information Package. Розуміння технічних специфікацій і вимог стандарту має вирішальне значення для правильного впровадження. Процеси тестування та валідації часто необхідні для того, щоб переконатися, що стандарт було правильно застосовано і що система або контент сумісні з іншими системами, які дотримуються того самого стандарту.

Стандарти IEEE LTSC мають кілька переваг. Вони надають комплексний набір рекомендацій для різних аспектів електронного навчання, гарантуючи, що навчальний контент і системи структуровані так, щоб підтримувати широке впровадження технологій і масштабованість. Це може спростити створення, поширення та управління навчальними матеріалами на різних платформах і системах. Стандарти також сприяють узгодженості в моделюванні та описі навчальних даних, підвищуючи їхню якість і зручність використання. Крім того, надаючи загальні рамки для структурування та опису навчального контенту і даних, ці стандарти можуть полегшити обмін і повторне використання навчальних матеріалів, сприяючи співпраці та підвищенню ефективності [4].

З іншого боку, впровадження стандартів IEEE LTSC може бути складним і потребувати значних технічних знань. Вони не завжди можуть бути сумісними з іншими системами або контентом, що передбачає значні модифікації або перероблення. Необхідність сумісності з цими стандартами може також обмежувати гнучкість розробників систем електронного навчання в розробленні та впровадженні своїх систем. Крім того, як і інші стандарти електронного навчання, стандарти IEEE LTSC не є загальноприйнятими, що може обмежити їхню ефективність у сприянні інтероперабельності.

Стандарти IEEE LTSC успішно застосовують у різних контекстах електронного навчання. Наприклад, університети використовують стандарт метаданих навчальних об'єктів (LOM) для організації своїх цифрових навчальних ресурсів, що сприяє ефективному зберіганню, пошуку та обміну цими ресурсами. Стандарт Reusable Competency Definitions (RCD) було використано у корпоративних навчальних програмах для стандартизації опису компетенцій, впорядкування оцінки навичок співробітників і адаптації навчальних програм. Ці приклади ілюструють, як стандарти IEEE LTSC можуть покращити організацію, управління та ефективність електронного навчання [6].

Порівняно зі стандартами SCORM, xAPI та IMS Global Learning Consortium, стандарти IEEE LTSC надають всеохопну і детальнішу основу для опису й організації навчального контенту та даних. SCORM і xAPI більше зосереджені на відстеженні та наданні навчального досвіду, стандарти IMS Global — на системній інтеграції та інтероперабельності, натомість стандарти IEEE LTSC акцентують увагу на структурі та описі навчального контенту і даних про учнів [1]. Тож ці стандарти доповнюють інші стандарти е-навчання, кожен з яких стосується різних аспектів цього процесу, а разом вони створюють всеосяжну основу для впровадження технологій е-навчання.

Оскільки сфера електронного навчання розширюється і розвивається, буде існувати постійна потреба в надійних стандартах, якими можна керуватися під час розробки та впровадження технологій навчання. Детальний і всеосяжний характер стандартів IEEE LTSC свідчить про те, що вони і надалі залишатимуться важливими в цьому сенсі. Однак, щоб ці стандарти залишалися актуальними, їх необхідно постійно оновлювати, щоб відображати досягнення в технологіях навчання і педагогіці. Враховуючи динамічний характер сфери електронного навчання, завдання підтримки та оновлення цих стандартів буде складним, але вкрай важливим.

Отже, стандарти IEEE LTSC забезпечують цінну основу для структурування та опису контенту і даних електронного навчання. Їх впровадження може підвищити узгодженість, стабільність і масштабованість систем електронного навчання. Хоча стандарти передбачають технічну експертизу для впровадження і потребують оновлення, щоб залишатися актуальними, вони пропонують значні потенційні переваги для сфери електронного навчання. Разом з іншими стандартами е-навчання вони формують комплексний набір керівних принципів для розроблення та впровадження ефективних і результативних технологій е-навчання.

### **IMS Global Learning Consortium Standards**

IMS Global Learning Consortium — це міжнародна неприбуткова організація, відповідальна за створення та просування світових стандартів для освітніх і навчальних технологій. Стандарти, які вона розробляє, покликані забезпечити кращу інтеграцію та інтероперабельність навчальних систем і контенту, поліпшуючи навчальний процес. Вони охоплюють широкий спектр аспектів, пов'язаних з електронним на-

вчанням, зокрема доступність навчальних інструментів і цифрового контенту, академічні інформаційні послуги, навчальну аналітику. Мета цих стандартів — забезпечити безперешкодний і ефективний обмін даними і контентом між різними навчальними платформами, підтримуючи інноваційне та ефективне цифрове навчання.

Стандарти IMS Global Learning Consortium охоплюють широкий спектр технічних специфікацій, спрямованих на різні аспекти електронного навчання. Наприклад, стандарт Learning Tools Interoperability (LTI) полегшує інтеграцію навчальних додатків із такими платформами, як системи управління навчанням, портали та інші освітні середовища. Стандарт OneRoster спрямований на безпечний і точний обмін інформацією про списки студентів, навчальні матеріали та оцінки між системами. Ці стандарти покладаються на відкриті технології, як-от XML і JSON, для обміну даними і використовують стандартні протоколи HTTPS для безпечного зв'язку.

Упровадження моделі IMS Global Learning Consortium передбачає приведення певної платформи або контенту для електронного навчання у відповідність до специфікацій конкретного стандарту. Наприклад, для впровадження стандарту LTI необхідно розробити або адаптувати навчальний додаток для взаємодії з LMS юзерами за допомогою протоколу LTI. Це передбачає обмін обліковими даними безпеки, надсилання запитів на запуск LTI та оброблення відповідей на запуск певних функцій. Конкретний метод реалізації залежить від кожного стандарту та його призначення. Важливо ретельно вивчити технічні специфікації відповідного стандарту і, в багатьох випадках, взяти участь у процесі тестування та сертифікації, запропонованих IMS Global Learning Consortium, щоб забезпечити правильне впровадження.

Стандарти від IMS Global Learning Consortium мають кілька переваг. Вони полегшують сумісність між різними навчальними системами й інструментами, забезпечуючи безперебійний обмін даними та інтеграцію. Це підвищує ефективність операцій електронного навчання та зменшує час і ресурси, необхідні для управління кількома системами. Стандарти також підтримують інтеграцію інноваційних освітніх технологій в інші системи, сприяючи розвитку креативності та покращенню навчального досвіду. Крім того, стандарти надають пріоритет безпеці та конфіденційності даних, що є надзвичайно важливими в епоху цифрового навчання.

Однак є певні труднощі, пов'язані зі впровадженням цих стандартів. По-перше, для розуміння і правильного застосування специфікацій потрібен високий рівень технічної обізнаності. Невеликі установи можуть зіткнутися з проблемами, пов'язаними з ресурсами, необхідними для впровадження. По-друге, GLC стандарти не є загальноприйнятими, а це означає, що не всі системи та інструменти можуть бути сумісними. Нарешті, хоча стандарти сприяють інтеграції та інтероперабельності, вони не забезпечують якість або ефективність навчального контенту чи досвіду, якими треба керувати окремо.

IMS Global Learning Consortium стандарти успішно впроваджують у різних умовах. Наприклад, університети використовували стандарт LTI для інтеграції зовнішніх навчальних інструментів, як-от програми для перевірки на плагіат і відеоконференції, у свої системи управління навчанням, покращуючи навчальний процес і підвищуючи операційну ефективність. У деяких американських школах стандарт OneRoster використовують для автоматизації та захисту обміну даними про учнів між системами, зменшуючи адміністративний тягар і підвищуючи точність даних. Ці приклади використання підкреслюють, як стандарти можуть сприяти інтеграції, повноті впровадження та ефективності систем електронного навчання.

На відміну від SCORM та xAPI, IMS Global Learning Consortium стандарти більше зосереджені на інтеграції і розгортанні систем та інструментів, а не на відстеженні та доставленні навчального контенту. SCORM і xAPI передусім стосуються того, як структурується і реєструється навчальний досвід, а стандарти IMS Global забезпечують технічну основу для різноманітних систем і інструментів електронного навчання для ефективної і безпечної комунікації. Вони також охоплюють ширше коло тем, зокрема аналітику навчання, доступність цифрових ресурсів і безпечний обмін даними.

Оскільки цифрове навчання і надалі розвивається, потреба в ефективній інтероперабельності та інтеграції між різними навчальними системами та інструментами тільки зростатиме. Це свідчить про багатообіцяльне майбутнє стандартів Глобального навчального консорціуму IMS. Однак їхній успіх залежатиме від ширшого прийняття та постійного оновлення стандартів, щоб іти в ногу з розвитком технологій навчання. Очікують, що ці стандарти і надалі відіграватимуть ключову роль у формуванні широкої сфери вжитку та функціональності екосистем електронного навчання.

## Sharable Content Object Reference Model (SCORM)

SCORM — один із найпоширеніших стандартів і специфікацій для систем електронного навчання, що дає змогу розробникам створювати навчальні об'єкти багаторазового використання. Мета SCORM — підвищити сумісність навчальних матеріалів у різних системах електронного навчання. Він забезпечує стандартизовану основу для пакування контенту в такий спосіб, щоб його могла зрозуміти і використовувати будь-яка SCORM-сумісна система управління навчанням (LMS). Сфера застосування SCORM поширюється від простого доставлення контенту до складніших навчальних сценаріїв, що охоплюють оцінювання учнів, відстеження прогресу та персоналізовані навчальні моделі.

Технічні характеристики SCORM ґрунтуються на його інтероперабельності, повторному використанні та адаптивності. По-перше, інтероперабельність SCORM гарантує, що будь-який SCORM-сумісний контент може бути інтегрований у різні SCORM-сумісні LMS без будь-яких змін. По-друге, вона сприяє багаторазовому використанню, полегшуючи впровадження того самого навчального контенту в різних системах і курсах. Нарешті, адаптивність SCORM забезпечується елементами керування послідовністю та навігацією, які дають змогу персоналізувати навчальні моделі на основі успішності та потреб учня. SCORM також надає можливості для відстеження прогресу учнів і зберігання таких даних, як статус завершення курсу, витрачений час і оцінки, тим самим допомагаючи навчальним закладам ефективно контролювати і оцінювати успішність учнів.

Впровадження SCORM у платформу електронного навчання передбачає низку кроків. Першим кроком є створення або розгортання навчального контенту, сумісного зі стандартом SCORM. Різні авторські інструменти, як-от Adobe Captivate або Articulate Storyline, можуть допомогти у створенні пакетів SCORM. Ці пакети складаються з комбінації HTML, JavaScript і мультимедійних файлів, що супроводжуються файлом XML, який описує структуру і метадані курсу. Потім SCORM-пакет завантажується в SCORM-сумісну систему управління навчанням (LMS), яка може інтерпретувати і відображати вміст SCORM для учнів, відстежувати їхній прогрес і зберігати дані про їхню успішність.

Модель SCORM має кілька переваг, які роблять її популярним вибором серед навчальних закладів. Її здатність безперешкодно працювати

на різних платформах забезпечує широке охоплення та легку доступність навчального контенту. Вона також підвищує економічну ефективність електронної освіти, оскільки той самий контент можна використовувати багаторазово на різних платформах. Крім того, можливості відстеження SCORM дають змогу навчальним закладам контролювати прогрес і успішність студентів, сприяючи персоналізованому навчанням і прийняттю рішень на основі отриманих даних. Нарешті, підтримка SCORM як онлайн-, так і офлайн-навчання гарантує, що воно може відбуватися в будь-який час і в будь-якому місці, тим самим підвищуючи гнучкість і зручність академічного процесу.

Окрім численних переваг, модель SCORM має і недоліки. Її технічна складність може стати проблемою для творців контенту, які не знайомі з її специфікаціями та мовою сценаріїв. Хоча SCORM сприяє інтероперабельності, все ж можуть виникати проблеми сумісності з різними її версіями або між SCORM та іншими стандартами електронного навчання. Більше того, можливості відстеження SCORM хоч і корисні, але обмежені відстеженням базових показників, таких як статус завершення і бали. Модель не може ефективно відстежувати більш складні види навчальної діяльності, наприклад групові проекти або інтерактивні симуляції. До того ж, підтримка та оновлення контенту, сумісного зі SCORM, можуть бути затратними у фінансовому плані, що потенційно обмежує її впровадження в невеликих установах з обмеженими ресурсами.

Модель SCORM успішно використовують у різних академічних ланках. Наприклад, університети та коледжі застосовують її для проведення онлайн-курсів, надаючи студентам доступ до того самого контенту з будь-якої LMS, сумісної з SCORM. Корпоративні навчальні відділи за допомогою SCORM проводять і відстежують Compliance-тренінги, що є критично важливим для компаній, які працюють у галузях із високим рівнем регулювання. Крім того, урядові установи використовують SCORM для забезпечення послідовного навчання співробітників у різних місцях. Ці приклади використання підкреслюють універсальність SCORM і переваги її інтероперабельності та можливостей відстеження.

Порівняно з іншими стандартами, SCORM має унікальні сильні та слабкі сторони. На відміну від SCORM, xAPI (Experience API) може відстежувати навчальний досвід як онлайн, так і офлайн, а також із різних пристроїв, надаючи більш змістовні дані про діяльність учнів. Однак SCORM простіший у впровадженні та ширше



підтримується наявними системами управління навчанням (LMS). Стандарти IMS Global Learning Consortium і IEEE LTSC більше зосереджені на доступності, інтероперабельності та довгостроковій стійкості цифрових навчальних ресурсів. Хоча ці стандарти пропонують більш цілісний підхід, їм бракує широкого впровадження SCORM і широкої підтримки серед інструментів і платформ електронного навчання [7].

Із розвитком технологій і методологій навчання SCORM також потребує адаптації. Фокус поступово змінюється від SCORM до новіших стандартів, таких як xAPI, які пропонують більш комплексні можливості відстеження та аналітики. Однак, зважаючи на широке впровадження та підтримку, SCORM, імовірно, залишиться актуальним у найближчому майбутньому. Нині докладають зусиль для модернізації SCORM із метою усунення його обмежень, покращення можливостей та забезпечення сумісності з новими технологіями й освітніми тенденціями.

### **Experience API, також відомий як Tin Can API (xAPI)**

Стандарт Experience API, також відомий як xAPI або Tin Can API, — це специфікація технології навчання, яка дає змогу записувати, відстежувати та аналізувати навчальний досвід як онлайн, так і офлайн. Ба більше, стандарт xAPI було розроблено для подолання деяких обмежень SCORM, зокрема можливості відстежувати ширший спектр навчальної діяльності. Метою xAPI є фіксація досвіду учня (або групи учнів) у послідовному форматі, що дає можливість аналізувати дані для розуміння прогресу та ефективності навчання. Його сфера застосування не обмежується традиційними середовищами онлайн-навчання; стандарт може відстежувати академічний процес із багатьох джерел, це зокрема мобільні пристрої, симуляції, віртуальна реальність і навіть діяльність у реальному світі.

Академічний стандарт xAPI працює, записуючи дані про досвід у формі «Діяч-Дієслово-Об'єкт», наприклад, «Богдан пройшов тренінг із техніки безпеки». Потім ці твердження надсилаються до сховища навчальних записів (Learning Record Store, LRS) — спеціалізованої системи зберігання даних, яка може бути частиною LMS або працювати окремо. LRS фіксує, зберігає та витягує ці навчальні записи, роблячи їх доступними для аналізу. Модель xAPI використовує стандартний HTTPS API та формат даних JSON, що робить її сумісною з широким спектром технологій і платформ.

Упровадження xAPI передбачає інтеграцію навчального контенту з підтримкою xAPI та сховища навчальних записів (LRS). Академічний контент може бути курсом, відео, грою, симуляцією або навіть таким видом діяльності, як відвідування конференції. xAPI-виклики, що генеруються в результаті цієї навчальної діяльності, потім зберігаються в LRS. Це можна зробити безпосередньо або через LMS, якщо доступна підтримка xAPI. Згодом збережені дані можна отримати та проаналізувати за допомогою інструментів звітності та аналітики, щоб отримати уявлення про успішність учня та ефективність навчального процесу. Гнучкість xAPI дає змогу впроваджувати таку модель різними способами, щоб відповідати різним навчальним середовищам і вимогам [2].

Основна перевага стандарту xAPI полягає в його здатності фіксувати широкий спектр навчального досвіду, що виходить за рамки можливостей SCORM. Він може відстежувати як онлайн-, так і офлайн-діяльність, наприклад читання книжки, відвідування семінару або взаємодію з симуляцією. Крім того, використання сховища навчальних записів у xAPI дає можливість більш детально відстежувати та аналізувати навчальні дані, уможливаючи персоналізоване навчання та прийняття рішень на основі даних. Ще однією значною перевагою стандарту xAPI є його гнучкість; xAPI не залежить від платформи і може безперешкодно працювати на різних пристроях і технологіях. Це дає учням свободу навчатися з різних джерел і контекстів, при цьому їхній прогрес відстежується і аналізується.

Однак xAPI також має деякі недоліки. Впровадження xAPI може бути складнішим, ніж інших стандартів, як-от SCORM, через потребу в окремому сховищі навчальних записів. Крім того, хоча гнучкість і масштабність xAPI є перевагами, ці два параметри також можуть призвести до суперечливості даних, якщо ними не керувати належним чином. Існує також ризик генерування занадто великої кількості даних, що ускладнює фільтрацію та аналіз найбільш значущих аспектів. До того ж, xAPI не настільки поширений, як SCORM, а це означає, що не всі системи управління навчанням підтримують xAPI, що може обмежити його використання в деяких випадках.

Стандарт xAPI ефективно використовують у різних сферах. У корпоративному секторі його впроваджують для відстеження ефективності роботи на робочому місці, надаючи уявлення про те, як навчання впливає на ефективність

роботи. Наприклад, тренінг із симуляції продажів, який відстежується за допомогою xAPI, може надати дані про те, наскільки добре слухачі застосовують отримані знання в реальних ситуаціях. В освітньому секторі xAPI може відстежувати взаємодію студентів із різними ресурсами як в аудиторії, так і за її межами. Наприклад, взаємодію студента з навчальним додатком на планшеті можна відстежувати паралельно з його діяльністю в класі, щоб отримати повну картину його навчання. Ці приклади використання демонструють здатність xAPI фіксувати багатий, різноманітний навчальний досвід і перетворювати його на змістовні дані.

Порівняно зі SCORM, xAPI пропонує ширші можливості для відстеження навчального досвіду, зокрема під час офлайн- і неформального навчання. SCORM зосереджується насамперед на доставленні контенту в LMS, натомість xAPI дає змогу відстежувати академічні активності на різних платформах і пристроях. На відміну від стандартів IMS Global Learning Consortium та IEEE LTSC, xAPI не лише наголошує на інтероперабельності, а й надає стандартизований метод для збирання та зберігання широкого спектра навчальних даних, забезпечуючи більш глибоке розуміння досвіду учнів.

Майбутнє xAPI видається багатообіцяльним, оскільки прискорюється перехід до більш різноманітних, мультиплатформних і орієнтованих на учня освітніх послуг. Оскільки технології продовжують розвиватися, а навчання виходить за межі класу і переходить у реальну діяльність, потреба у гнучкому та всебічному відстеженні, яке забезпечує xAPI, лише зростатиме. Однак для того, щоб стандарт xAPI повністю реалізував свій потенціал, необхідне широке впровадження і стандартизація на різних навчальних платформах і пристроях. Подальший розвиток, спрямований на усунення поточних обмежень, таких як узгодженість даних та їх надмірна кількість, також матиме вирішальне значення.

### **Висновки та шляхи імплементації стандартів електронного навчання**

У сфері електронного навчання стандарти відіграють важливу роль у забезпеченні масштабованості та ефективності академічних процесів. Чотири розглянуті стандарти — SCORM, xAPI, IMS Global Learning Consortium та IEEE LTSC — мають свої унікальні переваги та недоліки.

SCORM — це усталений стандарт, який дає змогу упаковувати та відстежувати навчальний контент у LMS, однак він має обмеження у від-

стеженні навчального досвіду за межами системи. xAPI, навпаки, чудово підходить для відстеження різноманітного навчального досвіду на різних платформах, пропонуючи детальну аналітику навчання, однак його впровадження може бути складним.

Стандарти IMS Global Learning Consortium надають широкий спектр стандартів, що сприяють ефективній інтеграції та інтероперабельності між різними навчальними системами та інструментами, незважаючи на вимогу технічної експертизи для впровадження. Нарешті, стандарти IEEE LTSC пропонують комплексний набір рекомендацій щодо структурування та організації навчального контенту і даних, забезпечуючи високий рівень керованості та масштабованості, хоча і не без складнощів у впровадженні.

Стандарти IMS Global Learning Consortium відіграють життєво важливу роль у забезпеченні інтероперабельності та інтеграції різних систем та інструментів електронного навчання. Хоча впровадження цих стандартів потребує технічних знань і ресурсів, підвищена ефективність, безпека і гнучкість, які вони забезпечують, є ключовими перевагами. Оскільки сфера електронного навчання продовжує зростати і розвиватися, ці стандарти, ймовірно, залишатимуться важливим інструментом у формуванні ефективних і результативних цифрових навчальних середовищ.

SCORM є цінним стандартом для представлення навчальних матеріалів у системах електронного навчання. Завдяки інтероперабельності, можливості багаторазового використання та відстеження він є популярним вибором серед освітян і провайдерів навчання. Однак його технічна складність і обмежені можливості відстеження підкреслюють необхідність постійного вдосконалення та адаптації. Незважаючи на появу нових стандартів, широке впровадження SCORM свідчить про те, що він й надалі відіграватиме значну роль в електронному навчанні. Як і у випадку з будь-якою технологією, успіх залежить від того, наскільки добре вона адаптується до мінливих освітніх потреб і технологій.

Впровадження xAPI стандарту може бути складнішим, ніж інших моделей, однак його переваги роблять його переконливим вибором для закладів, які прагнуть забезпечити цілісний, орієнтований на учня освітній досвід. Оскільки середовище електронного навчання продовжує розвиватися, можливості xAPI роблять його добре пристосованим для задоволення мінливих потреб як учнів, так і викладачів.

Усі ці стандарти вказують на перспективні напрями, що відповідають розвитку технологій і змінам у педагогічних стратегіях. Вони мають постійно розвиватися, щоб відповідати новим

технологіям і підходам до навчання. Вибір правильного стандарту значною мірою залежить від конкретних потреб і цілей навчального закладу або організації.

#### Список літератури

1. Downes S. E-learning 2.0 / S. Downes // eLearn Magazine. — 2005. — Issue 10.
2. Duval E. Metadata principles and practicalities / E. Duval, W. Hodgins, S. Sutton, & S. L. Weibel // D-Lib Magazine. — 2002. — Vol. 8, no 4.
3. Greenberg, J. Metadata extraction and harvesting: a comparison of two automatic metadata generation applications / J. Greenberg // Journal of Internet Cataloging. — 2004. — Vol. 6. — Pp. 59–82.
4. IEEE Learning Technology Standards Committee. Draft Standard for Learning Object Metadata. — Piscataway, NJ : IEEE, 2002. — 10 p.
5. Koper R. Modeling units of study from a pedagogical perspective: the pedagogical metamodel behind EML / R. Koper. — Open University of the Netherlands, 2003. — 15 p.
6. Polsani P. R. Use and abuse of reusable learning objects / P. R. Polsani // Journal of Digital information. — 2003.
7. Wiley D. A. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy / D. A. Wiley. — The Instructional Use of Learning Objects : Online Version, 2000.

#### References

- Downes, S. (2005). E-learning 2.0. *eLearn Magazine*, 10.
- Duval, E., Hodgins, W., Sutton, S., & Weibel, S. L. (2002). Metadata principles and practicalities. *D-Lib Magazine*, 8 (4), 16.
- Greenberg, J. (2004). Metadata extraction and harvesting: a comparison of two automatic metadata generation applications. *Journal of Internet Cataloging*, 6 (4), 59–82.
- IEEE Learning Technology Standards Committee. (2002). *Draft Standard for Learning Object Metadata*. IEEE.
- Koper, R. (2003). *Modeling units of study from a pedagogical perspective: the pedagogical metamodel behind EML*. Open University of the Netherlands.
- Polsani, P. R. (2003). Use and abuse of reusable learning objects. *Journal of Digital information*, 3 (4).
- Wiley, D. A. (2000). *Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy*. In The Instructional Use of Learning Objects: Online Version.

A. Nahnybida

## STANDARDS FOR PRESENTING EDUCATIONAL MATERIALS IN ELECTRONIC EDUCATION SUPPORT SYSTEMS

*As digital education experiences rapid growth, an increasing number of institutions are adopting e-learning standards for effective and efficient dissemination of information and educational materials. E-learning standards simplify content delivery, tracking, and management across various systems, thereby fostering flexibility and control in the learning process. Institutions can harness these standards to optimize content creation and delivery processes, saving time and resources, and enhancing student engagement.*

*This article delves into four pivotal e-learning standards employed for presenting educational content in electronic learning systems: SCORM, xAPI, the IMS Global Learning Consortium standards, and the IEEE Learning Technology Standards Committee (LTSC). The IEEE LTSC, a division of the IEEE Standards Association, is responsible for crafting technical benchmarks for learning technologies. Its primary goal is to curate a comprehensive suite of standards to support the implementation of learning technologies that are compatible, enduring, and scalable. These standards encompass a broad spectrum of topics, including content object metadata, learner information, educational content packages, and data modeling. Their application spans various e-learning facets, ranging from content creation and delivery to tracking student progress.*

*Implementing IEEE LTSC standards typically entails adapting or developing e-learning systems or content in line with the respective standard's specifications. This can involve structuring learning content metadata per the LOM standard or modeling learner data according to the Learner Information Package standard. Proper comprehension of technical specifications and standard requirements is paramount for accurate implementation. Testing and validation processes are often essential to ensure correct standard application and system or content compatibility with other systems adhering to the same standard.*

**Keywords:** digital education, e-learning standards, SCORM, xAPI, IMS Global Learning Consortium, IEEE Learning Technology Standards Committee (LTSC), e-learning technical specifications, e-learning software systems, structuring of learning content.

Матеріал надійшов 20.08.2023



## НОВА ВЕРСІЯ СЕРЕДОВИЩА ВИКОНАННЯ MATHPAR-DAP

У цій статті згадано основні особливості середовища виконання децентралізованого управління розподіленими обчисленнями DAP (Drop-Pine-Amine), які було опубліковано в [4]. Головною метою цієї статті є опис нових функціональних можливостей, які з'явилися в останньому випуску. Як приклад алгоритму з блоковою рекурсією описано факторизацію Холецького симетричної позитивно означеної матриці у вигляді блокового дихотомічного алгоритму. Результати експериментів демонструють гарну масштабованість запропонованого рішення. Запропоновано розвивати співпрацю у цій науковій сфері. Розроблений програмний пакет відкритий для спільного розроблення, його можна вільно використовувати для наукових і освітніх цілей.

**Ключові слова:** розподілені обчислення, паралельне програмування, середовище виконання, OpenMPI.

### Вступ

Сучасні суперкомп'ютерні системи, що містять сотні тисяч ядер, стикаються з труднощами в організації паралельних обчислень (див. [1]). Основними проблемами є нерівномірне навантаження на апаратні засоби, накопичення помилок під час обчислень із великими матрицями та можливі відмови ядер під час процесу обчислень.

Нещодавно було розроблено універсальну схему виявлення динамічних задач (DTD) для середовища виконання PaRSEC [2; 3]. Це середовище може підтримувати системи з розділеною та спільною пам'яттю. Цей новий парадигм показав кращу продуктивність порівняно з параметризованим плануванням задач, яке використовувалося раніше.

У [4] описано нове середовище виконання для суперкомп'ютерів із розділеною пам'яттю. Воно призначене для вирішення матричних проблем за допомогою алгоритмів із блоковою рекурсією. Його головною перевагою є забезпечення ефективного обчислювального процесу та гарної масштабованості програм як для розріджених, так і для щільних матриць на кластері з розділеною пам'яттю. Ще однією перевагою є можливість реорганізації обчислювального процесу у випадку відмови окремих вузлів під час обчислень.

### Приклад: факторизація Холецького

Дано симетричну позитивно означену матрицю  $A$ . Потрібно знайти нижню трикутну матрицю  $L$ , таку що  $A = LL^T$ .

Запропоновано дихотомічний блоковий рекурсивний алгоритм для факторизації Холецького. Обчислювальна складність всього алгоритму така сама, як у алгоритму множення матриць, який буде використовуватися для блокового множення.

Позначимо блоки матриці як  $\alpha$ ,  $\beta$ ,  $\beta^T$ ,  $\gamma$  а блоки матриці  $L$ , які потрібно знайти, позначимо як  $a$ ,  $0$ ,  $b$ ,  $c$ ; і прирівняємо добуток  $LL^T$  до  $A$ :

$$L \times L^T = \begin{pmatrix} a & 0 \\ b & c \end{pmatrix} \times \begin{pmatrix} a^T & b^T \\ 0 & c^T \end{pmatrix} = \\ = \begin{pmatrix} aa^T & ab^T \\ ba^T & bb^T + cc^T \end{pmatrix} = \begin{pmatrix} \alpha & \beta \\ \beta^T & \gamma \end{pmatrix} = A.$$

Із цього випливає рівність для блоків:

$$aa^T = \alpha, b = \beta^T (a^{-1})^T, cc^T = \gamma - bb^T.$$

Для обчислення блоків  $a$ ,  $b$  та  $c$  потрібно виконати блокове множення, транспонування, обернення та факторизацію Холецького для двох блоків.

Обчислення оберненої матриці можна уникнути, якщо обчислювати обернену матрицю  $L$  разом із матрицею  $L^{-1}$ .

---

#### Algorithm 1 Standard blocked Cholesky factorization

---

- 1: for each panel left to right do
  - 2: Partition  $A = \begin{bmatrix} A_{11} & \\ A_{21} & A_{22} \end{bmatrix}$ , where  $A_{11}$  is  $\text{NB} \times \text{NB}$
  - 3: Factorize  $A_{11} = LL^T$  using unblocked algorithm
  - 4: Update panel  $A_{21} := A_{21}L^{-T}$  using triangular solver
  - 5: Update trailing matrix  $A_{22} := A_{22} - A_{21}A_{21}^T$  using symmetric rank- $k$  update
  - 6: Continue with  $A = A_{22}$
  - 7: end for
- 

Процедуру факторизації Холецького розширено таким чином, щоб вона повертала не тільки матрицю  $L$ , а й її обернену матрицю  $L^{-1}$ . Потім,

разом з блоками  $a$  та  $b$ , обернені блоки  $a^{-1}$  та  $b^{-1}$ . У цьому випадку обернена матриця до  $L$  матиме вигляд:

$$L^{-1} = \begin{pmatrix} a^{-1} & 0 \\ -c^{-1}ba^{-1} & c^{-1} \end{pmatrix}.$$

Граф цього рекурсивного алгоритму показано на рис. 1.

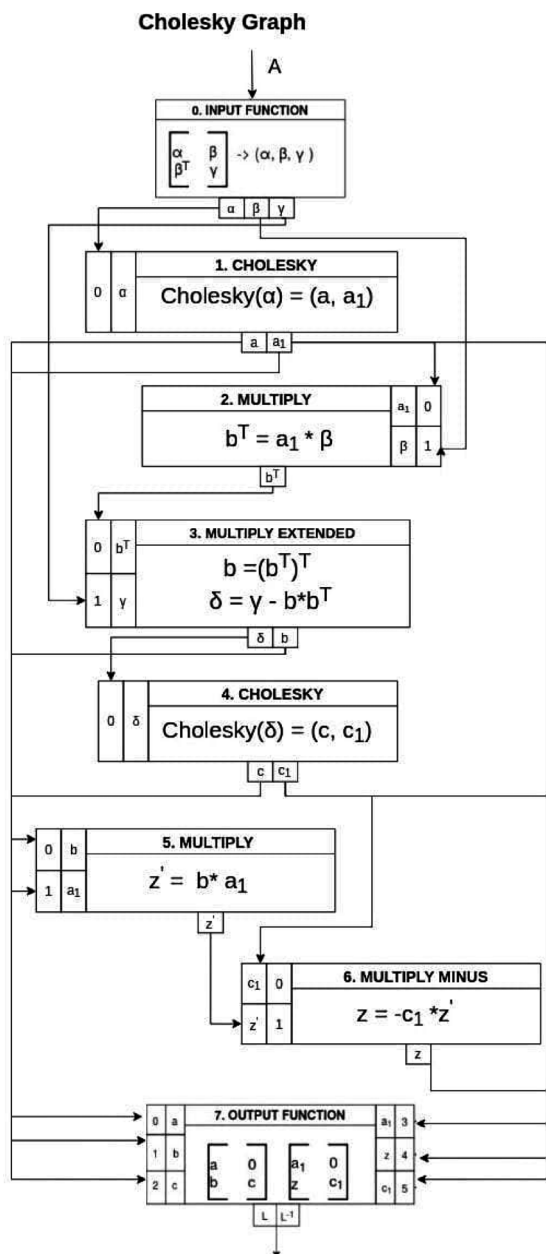


Рис. 1. Граф факторизації Холецького

На рисунку 2 наведено механізм управління обчислювальним процесом:

- Drop — це блоково-рекурсивна функція разом зі вхідними даними, яку можна передати на будь-який процесор;
- Amine — це набір усіх дропів першого рівня рекурсії деякої блоково-рекурсивної функції

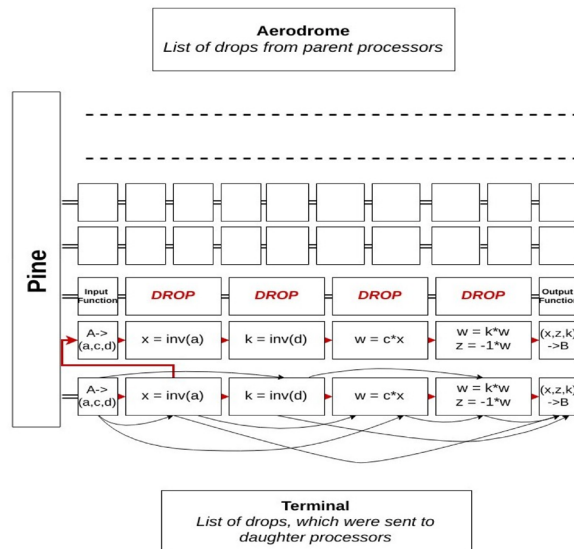


Рис. 2. Механізм управління обчислювальним процесом

та набір їхніх зв'язків. Щоб обчислити Drop, потрібно створити для нього Amine;

- Pine — список усіх Amine, які було створено на цьому процесорі;
- Aerodrome — список усіх дропів, які було передано на цей процесор;
- Terminal — список дропів, які було відправлено на інші процесори.

### Експерименти.

#### Алгоритм факторизації Холецького

У таблиці 1 подано результати трьох серій експериментів. У кожній серії розмір листового блоку (LS), тобто розмір блоку, який обчислювався на одному процесорі, був постійним. У першій серії він становив 64, у другій — 128, а в третій серії — 256.

У кожній серії підтримувалося постійне обчислювальне навантаження на одне ядро: коли розмір матриці подвоювався, кількість ядер збільшувалась у вісім разів. Отже, кількість ядер була 1, 8, 64 і 512, а розмір матриці становив 256, 512, 1024 і 2048. Використовувалися числа BigDecimal зі 100 десятковими цифрами у дробовій частині.

Експерименти показали, що найкращим листовим блоком для кластеру був 128.

З другого рядка табл. 1 видно, що коли кількість ядер збільшується у вісім разів, час обчислень збільшується у  $364/121=3.0$ ,  $896/364=2.4$  та  $1996/896=2.2$  рази відповідно. Отже, відповідний TLR становить  $[\sqrt[3]{3.0}, \sqrt[3]{2.4}, \sqrt[3]{2.2}] = [1.44, 1.34, 1.3]$ .

Ці дані вказують на гарну масштабованість запропонованого програмного рішення.

Обчислювальні експерименти проводили на кластері MVS-10P CASCADE LAKE (2256 ядер, Intel Xeon Platinum 8268, 28 ядер / 48 потоків, 2.9 ГГц, 35.75 МБ кеш-пам'яті, 96 ГБ ОЗП на процесор).

### Подолання проблеми синхронізації повідомлень із блокуванням

Під час відлагодження системи було виявлено проблему синхронізації повідомлень. Спочатку всі пересилання були неблокуючими. Однак за пересилання великих обсягів даних було виявлено переповнення буферів і руйнування масивів. Тому було прийнято рішення використовувати блокувальне пересилання для цих повідомлень.

Однак за таких умов постала проблема «мертвого замка». Почали виникати ситуації, коли декілька процесорів входили в команду відправлення повідомлень, але при цьому перший відправляв другому, другий — третьому і так далі, а останній відправляв першому. Очевидно, що програма зависала.

Для подолання цієї проблеми було введено певні правила. Усі повідомлення, крім тих, що відправляють великі об'єкти даних, можна відправляти в режимі iSend (неблокувальний режим). Для відправлення великих об'єктів даних необхідно використовувати блокувальні повідомлення. Однак для цього потрібно забезпечити, щоб повідомлення в блокувальному режимі не можна було відправити, якщо є ризик, що під час їхнього передавання є можливість отримати чиесь повідомлення в блокувальному режимі. Тому необхідно заборонити відправлення блокувальних повідомлень, коли очікується прийом блокувального повідомлення. Для цього достатньо відокремити час відправлення та прийому блокувальних повідомлень.

Для розділення цих двох подій (відправлення та приймання блокувального повідомлення) введено булеву змінну `recv`, яка має такі стани:

- `False` (за замовчуванням) — дозволено відправлення блокувальних повідомлень, але заборонено їх приймання;

- `True` — очікування на приймання блокувальних повідомлень, заборонено відправлення блокувальних повідомлень.

Стан змінної `recv` має бути синхронізовано на всіх процесорах. Для цього введено нові структури даних. Виняток становить кореневий процесор, який може відправляти блокувальні повідомлення без запиту. Також, якщо номер процесора парний, він може відправляти повідомлення без запиту на вільний процесор із непарним номером. У всіх інших випадках потрібен запит.

Ці винятки введено для прискорення виконання програми, уникаючи зависання.

Усі операції пересилання поділяють на блокувальні та неблокувальні. Неблокувальне пересилання (`iSend`) передбачає відправлення стану процесора, відправлення списку вільних процесорів, відправлення запиту на відправку блокувального повідомлення, відправлення підтвердження прийому блокувального повідомлення та відправлення відмови на відправку блокувального запиту. Блокувальне пересилання (`send`) — відправлення завдання видалення на вільні процесори та відправлення результату на батьківські процесори.

Для синхронізації режиму приймання-відправлення було введено додаткові команди для запиту дозволу на відправлення блокувального повідомлення та повідомлення про те, що процесор готовий прийняти блокувальне повідомлення. Для керування цими повідомленнями створено три списки:

- `waitingFromOthers` — список процесорів, які запитали дозвіл на відправлення блокувального повідомлення;
- `waitingOutput` — список процесорів, яким уже було відправлено запити на дозвіл відправити блокувальне повідомлення;
- `approvedOutput` — список процесорів, які вже дали згоду на відправлення блокувального повідомлення.

Стани змінної `recv` змінюються в таких випадках:

- `recv = false`: після прийому одного блокувального повідомлення — не дозволено приймати;

Таблиця 1

Масштабованість алгоритму Холецького для чисел **BigDecimal** зі 100 десятковими цифрами (час у секундах)

#core / matrix size	1/256	TLR	8/512	TLR	64/1024	TLR	512/2048
Time for leaf size 64	1.22 min	1.14	1.8 min	2.4	24.84 min	1.05	28.92 min
Time for leaf size 128	1.21 min	1.44	3.64 min	1.35	8.96 min	1.31	19.96 min
Time for leaf size 256	1.15 min	1.96	8.72 min	1.21	15.5 min	1.16	23.96 min

– `recv = true`: після відправлення підтвердження на запит на відправку блокувального повідомлення — заборонено відправляти.

### Висновки

У статті було описано середовище виконання DAP (drop-amine-pine). Основною особливістю DAP є його здатність послідовно розгортати функції в глибину, підтримуючи всі стани на будь-якому рівні вкладеності до завершення всіх обчислень у поточному обчислювальному піддереві. Такий підхід дає змогу будь-якому процесору вільно перемикатися між підзадачами без очікування завершення поточної підзадачі.

Важливою особливістю цього середовища виконання є захист від відмов деяких вузлів під

час обчислень. Батьківський вузол, який надіслав дроп (drop) своєму дочірньому вузлу, повинен отримати результат. Однак, замість результату, він може отримати повідомлення про статус дочірнього вузла. У таких випадках завдання цього дропа перенаправляється на альтернативний вузол. Додаткові зміни в інших вузлах не потрібні. В результаті буде втрачено та в подальшому перераховано лише піддерево, що відповідає цьому дропу.

Це середовище виконання було реалізоване на мові програмування Java і використовує інтерфейс для комунікації між процесорами OpenMPI. Вихідний код програми доступний за URL-адресою: <https://bitbucket.org/mathpar/dap01/src/master/src/main/java/com/mathpar/parallel/dap/>.

### Список літератури

1. Dongarra J. With Extreme Scale Computing the Rules Have Changed. *Mathematical Software / J. Dongarra // ICMS 2016, 5th International Congress, Procdistributed memoryeedings (G.-M. Greuel, T. Koch, P. Paule, A. Sommese, eds.), Springer. — 2016. — No. 9725. — Pp. 3–6.*
2. Dynamic Task Discovery in PaRSEC- A data-flow task-based Runtime / R. Hoque, T. Hérault, G. Bosilca, J. Dongarra // *Proc. ScalA17, Proceedings of the 8th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems, November 12–17, 2017, Denver, CO, USA. — 2017.*
3. PaRSEC: A programming paradigm exploiting heterogeneity for enhancing scalability [Electronic resource] / [G. Bosilca, A. Bouteiller, A. Danalis et al.] // *Computing in Science and Engineering 99 (2013). — 2013. — https://doi.org/10.1109/MCSE.2013.98.*
4. Supercomputer Environment for Recursive Matrix Algorithms [Electronic resource] / G. I. Malaschonok, A. A. Sidko // *Program Comput Soft. — 2022. — Vol. 48. — Pp. 90–101. — 2022. — https://doi.org/10.1134/S0361768822020086.*

### References

- Bosilca, G., Bouteiller, A., Danalis, A., Faverge, M., Hérault, T., & Dongarra, J. (2013). PaRSEC: Exploiting Heterogeneity to enhance scalability. *Computing in Science and Engineering*, 15 (6), 36–45. <https://doi.org/10.1109/mcse.2013.98>.
- Dongarra, Jack. (2016). With Extreme Scale Computing the Rules Have Changed. In *Proceedings of the 25th ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC '16)*. Association for Computing Machinery, New York, NY, USA, 123. <https://doi.org/10.1145/2907294.2926972>.
- Hoque, Reazul, Hérault Thomas, Bosilca George, & Dongarra, Jack. (2017). Dynamic task discovery in PaRSEC: a data-flow task-based runtime. In *Proceedings of the 8th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA '17)* (Article 6, pp. 1–8). Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/3148226.3148233>.
- Malaschonok, G., & Sidko, A. (2022). Supercomputer environment for recursive matrix algorithms. *Programming and Computer Software*, 48 (2), 90–101. <https://doi.org/10.1134/s0361768822020086>.

A. Sidko

## WHAT IS NEW IN THE LATEST RELEASE OF MATHPAR-DAP RUNTIME

*In this paper, we recall the main features of the DAP runtime, that was published in [4]. But the main purpose of this paper is to describe the new functionality that appeared in our latest release. As an example of a block recursive algorithm, the Cholesky factorization of a symmetric positive definite matrix in the form of a block dichotomous algorithm is described. The results of experiments demonstrate good scalability of the proposed solution. Modern supercomputer systems containing hundreds of thousands of cores face difficulties in the organization of parallel computations (e.g., see [1]). The three main difficulties are the nonuniform hardware workload, accumulation of errors in the process of computations with large matrices, and possible failures of cores during the computation process.*

Recently, a universal Dynamic Task Discovery (DTD) scheme for the PaRSEC runtime environment [2], [3] has been developed. This environment can support systems with shared and distributed memory. This new paradigm demonstrated better performance compared with the parameterized task scheduling that was used earlier.

In [1] we described a new runtime environment for supercomputers with distributed memory. It is designed for solving matrix problems using block recursive algorithms.

Its main advantage is to provide an efficient computational process and good scalability of programs both for sparse and dense matrices on a cluster with distributed memory. Another advantage is the ability to reorganize the computational process in the event of failure of individual nodes during computations.

A key feature of DAP is its ability to sequentially unroll functions in depth, maintaining all states at any nesting level until all computations in the current computational subtree are complete. This design allows any processor to switch freely between subtasks without waiting for the completion of the current subtask.

An important feature of this runtime environment is protection against failures of some nodes during computations. The parent node that sent a drop to its child node must receive a result. However, instead of a result, it may receive a message regarding the status of the child node. In such cases, the drop task is redirected to an alternate node. No additional changes to the other nodes are required. As a result, only the subtree corresponding to this drop will be lost and subsequently recalculated.

We would like to develop cooperation in this scientific area. The software package developed by us is open for joint development, and can be freely used for scientific and educational purposes.

**Keywords:** distributed computing, parallel programming, OpenMPI, runtime.

Матеріал надійшов 26.11.2023



Creative Commons Attribution 4.0 International License (CC BY 4.0)



## ВІДОМОСТІ ПРО АВТОРІВ

- Бучко Олена Андріївна* — Ph.D. технічних наук, доцент факультету інформатики Національного університету «Києво-Могилянська академія», [olena.buchko@ukma.edu.ua](mailto:olena.buchko@ukma.edu.ua)
- Вакуленко Дмитро Вікторович* — доктор біологічних наук, кандидат технічних наук, професор, завідувач кафедри медичної інформатики Тернопільського національного медичного університету імені І. Я. Горбачевського, [dmitro\\_v@ukr.net](mailto:dmitro_v@ukr.net)
- Глибовець Андрій Миколайович* — доктор технічних наук, професор, декан факультету інформатики Національного університету «Києво-Могилянська академія», [a.glybovets@ukma.edu.ua](mailto:a.glybovets@ukma.edu.ua)
- Глибовець Микола Миколайович* — доктор фізико-математичних наук, професор кафедри інформатики факультету інформатики Національного університету «Києво-Могилянська академія», [glib@ukma.edu.ua](mailto:glib@ukma.edu.ua)
- Гуминський Віктор Валентинович* — кандидат фізико-математичних наук, старший викладач кафедри мережних технологій Національного університету «Києво-Могилянська академія», [v.guminskiy@ukma.edu.ua](mailto:v.guminskiy@ukma.edu.ua)
- Давиденко Андрій Миколайович* — студент Ph.D. програми «Інженерія програмного забезпечення» факультету інформатики Національного університету «Києво-Могилянська академія», [andrii.davydenko@ukma.edu.ua](mailto:andrii.davydenko@ukma.edu.ua)
- Жежерун Олександр Петрович* — кандидат фізико-математичних наук, доцент кафедри мультимедійних систем факультету інформатики Національного університету «Києво-Могилянська академія», [zhezherun@ukma.edu.ua](mailto:zhezherun@ukma.edu.ua)
- Кирієнко Оксана Валентинівна* — старший викладач факультету інформатики Національного університету «Києво-Могилянська академія», [o.kyriienko@ukma.edu.ua](mailto:o.kyriienko@ukma.edu.ua)
- Ліп'яніна-Гочаренко Христина Володимирівна* — кандидат технічних наук, доцент, доцент кафедри інформаційно-обчислювальних систем і управління Західноукраїнського національного університету, [kh.lipianina@wunu.edu.ua](mailto:kh.lipianina@wunu.edu.ua)
- Ляшко Володимир Іванович* — кандидат фізико-математичних наук, доцент кафедри інформатики Національного університету «Києво-Могилянська академія», [v.lyashko@ukr.net](mailto:v.lyashko@ukr.net)
- Нагнибіда Андрій Андрійович* — студент PhD програми «Комп'ютерні науки» факультету інформатики Національного університету «Києво-Могилянська академія», [a.nahnybida@ukma.edu.ua](mailto:a.nahnybida@ukma.edu.ua)
- Нгуєн Сан Бинь Ванович* — студент Ph.D. програми «Інженерія програмного забезпечення» факультету інформатики Національного університету «Києво-Могилянська академія», [nguen.sun@ukma.edu.ua](mailto:nguen.sun@ukma.edu.ua)
- Олецький Олексій Віталійович* — кандидат технічних наук, доцент кафедри мультимедійних систем факультету інформатики Національного університету «Києво-Могилянська академія», [oletsky@ukma.edu.ua](mailto:oletsky@ukma.edu.ua)
- Печкурова Олена Миколаївна* — старший викладач кафедри мультимедійних технологій факультету інформатики Національного університету «Києво-Могилянська Академія», [rueschkurova@ukma.edu.ua](mailto:rueschkurova@ukma.edu.ua)
- Пруднікова Анастасія Олександрівна* — студентка 4 року бакалаврської програми «Комп'ютерні науки» факультету інформатики Національного університету «Києво-Могилянська академія», [a.prudnikova@ukma.edu.ua](mailto:a.prudnikova@ukma.edu.ua)
- Пухальський Богдан Мирославович* — магістр програми «Інженерія програмного забезпечення» факультету інформатики Національного університету «Києво-Могилянська академія», [bohdan.pukhalskyi@ukma.edu.ua](mailto:bohdan.pukhalskyi@ukma.edu.ua)

- Сідько Алла Анатоліївна* — старший викладач кафедри мережних технологій факультету інформатики Національного університету «Києво-Могилянська академія», Ph.D. студентка Інституту програмних систем НАН України, a.sidko@ukma.edu.ua
- Смиш Олег Русланович* — старший викладач кафедри мультимедійних систем факультету інформатики Національного університету «Києво-Могилянська академія», магістр інженерії програмного забезпечення, аспірант програми «Комп'ютерні науки» у Національному університеті «Києво-Могилянська академія» та Інституті програмних систем Національної академії наук України, o.smysh@ukma.edu.ua
- Стецюк Петро Іванович* — доктор фізико-математичних наук, завідувач відділу методів негладкої оптимізації Інституту кібернетики імені В. М. Глушкова НАН України, stetsyukp@gmail.com
- Федорченко Віталій Михайлович* — магістр програми «Комп'ютерні науки» факультету інформатики Національного університету «Києво-Могилянська академія», fedorchenko@gmail.com
- Франчук Іван Олегович* — аспірант факультету інформатики Національного університету «Києво-Могилянська академія», i.franchuk@ukma.edu.ua
- Хмель Світлана Миколаївна* — магістр програми «Інженерія програмного забезпечення» факультету інформатики Національного університету «Києво-Могилянська академія», svitlana.khmel@ukma.edu.ua
- Царинюк Олександр Васильович* — студент Ph.D. програми «Комп'ютерні науки» факультету інформатики Національного університету «Києво-Могилянська академія», o.tsaryniuk@ukma.edu.ua

## NOTES ABOUT AUTHORS

- Olena Buchko* — Ph.D. Computer Sciences, Senior Lecturer, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [olena.buchko@ukma.edu.ua](mailto:olena.buchko@ukma.edu.ua)
- Andrii Davydenko* — Ph.D. student of Software Engineering, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [andrii.davydenko@ukma.edu.ua](mailto:andrii.davydenko@ukma.edu.ua)
- Vitalii Fedorchenko* — Master of Computer Science, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [fedorchenko@gmail.com](mailto:fedorchenko@gmail.com)
- Ivan Franchuk* — Ph.D. student, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [i.franchuk@ukma.edu.ua](mailto:i.franchuk@ukma.edu.ua)
- Mykola Glybovets* — Doctor of Physical and Mathematical Sciences, Professor, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [glib@ukma.edu.ua](mailto:glib@ukma.edu.ua)
- Andrii Hlybovets* — Doctor of Technical Sciences, Dean, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [a.glybovets@ukma.edu.ua](mailto:a.glybovets@ukma.edu.ua)
- Viktor Humynskiy* — Candidate of Physical and Mathematical Sciences, Senior lecturer, Department of Network Technologies, National University of Kyiv-Mohyla Academy, [v.guminskiy@ukma.edu.ua](mailto:v.guminskiy@ukma.edu.ua)
- Svitlana Khmel* — Master of Software Engineering, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [svitlana.khmel@ukma.edu.ua](mailto:svitlana.khmel@ukma.edu.ua)
- Oksana Kyriienko* — Senior Lecturer, Department of Informatics, National University of Kyiv-Mohyla Academy, [o.kyriienko@ukma.edu.ua](mailto:o.kyriienko@ukma.edu.ua)
- Khrystyna Lipianina-Honcharenko* — Ph.D. in Information Technologies, Associate professor, Associate professor, Department for Information Computer Systems and Control, West Ukrainian National University, [kh.lipianina@wunu.edu.ua](mailto:kh.lipianina@wunu.edu.ua)
- Volodymyr Lyashko* — Candidate of Physical and Mathematical Sciences, Associate Professor, Department of Informatics, National University of Kyiv-Mohyla Academy, [v.lyashko@ukr.net](mailto:v.lyashko@ukr.net)
- Andrii Nahnybida* — PhD student in the “Computer Science” program of the Faculty of Informatics at the National University “Kyiv-Mohyla Academy”, [a.nahnybida@ukma.edu.ua](mailto:a.nahnybida@ukma.edu.ua)
- San Byn Nhuien* — Ph.D. student, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [nguen.sun@ukma.edu.ua](mailto:nguen.sun@ukma.edu.ua)
- Oleksii Oletsky* — Ph.D. (Candidate of Technical Sciences), Associate Professor, Department of Multimedia Systems, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [oletsky@ukma.edu.ua](mailto:oletsky@ukma.edu.ua)
- Anastasiia Prudnikova* — Computer Science senior student, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [a.prudnikova@ukma.edu.ua](mailto:a.prudnikova@ukma.edu.ua)
- Olena Pyechkurova* — Senior Lecturer, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [pyechkurova@ukma.edu.ua](mailto:pyechkurova@ukma.edu.ua)
- Bohdan Pukhalskyi* — Master of Software Engineering, Faculty of Informatics, National University of Kyiv-Mohyla Academy, [bohdan.pukhalskyi@ukma.edu.ua](mailto:bohdan.pukhalskyi@ukma.edu.ua)
- Alla Sidko* — Senior Lecturer, Department of Network Technologies, Faculty of Informatics, National University of Kyiv-Mohyla Academy, Ph.D. student at the Institute of Software Systems, National Academy of Sciences of Ukraine, [a.sidko@ukma.edu.ua](mailto:a.sidko@ukma.edu.ua)
- Oleh Smysh* — Senior Lecturer, Department of Multimedia Systems, Faculty of Informatics, National University of Kyiv-Mohyla Academy, Master of Software Engineering, Computer Science Ph.D. student at the National University of Kyiv-Mohyla Academy and the Institute of Software Systems of National Academy of Sciences of Ukraine, [o.smysh@ukma.edu.ua](mailto:o.smysh@ukma.edu.ua)

*Petro Stetsyuk* — Doctor of Sciences (Physics and Mathematics), Head of the Department of Nonsmooth Optimization Methods, V. M. Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine, stetsyukp@gmail.com

*Oleksandr Tsaryniuk* — Student of Ph.D. Program Computer Science, Faculty of Informatics, National University of Kyiv-Mohyla Academy, o.tsaryniuk@ukma.edu.ua

*Dmytro Vakulenko* — Doctor of Biological Sciences, Candidate of Technical Sciences, Professor, Head of the Department of Medical Informatics, I. Horbachevsky Ternopil national Medical University, dmitro\_v@ukr.net

*Oleksandr Zhezherun* — Candidate of Physical and Mathematical Sciences, Associate Professor, Department of Multimedia Systems, Faculty of Informatics, National University of Kyiv-Mohyla Academy, zhezherun@ukma.edu.ua

## ЗМІСТ

<i>Олецький О. В., Франчук І. О., Гуминський В. В.</i> Про підхід до формування дворівневої моделі «стан–імовірність дії» на основі попарних порівнянь та методу аналізу ієрархій .....	4
<i>Стецюк П. І., Вакуленко Д. В., Ляшко В. І.</i> Оптимізаційні задачі для оцінок відносної важливості елементів у методі аналізу ієрархій .....	11
<i>Жежерун О. П., Смиш О. Р., Пруднікова А. О.</i> Підходи до побудови виводу в онтологічній базі знань .....	17
<i>Бучко О. А., Нгуєн С. Б. В.</i> Порівняльний аналіз алгоритмів суперроздільної здатності для стиснення зображень.....	24
<i>Ліп'яніна-Гончаренко Х. В.</i> Метод формування навчальної вибірки для масивів даних на основі машинного навчання.....	30
<i>Царинюк О. В.</i> Застосування методів машинного навчання для класифікації супутникових зображень: огляд літератури та ключових фреймворків .....	36
<i>Давиденко А. М., Глибовець М. М.</i> Веббазована система групового розподіленого розроблення програм .....	41
<i>Глибовець А. М., Хмель С. М., Печкурова О. М.</i> Використання нейрокомп'ютерних інтерфейсів для збирання та накопичення інформації про користувачів вебресурсів .....	48
<i>Федорченко В. М.</i> Декларативне конфігурування інверсії контролю в сучасних .NET8 застосунках.....	57
<i>Пухальський Б. М., Кирієнко О. В.</i> Системи оброблення неструктурованої мультимедійної інформації .....	62
<i>Нагнибіда А. А.</i> Стандарти подання навчальних матеріалів у програмних системах підтримки електронної освіти .....	69
<i>Сідько А. А.</i> Нова версія середовища виконання Mathpar-DAP.....	76
Відомості про авторів .....	81

## CONTENTS

<i>O. Oletsy, I. Franchuk, V. Humynskiy.</i> On an Approach to Forming Two-Level Model “State-Probability of Action” on the Base of Pairwise Comparisons on the and Analytic Hierarchy Process .....	4
<i>P. Stetsyuk, D. Vakulenko, V. Lyashko.</i> Optimization Problems for Assessing the Relative Importance of Elements in the Analytic Hierarchy Process .....	11
<i>O. Zhezherun, O. Smysh, A. Prudnikova.</i> Approaches to Inference Search in the Ontological Knowledge Base .....	17
<i>O. Buchko, S. B. Nhuien.</i> Comparative Analysis of Super-Resolution Algorithms for Image Compression.....	24
<i>Kh. Lipianina-Honcharenko.</i> Method for Forming Training Samples for Data Arrays Based on Machine Learning .....	30
<i>O. Tsaryniuk.</i> Application of Machine Learning Methods for Satellite Image Classification: a Literature Review and Overview of Key Frameworks .....	36
<i>A. Davydenko, M. Glybovets.</i> Web-Based System for Distributed Groupware Software Development.....	41
<i>A. Hlybovets, S. Khmel, O. Pyechkurova.</i> Usage of Neuro-Computer Interfaces for Collecting and Storing Information About Web Resource Users.....	48
<i>V. Fedorchenko.</i> Declarative Inversion of Control Configuration in Modern .NET8 Applications.....	57
<i>B. Pukhalskyi, O. Kyriienko.</i> Systems for Processing Unstructured Multimedia Information.....	62
<i>A. Nahnybida.</i> Standards for Presenting Educational Materials in Electronic Education Support Systems.....	69
<i>A. Sidko.</i> What is New in the Latest Release of Mathpar-DAP Runtime .....	76
Notes about authors.....	81

**НАУКОВІ ЗАПИСКИ НаУКМА  
КОМП'ЮТЕРНІ НАУКИ**

**NaUKMA RESEARCH PAPERS  
COMPUTER SCIENCE**

**Том 6**

Редакторка і коректорка *О. Пазюк*  
Комп'ютерна верстка *М. Кулікова*

**Повну електронну версію наукового журналу  
«Наукові записки НаУКМА. Комп'ютерні науки»  
представлено на вебсайті  
[prcomp.ukma.edu.ua](http://prcomp.ukma.edu.ua)**

Підписано до друку 24.03.2024.  
Формат 60 × 84 <sup>1</sup>/<sub>8</sub>. Гарнітура Times New Roman.  
Папір офсетний № 1. Ум. друк. арк. 10.11.  
Наклад 50 прим.

Адреса редакції:  
вул. Г. Сковороди, 2, м. Київ, 04070,  
тел.: (044) 463-66-68

Свідоцтво про внесення до Державного реєстру видавців, виготівників і розповсюджувачів  
книжкової продукції серія ДК № 3631 від 23.11.2009