

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

**АВТОМАТИЗОВАНІ СИСТЕМИ
УПРАВЛІННЯ І ПРИЛАДИ
АВТОМАТИКИ**

Всеукраїнський міжвідомчий
науково-технічний збірник

Заснований у 1965 р.

Випуск 179

Харків
2023

У збірнику наведено результати досліджень, що стосуються моделей і методів модернізації та рефакторингу інформаційних систем різного призначення, моделей експертних та рекомендаційних систем. Запропоновано нові підходи та рішення в області автоматизованого управління обробкою великих даних, машинного навчання та інженерії вимог до інформаційних систем.

Для викладачів університетів, науковців, фахівців, аспірантів.

The collection contains research results related to models and methods of modernization and refactoring of information systems for various purposes, models of expert and recommendation systems. New approaches and solutions in the field of automated management of big data processing, machine learning and requirements engineering for information systems are proposed.

For university professors, researchers, specialists, graduate students.

Редакційна колегія:

В.В. Семенець, д-р техн. наук, проф. (гол. ред.), *В.М. Левикін*, д-р техн. наук, проф. (відпов. ред.), *М.В. Євланов*, д-р техн. наук, проф. (відпов. секр.), *Є.В. Бодяньський*, д-р техн. наук, проф., *І.В. Гребеннік*, д-р техн. наук, проф., *А.Л. Єрохін*, д-р техн. наук, проф., *А.О. Каргін*, д-р техн. наук, проф., *Б.О. Мороз*, д-р техн. наук, проф., *І.Ш. Невлюдов*, д-р техн. наук, проф., *К.Е. Петров*, д-р техн. наук, проф., *І.В. Рубан*, д-р техн. наук, проф., *С.Г. Удовенко*, д-р техн. наук, проф., *О.Є. Федорович*, д-р техн. наук, проф., *В.О. Філатов*, д-р техн. наук, проф., *Г.З. Халімов*, д-р техн. наук, проф.

Свідоцтво про держреєстрацію
печатного засобу масової інформації

КВ № 4619 від 18.10.2000 р.

Адреса редакційної колегії: Україна, 61166, Харків, просп. Науки, 14, Харківський національний університет радіоелектроніки, кімн. 254, тел. (057) 70-21-451

© Харківський національний університет
радіоелектроніки, 2023

ЗМІСТ

| | |
|---|----|
| ЄВЛАНОВ М.В., ЮР'ЄВ І.О., МІРОШНИЧЕНКО Д.О. ОЦІНЮВАННЯ ВИТРАТ НА ПРОВЕДЕННЯ РЕФАКТОРИНГУ БАЗИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ, ЯКА ЕКСПЛУАТУЄТЬСЯ..... | 4 |
| ЛУТВУНЕНКО М., РЕБЕЗУК Л. EXPERT INFORMATION PROCESSING SYSTEM FOR TRAFFIC LIGHT SYSTEM DECISION-MAKING WITH ADAPTIVE CONDITIONAL TRAM PRIORITY..... | 17 |
| МІХНОВА А.В., МІХНОВ Д.К., ЧИРКОВА К.С. МОДЕРНІЗАЦІЯ СПЕЦІАЛІЗОВАНИХ ІНФОРМАЦІЙНИХ СИСТЕМ СЛУЖБИ КРОВІ..... | 25 |
| СИТНИКОВА П.Е., ГРЕБЕНЮК М.О. РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ КОМПАКТНОЇ ГІБРИДНОЇ МОДЕЛІ КОРИСТУВАЧА..... | 32 |
| ПЕРОВА І.Г., МІРОШНИЧЕНКО Н.С. ОГЛЯД ІСНУЮЧИХ МЕТОДІВ ЗМЕНШЕННЯ РОЗМІРНОСТІ ТА КЛАСИФІКАЦІЇ ВЕЛИКИХ ВИБІРОК ДАНИХ..... | 42 |
| ПАНФЬОРОВА І.Ю., БУЦЬКА А.С. ДОСЛІДЖЕННЯ АНСАМБЛЮВАННЯ МОДЕЛЕЙ MACHINE LEARNING В МЕДИЧНІЙ ДІАГНОСТИЦІ..... | 50 |
| КОЖАНОВ А.Є. ДОСЛІДЖЕННЯ МОДЕЛЕЙ ФОРМАЛЬНОГО ОПИСУ ЗНАНЬ ПРО ФУНКЦІОНАЛЬНІ ВИМОГИ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ EER-ДІАГРАМИ..... | 58 |
| РЕФЕРАТИ..... | 73 |

М.В. ЄВЛАНОВ, І.О. ЮР'ЄВ, Д.О. МІРОШНИЧЕНКО

ОЦІНЮВАННЯ ВИТРАТ НА ПРОВЕДЕННЯ РЕФАКТОРИНГУ БАЗИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ, ЯКА ЕКСПЛУАТУЄТЬСЯ

Розглянуто основні особливості прийняття рішень щодо доцільності здійснення рефакторингу бази даних. Встановлено, що подібні рішення рекомендується приймати виключно на основі інтуїції та досвіду особи, яка приймає рішення. Розроблені моделі оцінювання кількості рядків основних SQL-команд, які використовуються в процесі рефакторингу бази даних. Вдосконалено модель СОСОМО II, яка дозволяє оцінити трудовитрати та витрати часу на проведення рефакторингу бази даних. Проведено експериментальну перевірку отриманих результатів.

1. Вступ

Актуальне визначення, запропоноване М. Фаулером, трактує термін «рефакторинг» як зміну внутрішньої структури програмного забезпечення з метою зробити його легшим для розуміння та дешевшим для модифікації без зміни його спостережуваної поведінки [1]. Суть рефакторингу за М. Фаулером полягає у серії невеликих трансформацій, що зберігають поведінку програмної системи. Кожна така трансформація вносить незначні зміни до структури системи, але послідовність цих трансформацій може призвести до значної реструктуризації. Після кожного рефакторингу працездатність системи повністю підтримується, що зменшує ймовірність того, що система може отримати серйозний злам під час реструктуризації, що зменшує ймовірність того, що система може отримати серйозний злам під час реструктуризації [1].

Серед розмаїття методів рефакторингу особливо виділяють рефакторинг бази даних (БД). Даний різновид рефакторингу у [2] визначається як розвиток існуючої схеми БД шляхом одноразового внесення невеликих змін з метою підвищення якості проекту БД без зміни її семантики. Рефакторингу можуть бути піддані або структурні аспекти схеми БД (визначення таблиць, представлень тощо), або функціональні аспекти (процедури, що зберігаються, тригери тощо). Під час проведення рефакторингу схеми БД доводиться не тільки перевизначати саму схему, а й вносити зміни у зовнішні системи, такі як бізнес-додатки чи засоби вибірки даних, які прив'язані до конкретної схеми [2].

Сучасні дослідження в галузі рефакторингу спрямовані, переважно, на вирішення прикладних питань з організації та особливостей виконання окремих дій під час проведення рефакторингу. Для рефакторингу БД вирішення цих питань привело до створення Скоттом Амблером та його колегами прикладної методики проведення рефакторингу БД [2, 3]. Вона включає в себе набір процедур та практик, які допомагають безпечно та ефективно проводити рефакторинг БД у рамках проекту розробки програмного забезпечення систем різного призначення.

Використання цієї методики передбачає проведення усіх операцій з рефакторингу БД тільки в тому випадку, якщо на початку процесу приймається рішення щодо доцільності проведення такого рефакторингу. Робота з формування та прийняття цього рішення у [2] розглядається як послідовний пошук відповідей на такі питання:

- а) «Чи має сенс операція рефакторингу, яка пропонується?»;

- б) «Чи повинні зміни бути виконані негайно?»;
- в) «Чи виправдовує результат витрачені зусилля?».

Для вирішення цих питань у [2] пропонується використовувати інтуїцію та досвід, набуті особою, що приймає ці рішення, під час професійної діяльності в галузі проектування і розробки БД.

Ця рекомендація залишається актуальною й досі. Так, проведення рефакторингу БД в умовах використання Agile-методологій для управління відповідним ІТ-проектом рекомендується починати з визначення необхідності і доцільності проведення рефакторингу БД. Після цього рекомендується оцінити ймовірність того, що зміни дійсно потрібні. Ці визначення та оцінювання повинні виконуватися з використанням «інтуїції», яка виникає на основі попереднього досвіду роботи інженера з обробки даних Agile з розробником програми [3]. Але в таких випадках рішення про доцільність рефакторингу БД є значною мірою суб'єктивним і сильно залежить від тієї особи, яка приймає це рішення у кожному конкретному ІТ-проекті. Моделі і методи, які дозволяють приймати об'єктивне рішення про доцільність проведення рефакторингу БД, залишаються майже недослідженими.

2. Аналіз літературних даних і постановка проблеми дослідження

Аналіз когнітивної структури досліджень з рефакторингу та розвитку цих досліджень з часом був проведений у [4]. Результати цього аналізу підтверджують, що дослідження з рефакторингу залишалися в переліку головних тем наукових публікацій в період з 2001 р. до 2014 р. Але, починаючи з 2005 р., головним фактором об'єднання публікацій досліджень в галузі рефакторингу у тематичні групи стали Agile-методології управління ІТ-проектами. Ця тенденція спостерігалася до 2009 р., після чого головними групами досліджень з рефакторингу стають «Experimentation» і (меншою мірою) «Software-Evolution» і «Specific-Refactorings» [4].

Тенденціями розвитку досліджень з рефакторингу, починаючи з 2015 р., у [4] вважалися:

а) поява значної кількості вузькоспеціалізованих тем досліджень, що, ймовірно, пов'язано зі збільшенням використання інструментів рефакторингу, недоступних у минулому;

б) присутність тематики «Agile methodologies» серед рушійних (тобто, перспективних) тем у дослідницькій галузі рефакторингу, хоча й не в такому обсязі, як у 2005-2009 рр.

Цей прогноз значною мірою виправдався. Протягом трьох наступних років з'являються дослідження, присвячені вузькоспеціалізованим питанням рефакторингу БД. Так, у [5] досліджувалися наслідки прикладних робіт з рефакторингу БД системи оптимізації логістичних процесів. Результати досліджень були сформульовані у вигляді п'яти ключових уроків, які були присвячені виключно практичним аспектам і не давали нічого нового теорії комп'ютерних наук і, зокрема, рефакторингу БД. У [6] досліджувалися питання допомоги адміністраторам БД у діагностиці антишаблонів SQL і пропонувалися методи рефакторингу для вирішення антишаблонів. Було встановлено, що виявлення антишаблонів SQL все ще значною мірою залежить від семантики даних, а інструмент виявлення слід використовувати напівавтоматично, тобто він може вказувати на потенційно проблемні місця в схемі БД, які потребують подальшої діагностики адміністратором цієї БД. Ці результати дослідження мали також виключно прикладний характер.

Слід також зазначити появу дослідницьких робіт, присвячених розробці інструментальних засобів рефакторингу БД. Так, у [7] було запропоновано метод автоматичного синтезу нової версії БД з урахуванням її оригінальної версії, вихідної та цільової схем. Цей метод не вимагає ручного керування користувачем і гарантує, що

синтезована БД еквівалентна вихідній. Але запропонований у [7] метод не дозволяв вирішувати розглянуті вище питання доцільності проведення рефакторингу БД.

Окремо серед цієї множини дослідницьких робіт слід виділити роботу [8], в якій розглядається питання розробки методу реінжинірингу реляційних БД з урахуванням наявності неявних взаємопов'язаних функціонально залежних даних, які впливають на структуру логічної моделі. Одним з результатів, отриманих у [8], є дослідження спільного для задач адаптації та рефакторингу етапу формування цільової логічної схеми БД. Однак цей результат є цікавим з точки зору теорії організації виконання робіт з рефакторингу БД вже після того, як прийнято позитивне рішення щодо доцільності такого рефакторингу.

Сучасні дослідження також підтверджують цю тенденцію. Так, у [9] отримали значний розвиток питання пошуку та ліквідації антишаблонів SQL. У [10] розглядаються питання вдосконалення підходу до автоматизації рефакторингу одного з можливих антишаблонів, а саме антишаблону Unlimited Records. Одним з результатів цього вдосконалення є алгоритми виявлення та рефакторингу для двох різних підвидів антишаблону Unlimited Records. Цікавішим з наукової точки зору є дослідження [11], в якому пропонується автоматизований підхід для визначення еквівалентності SQL-запитів та пов'язаних з ними програм до та після проведення рефакторингу. У цьому підході використовується спеціальне логічне моделювання SQL-запитів першого порядку, яке відповідає імперативній семантиці. У [12] результатом дослідження є розроблені методи такого специфічного різновиду рефакторингу, як рефакторинг БД для нормалізації. Але у всіх цих дослідженнях питання визначення доцільності проведення рефакторингу БД практично не розглядаються і не вирішуються.

Необхідність подолання розриву між рефакторингом як дослідженням і його впровадженням на практиці шляхом виділення загальних намірів рефакторингу, які більше підходять для того, з чим стикаються розробники в реальності, визначена у [13]. Результатом цього дослідження є, зокрема, визначення п'яти областей, у яких розробникам зазвичай потрібна допомога з рефакторингом: оптимізація коду, інструменти та IDE, архітектура та шаблони дизайну, модульне тестування та БД. Крім того, у [13] проведено детальне дослідження можливості і доцільності використання такого різновиду рефакторингу, як рефакторинг Stack Overflow, за допомогою серії кількісних і якісних експериментів. Але результати, отримані у [13], дають лише загальні відповіді щодо можливості та доцільності виконання рефакторингу БД. Проблема пошуку відповіді на питання «Чи виправдовує результат витрачені зусилля?» залишається нерозв'язаною.

Аналіз розглянутих публікацій дозволяє сформулювати висновок про те, що вирішення проблеми формування та прийняття рішення щодо доцільності проведення рефакторингу БД за останні роки набуває актуальності не тільки з теоретичної, а й з прикладної точки зору. Однак навіть загальні аспекти вирішення питань з визначення доцільності рефакторингу БД залишаються майже недослідженими. Між тим, вирішення цих питань хоча б на загальному рівні є актуальним не тільки для формування прикладних технік формування і прийняття подібних рішень, а й для управління ІТ-проектами. Зокрема, проблема пошуку відповіді на питання «Чи виправдовує результат витрачені зусилля?» під час прийняття рішення щодо доцільності проведення рефакторингу БД стає особливо важливою, якщо проводити оцінювання витрат на рефакторинг БД як на різновид ІТ-проекту. Тому проблему даного дослідження слід сформулювати як проблему проведення кількісного оцінювання витрат на рефакторинг БД як на різновид ІТ-проекту.

3. Мета і задачі дослідження

Метою даного дослідження є вдосконалення параметричної моделі оцінювання трудовитрат на ІТ-проекти СОСОМО ІІ для її використання під час оцінки витрат на рефакторинг БД. Застосування моделі СОСОМО ІІ хоча й відзначається підвищеною складністю, дозволяє отримати досить точні оцінки трудовитрат на виконання різних ІТ-проектів. Тому досягнення цієї мети дозволить обґрунтовано прийняти рішення щодо доцільності проведення рефакторингу БД в умовах діючих обмежень ресурсів та часу виконання ІТ-проекту такого рефакторингу.

Для досягнення цієї мети у статті вирішуються такі задачі:

- розробка моделей оцінювання кількості рядків вихідного коду SQL-команд, які використовуються під час проведення рефакторингу БД;
- вдосконалення елементів моделі СОСОМО ІІ для її застосування під час оцінювання трудовитрат на проведення рефакторингу БД;
- експериментальна перевірка результатів вдосконалення моделі СОСОМО ІІ.

4. Моделі, що використовуються для оцінки витрат на ІТ-проекти

У ході дослідження пропонується використовувати моделі СОСОМО ІІ. Вибір даних моделей обумовлений можливістю їх використання не тільки для оцінювання трудовитрат на ІТ-проекти створення програмних систем різного призначення, а й для оцінювання трудовитрат на модифікацію чи розвиток програмних систем, які експлуатуються.

Основою моделі СОСОМО ІІ є модель розрахунку трудовитрат у людино-місяцях (person months, PM), яка має загальний вигляд [14]:

$$PM = A \times [Size']^E \times \prod_{i=1}^7 EM_i + PM_M, \quad (1)$$

де A – числовий коефіцієнт, який для моделей СОСОМО та СОСОМО ІІ дорівнює 2,94; $Size'$ – кількість рядків вихідного програмного коду, яка розраховується за такою формулою [14]:

$$Size' = Size \times \left(1 + \frac{BRAK}{100} \right); \quad (2)$$

$BRAK$ – показник, який відображує змінність вимог у ІТ-проекті і являє собою відсоток коду, який видаляється з ІТ-проекту в результаті зміни вимог; E – показник ступеня, в який слід возвести значення $Size'$, розрахунок цього показника здійснюється за формулою [14]:

$$E = 0,91 + 0,01 \times \sum_{j=1}^5 SF_j, \quad (3)$$

SF_j – драйвери масштабів ІТ-проекту ($PREC$ (показник безпрецедентності продукту); $FLEX$ (показник гнучкості розробки продукту); $RESL$ (показник надання переваги архітектурним або ризикованим рішенням); $TEAM$ (показник згуртованості команди); $PMAT$ (показник зрілості процесів розробки продукту)); EM_i – драйвери витрат ІТ-проекту ($RCPX$ (реалізованість і складність продукту); $RUSE$ (повторне використання, яке вимагається); $PDIF$ (складність

платформи); *PERS* (характеристики персоналу); *PREX* (досвід персоналу); *FCIL* (використовувані засоби); *SCED* (розклад); *PM_M* – частина моделі COCOMO II, яка не використовується для формули зміни коду [14].

Для оцінки трудовитрат на модифікацію чи розвиток програмних систем, які експлуатуються, модель (1) була скоригована. Суть цієї корекції полягала у використанні замість елемента *Size* елемента (*Size*)_M, який розраховується за формулою [14]

$$(Size)_M = (SizeAdded + SizeModified) \times MAF, \quad (4)$$

де *SizeAdded* – кількість рядків вихідного коду, які було додано під час модифікації програмної системи; *SizeModified* – кількість рядків вихідного коду, які було змінено під час модифікації програмної системи; *MAF* – коригуючий коефіцієнт модифікації програмної системи, який розраховується за формулою [14]

$$MAF = 1 + \left(\frac{SU}{100} \times UNFM \right); \quad (5)$$

SU – штраф, викликаний витратами на розуміння програмного коду; *UNFM* – показник рівня відокремленості команди.

Значення драйверів масштабів ІТ-проєкту і драйверів витрат ІТ-проєкту наведені у [14].

Отримана оцінка трудовитрат (1) далі використовується для розрахунку витрат часу за формулою [14]

$$TDEV = \left[3,67 \times (PM_{NS})^F \right] \times \frac{SCED\%}{100}, \quad (6)$$

де *PM_{NS}* – людино-місяці, оцінені без драйвера витрат SCED (номінальний графік); *SCED%* – необхідне стиснення розкладу; *F* – показник масштабування для рівняння розкладу, значення якого розраховується за формулою [14]

$$F = (0,28 + 0,2 \times [E - 0,91]), \quad (7)$$

де *E* – масштабний показник для рівняння зусиль, який розраховується за формулою (3). Моделі COCOMO II створювалися виключно для оцінювання витрат на створення чи модифікацію програмних систем. Використання моделей COCOMO II для оцінювання витрат на проведення рефакторингу БД стикається із значними труднощами. Ці труднощі пов'язані із визначенням правил розрахунку значень елементів *SizeAdded* та *SizeModified*. Тому виникає задача модифікації моделей COCOMO II з врахуванням особливостей рефакторингу БД як різновиду ІТ-проєкту.

5. Вирішення задачі оцінювання витрат на проведення рефакторингу бази даних інформаційної системи, яка експлуатується.

5.1. Оцінювання кількості рядків вихідного коду SQL-команд які використовуються під час проведення рефакторингу бази даних

Оцінювання рефакторингу БД з використанням такого показника, як кількість рядків вихідного коду, неодмінно приводить до визначення кількості рядків вихідного коду, який написано мовами створення, модифікації та управління даними в БД. Найрозповсюдженішою подібною мовою є SQL. Тому пропонується розраховувати кількісні значення елементів *SizeAdded* і *SizeModified* формули (4) як сукупну кількість рядків коду SQL-команд, що відповідають за додавання та модифікацію елементів БД, яка підлягає рефакторингу.

Найчастіше під час рефакторингу використовуються SQL-команди, які відносяться до групи команд Data Definition Language. Ці команди безпосередньо відповідають за роботу з таблицями БД і дозволяють виконувати такі операції [15]:

- створення (create): використовується при рефакторингу БД для створення нової таблиці при додаванні нової функції інформаційної системи;
- видалення (drop): використовується при рефакторингу БД для видалення існуючої таблиці, наприклад, коли треба відмовитися від існуючої сутності або функції інформаційної системи;
- редагування (alter): використовується при рефакторингу БД для зміни структури таблиці, наприклад, додавання чи видалення стовпців, змінювання типу даних, встановлювання обмежень та індексів (цю операцію також можна використовувати для перейменування таблиці або її стовпців);
- очищення (truncate): використовується при рефакторингу БД для очищення існуючої таблиці.

Розглянемо шаблони цих команд, враховуючи їх типові структури. Так, шаблон SQL-команди створення наведено на рис. 1 [16]. Шаблон SQL-команди видалення наведено на рис. 2 [16]. Шаблон SQL-команди редагування наведено на рис. 3 [16]. Шаблон SQL-команди очищення наведено на рис. 4 [16].

Використання цих шаблонів дає змогу оцінити кількість рядків вихідного коду для кожної з цих SQL-команд під час рефакторингу БД за умови, що ця БД може бути описана у термінах моделі «сутність – зв'язок» (Entity – Relation Model, ERM). Для цього пропонується використовувати такі моделі:

```
CREATE TABLE <name> (  
    <column 1>,  
    <column 2>,  
    .....  
    <column n>,  
    PRIMARY KEY (<column 1>, <column 2>, ..., <column n>),  
    FOREIGN KEY (<column 1>, <column 2>, ..., <column n>)  
        REFERENCES <table name> (<column 1>, <column 2>, ..., <column n>)  
);
```

Рис. 1. Шаблон SQL-команди створення

$$Size_{CR_r} = \left| \left\langle atr_{E_r}^j \right\rangle \right| + 4; j = 1, \dots, N, \quad (8)$$

```
DROP TABLE/VIEW/CONSTRAIN <name>;
```

Рис. 2. Шаблон SQL-команди видалення

```
ALTER <ObjectType> <ObjectName>  
  <Operation 1> <Parameters 1>,  
  <Operation 2> <Parameters 2>,  
  .....  
  <Operation n> <Parameters n>;
```

Рис. 3. Шаблон SQL-команди редагування

```
TRUNCATE TABLE <name>;
```

Рис. 4. Шаблон SQL-команди очищення

$$Size_{DR_r} = 1, \quad (9)$$

$$Size_{AL_r} = \left| \left\langle atr_{E_r}^j \right\rangle \right| + 1; j = 1, \dots, N, \quad (10)$$

$$Size_{TR_r} = 1, \quad (11)$$

де $Size_{CR_r}$ – кількість рядків SQL-команди створення; r – ідентифікатор сутності E_r із запиту на зміну інформаційної системи (та, відповідно, її БД); $\left\langle atr_{E_r}^j \right\rangle$ – множина атрибутів сутності E_r ; N – кількість атрибутів, з яких складаються усі сутності із запиту на зміну інформаційної системи; $Size_{DR_r}$ – кількість рядків SQL-команди видалення; $Size_{AL_r}$ – кількість рядків SQL-команди редагування; $Size_{TR_r}$ – кількість рядків SQL-команди очищення.

У моделі (8) додатковий доданок становить суму кількості строк, в яких здійснюються чотири визначення: типу команди, первинного ключа, вторинного ключа та посилання цих ключів на іншу таблицю. Тому додатковий доданок дорівнює 4. У моделі (9) немає операцій з атрибутами, тому залишається тільки додатковий доданок, який дорівнює сумі кількості строк в команді видалення, яка завжди дорівнює 1. У моделі (10) додатковий доданок становить 1 і описує кількість рядків, у яких здійснюється об'явлення типу команди. У моделі (11) немає операцій з атрибутами, тому залишається тільки додатковий доданок, який дорівнює сумі кількості строк в команді очищення, що завжди дорівнює 1.

Використання моделей (8)-(11) дає змогу оцінити кількість рядків вихідного коду SQL-команд, які необхідно виконати для здійснення рефакторингу БД як ІТ-проєкту, що реалізує запити на зміну інформаційної системи та її БД, які експлуатуються. Для цього необхідно знати описи сутностей та атрибутів у запитах на зміну, які визнано необхідним реалізувати. Але ця інформація стає відомою ще під час пошуку відповіді на запитання «Чи має сенс операція рефакторингу, яка пропонується?», як показано авторами у [17].

5.2. Вдосконалення елементів моделі СОСОМО II для її застосування під час оцінювання трудовитрат на проведення рефакторингу бази даних

Використання моделей (8)-(11) дозволяє вдосконалити модель (4) як елемент моделей СОСОМО II, що дозволяє визначити кількість рядків вихідного коду для випадку ІТ-проєктів

вдосконалення, розвитку чи рефакторингу існуючих систем. Так, для ІТ-проєкту рефакторингу БД інформаційної системи, яка експлуатується, значення елементу $SizeAdded$ формули (4) можна розрахувати за формулою

$$SizeAdded = \sum_{r=1}^{N_{CR}} Size_{CR_r}, \quad (12)$$

де N_{CR} – загальна кількість операцій додавання у запиті на зміни.

Значення елементу $SizeModified$ формули (4) в цьому випадку можна розрахувати за формулою

$$SizeModified = \sum_{r=1}^{N_{DR}} Size_{DR_r} + \sum_{r=1}^{N_{AL}} Size_{AL_r} + \sum_{r=1}^{N_{TR}} Size_{TR_r}, \quad (13)$$

де N_{DR} – загальна кількість операцій видалення у запиті на зміни; N_{AL} – загальна кількість операцій редагування у запиті на зміни; N_{TR} – загальна кількість операцій очищення у запиті на зміни.

Тоді модель (4) як елемент моделей СОСОМО II для ІТ-проєкту рефакторингу БД інформаційної системи, яка експлуатується, матиме вигляд

$$(Size)_M = \left(\sum_{r=1}^{N_{CR}} Size_{CR_r} + \sum_{r=1}^{N_{DR}} Size_{DR_r} + \sum_{r=1}^{N_{AL}} Size_{AL_r} + \sum_{r=1}^{N_{TR}} Size_{TR_r} \right) \times MAF. \quad (14)$$

Таке вдосконалення дає змогу використовувати параметричні моделі СОСОМО II для оцінювання трудовитрат на ІТ-проєкти рефакторингу БД.

5.3. Експериментальна перевірка результатів вдосконалення моделі СОСОМО II.

Для експериментальної перевірки було обрано інформаційно-облікову систему управління послугами однієї з організацій, яка надає послуги Інтернет-провайдера. ER-діаграма БД цієї системи наведена на рис. 5.

Користувачі цієї інформаційно-облікової системи незадоволені тим, що вони не мають можливості самостійного гнучкого налаштування послуг (їх налаштування апріорно визначені). Ці недоліки планується виправити шляхом рефакторингу системи і, зокрема, рефакторингу БД цієї системи. Для опису запитів на зміни системи, яка експлуатується, були визначені запити на зміни, UserStory яких наведені на рис. 6 та рис. 7.

Лідер сумісно з командою розробників, проаналізувавши описи UserStory запитів на зміни, наведені на рис. 7 та рис. 8, створюють ER-діаграми, які описують схеми даних для цих запитів на зміни. Ці ER-діаграми наведені на рис. 9 та рис. 10.

На рис. 9 та рис. 10 українською мовою позначено таблиці та поля, що додаються запитом на зміну, англійською мовою позначено таблиці та поля актуальної БД, яка показана на рис. 6.

З рис. 9 видно, що під час проведення рефакторингу актуальної БД інформаційно-облікової системи як реалізації запиту на зміну «Створення замовлення» (далі позначається ідентифікатором 1) слід додати одну таблицю, яка складається з 5 атрибутів, та модифікувати одну таблицю,

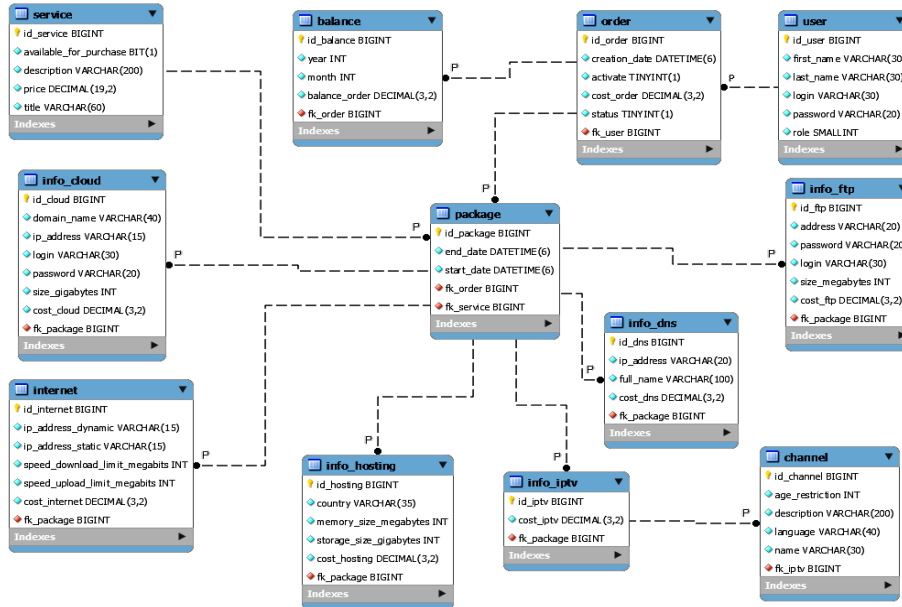


Рис. 5. ER-діаграма бази даних інформаційно-облікової системи організації, яка надає послуги Інтернет-провайдера



Рис. 6. UserStory запити на зміну «Створення замовлення»



Рис. 7. UserStory запити на зміну «Облік розподілу користувачів за населеним пунктом»

додавши один атрибут. Тому елементи *SizeAdded* і *SizeModified* будуть визначатися таким чином:

$$SizeAdded_1 = \sum_{r=1}^1 Size_{CR_r} = 5 + 4 = 9, \quad SizeModified_1 = \sum_{r=1}^1 Size_{AL_r} = 1 + 1 = 2. \quad (15)$$

Під час проведення рефакторингу актуальної БД інформаційно-облікової системи як реалізації запити на зміну «Облік розподілу користувачів за населеним пунктом» (далі

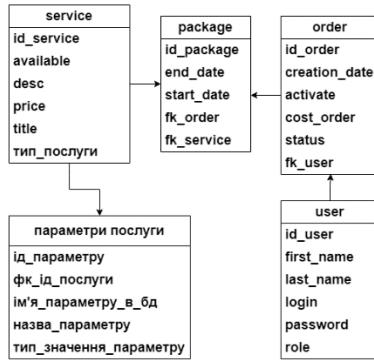


Рис. 8. ER-діаграма схеми даних для запиту на зміни «Створення замовлення»

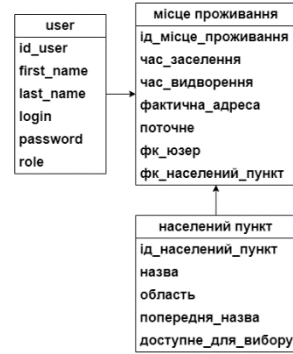


Рис. 9. ER-діаграма схеми даних для запиту на зміни «Облік розподілу користувачів за населеним пунктом»

позначається ідентифікатором 2) слід додати дві таблиці, перша з яких складається з 7 атрибутів, а друга – з 5 атрибутів. Тому для цього випадку визначається лише значення елемента *SizeAdded* таким чином:

$$SizeAdded_2 = \sum_{r=1}^2 Size_{CR_r} = (7 + 4) + (5 + 4) = 20 . \quad (16)$$

Інформаційно-облікова система має такі характеристики: високий рівень інтеграції, слабкий ступінь зв'язності елементів, добру кореляцію між програмою і додатком, добрий рівень присутності в коді коментарів та заголовків; корисну документацію, але є окремі слабкі місця. Тоді штраф, викликаний витратами на розуміння програмного коду, дорівнює 20. Команда проєкту повністю згуртована. Отже, розрахунок коригуючого коефіцієнту проведення рефакторингу БД *MAF* має наступний вигляд:

$$MAF = 1 + \left(\frac{SU}{100} \times UNFM \right) = 1 + \left(\frac{20}{100} \times 0 \right) = 1 + 0 = 1 . \quad (17)$$

Тоді загальна кількість рядків вихідного SQL-коду в запропонованому ІТ-проєкті рефакторингу БД за формулою (14) приймає таке значення:

$$(Size)_M = (9 + 2 + 20) \times 1 = 31 . \quad (18)$$

Оскільки запити на зміни не змінювалися під час їх надходження та аналізу, коефіцієнт *BRAK* буде дорівнювати 0. Тому кількість рядків вихідного програмного коду для моделі (1), яка розраховується за формулою (2), приймає таке значення:

$$Size' = Size \times \left(1 + \frac{BRAK}{100} \right) = 31 \times \left(1 + \frac{0}{100} \right) = 31 . \quad (19)$$

Виходячи з наявності досвіду виконання подібних проєктів, незначних відхилень у гнучкості, завдання повних (100 %) специфікацій елементів, повної взаємодії між учасниками команди виконавців, а також першого рівня зрілості організації, яка надає послуги Інтернет-провайдера, розрахунок значення показника ступеня E має вигляд:

$$E = 0,91 + 0,01 \times \sum_{j=1}^5 SF_j = 0,91 + 0,01 \times (0,03 + 0,02 + 0,05 + 0,05 + 0,01) =$$

$$= 0,91 + 0,01 \times 0,16 = 0,9116. \quad (20)$$

Значення драйверів витрат ІТ-проєкту EM_i визначалися, виходячи з таких особливостей:

- малий розмір БД;
- незначна складність продукту;
- основна увага приділяється надійності системи і, зокрема, документації;
- повторне використання коду дозволено в рамках продукту;
- можливі незначні зміни платформи розробки під час виконання проєкту;
- мінливість кадрів за рік становить менше 4 %;
- програмісти мають гарні навички та досить довго працюють над проєктом;
- кожен розробник має мінімум 4 роки досвіду виконання аналогічних проєктів;
- всі члени команди знаходяться в рамках одного будинку та мають достатню кількість засобів електронної комунікації;
- під час виконання робіт членами команди виконавців розклад цих робіт дотримується в середньому на 85 %.

Ці особливості дозволяють розрахувати значення драйверів витрат ІТ-проєкту EM_i таким чином:

$$\prod_{i=1}^7 EM_i = 0,93 \times 1,14 \times 1,11 \times 0,7 \times 0,75 \times 0,78 \times 1,1 = 0,53. \quad (21)$$

Тоді значення трудовитрат на ІТ-проєкт рефакторингу БД за розглянутими запитами на зміну буде дорівнювати

$$PM = A \times [Size']^E \times \prod_{i=1}^7 EM_i = 2,94 \times [0,031]^{0,9116} \times 0,53 = 0,065. \quad (22)$$

Таким чином, трудовитрати на рефакторинг БД інформаційно-облікової системи оцінюються у 0,065 людино-місяця.

Для розрахунку витрат часу спочатку вирахуємо значення показника масштабування для рівняння розкладу F за виразом (7). Це значення дорівнюватиме

$$F = (0,28 + 0,2 \times [E - 0,91]) = (0,28 + 0,2 \times [0,9116 - 0,91]) = 0,28032.$$

Тоді значення витрат часу на рефакторинг БД інформаційно-облікової системи дорівнюватиме (за виразом (6))

$$TDEV = \left[3,67 \times (PM_{NS})^F \right] \times \frac{SCED\%}{100} = \left[3,67 \times (0,065)^{0,28032} \right] \times \frac{110}{100} = 1,876 .$$

Таким чином, витрати часу на рефакторинг БД інформаційно-облікової системи оцінюються у 1,876 місяця.

Отримані оцінки трудовитрат і витрат часу на рефакторинг БД дозволяють позитивно відповісти на питання «Чи виправдовує результат витрачені зусилля?» для розглянутого рефакторингу БД інформаційно-облікової системи як ІТ-проєкту з реалізації двох запитів на зміни цієї системи.

6. Обговорення результатів дослідження

Запропоноване вдосконалення моделей СОСОМО II дозволяє, на відміну від загальноприйнятої версії цих моделей, оцінити трудовитрати на виконання рефакторингу БД інформаційної системи, яка експлуатується. Вибір для оцінювання доцільності проведення рефакторингу БД таких параметричних моделей, як моделі СОСОМО II, дозволяє отримати доволі точні оцінки за рахунок використання при побудові цих моделей досвіду великої кількості ІТ-проєктів різного призначення.

Використання запропонованих у дослідженні рішень дає змогу адміністратору інформаційної системи, яка експлуатується, або менеджеру проєкту отримати достатньо об'єктивні оцінки витрат на виконання подібного ІТ-проєкту рефакторингу цієї системи. Таке вдосконалення, хоча і вимагає деякого збільшення обсягу робіт під час формування і прийняття рішення щодо доцільності проведення рефакторингу БД, дозволяє приймати ці рішення, базуючись не тільки на власному досвіді та інтуїції (як рекомендується у [2, 3]), а й на результатах об'єктивного аналізу складності схеми актуальної БД та запитів на зміну системи, яка експлуатується.

Головним недоліком отриманих результатів дослідження слід вважати обмеженість під час оцінки витрат на рефакторинг БД тільки чотирма типами SQL-команд. Хоча ці команди вважаються найуживанішими під час проведення рефакторингу БД, можливо також використання інших SQL-команд, які не були враховані авторами. Ще одним недоліком отриманих результатів є їх перевірка тільки на одному ІТ-проєкті рефакторингу БД.

Виходячи із зазначених недоліків, пропонується розглядати такі подальші перспективи проведення досліджень за напрямом кількісного оцінювання витрат на рефакторинг БД інформаційної системи, яка експлуатується:

- проведення досліджень із встановлення частоти використання окремих SQL-команд в процесі рефакторингу БД;
- експериментальні перевірки запропонованого вдосконалення моделей СОСОМО II під час проведення рефакторингу БД інших інформаційних систем.

7. Висновки

У ході даного дослідження було вирішено задачу вдосконалення параметричної моделі оцінювання трудовитрат на ІТ-проєкти СОСОМО II для її використання під час оцінки витрат на рефакторинг БД. Під час вирішення цієї задачі було здійснено:

- розробку моделей (8)-(11) оцінювання кількості рядків вихідного коду SQL-команд створення, видалення, редагування та очищення, які найчастіше використовуються під час проведення рефакторингу БД;
- вдосконалення моделі СОСОМО II (4) шляхом представлення її у вигляді виразу (14), який дозволяє оцінити кількість рядків вихідного SQL-коду, які треба додати або модифікувати під час проведення рефакторингу БД;

– експериментальну перевірку розроблених і вдосконалених моделей в процесі оцінювання витрат на проведення рефакторингу БД інформаційно-облікової системи, яка експлуатується.

Перелік посилань:

1. Fowler M. Refactoring Home Page. *Refactoring*. URL: <https://www.refactoring.com/> (дата звернення: 20.04.2023).
2. Ambler S. W., Sadalage P. *Refactoring Databases: Evolutionary Database Design*. Addison-Wesley Longman, Incorporated, 2006. 384 p.
3. Database Refactoring: Improve Production Data Quality. *The Agile Data (AD) Method - Strategies for effective data-oriented development*. URL: <http://agiledata.org/essays/databaseRefactoring.html> (date of access: 07.04.2023).
4. The evolution of knowledge in the refactoring research field / M. Orrú et al. *XP 2015 Workshops: XP 2015 Scientific Workshops Proceedings*, Helsinki Finland. New York, NY, USA, 2015. <https://doi.org/10.1145/2764979.2764989>.
5. Vial G. Database Refactoring: Lessons from the Trenches. *IEEE Software*. 2015. Vol. 32. No. 6. P. 71–79. <https://doi.org/10.1109/ms.2015.131>.
6. Khumnin P., Senivongse T. SQL antipatterns detection and database refactoring process. *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, Kanazawa, Japan, 26–28 June 2017. 2017. <https://doi.org/10.1109/snpd.2017.8022723>.
7. Synthesizing database programs for schema refactoring / Y. Wang et al. *PLDI '19: 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, Phoenix AZ USA. New York, NY, USA, 2019. <https://doi.org/10.1145/3314221.3314588>.
8. Filatov V., Semenets V. Methods for Synthesis of Relational Data Model in Information Systems Reengineering Problems. *2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T)*, Kharkiv, Ukraine, 9–12 October 2018. 2018. <https://doi.org/10.1109/infocommst.2018.8632144>.
9. Bohm L. *Refactoring Legacy T-SQL for Improved Performance*. Berkeley, CA: Apress, 2020. <https://doi.org/10.1007/978-1-4842-5581-0>.
10. Fernandes I. P., Terra-Neves M., Seco J. C. Automated Refactoring of Unbounded Queries in Software Automation Platforms. *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, Fukuoka, Japan, 10–15 October 2021. 2021. <https://doi.org/10.1109/models-c53483.2021.00065>.
11. Spasić M., Jančić M. V. Verification supported refactoring of embedded SQL. *Software Quality Journal*. 2020. Vol. 29, no. 3. PP. 629–665. <https://doi.org/10.1007/s11219-020-09517-y>.
12. Managing Technical Debt in Database Normalization / M. Albarak et al. *IEEE Transactions on Software Engineering*. 2020. P. 1. <https://doi.org/10.1109/tse.2020.3001339>.
13. How do i refactor this? An empirical study on refactoring trends and topics in Stack Overflow / A. Peruma et al. *Empirical Software Engineering*. 2021. Vol. 27, no. 1. <https://doi.org/10.1007/s10664-021-10045-x>.
14. COCOMO II Model Definition Manual Version 2.1. 2000. 90 p. URL: https://www.rose-hulman.edu/class/cs/csse372/201310/Homework/CI_modelman2000.pdf (дата звернення: 12.04.2023).
15. SQL | DDL, DQL, DML, DCL and TCL Commands - GeeksforGeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/sql-ddl-dql-dml-dcl-tcl-commands/> (дата звернення: 12.04.2023).
16. SQL DDL Commands | Learn the DDL Commands of SQL. *EDUCBA*. URL: <https://www.educba.com/sql-ddl-commands/> (дата звернення: 12.04.2023).
17. Євланов М.В., Юр'єв І.О., Мірошніченко Д.О. Оцінювання доцільності проведення рефакторингу бази даних інформаційної системи, яка експлуатується. *АСУ і прилади автоматики*. 2022. Вип. 178. С. 13–22. DOI:10.30837/0135-1710.2022.178.013

Надійшла до редколегії 25.04.2023 р.

Євланов Максим Вікторович, доктор технічних наук, професор, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: maksym.ievlanov@nure.ua, ORCID: <https://orcid.org/0000-0002-6703-5166>
Юр'єв Іван Олексійович, кандидат технічних наук, доцент, доцент кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: ivan.iuriev@nure.ua, ORCID: <https://orcid.org/0000-0002-5178-519X>
Мірошніченко Дмитро Олександрович, здобувач вищої освіти, група УППІТм-21-1, факультет комп'ютерних наук, ХНУРЕ, м. Харків, Україна, e-mail: dmytro.miroshnychenko@nure.ua

M. LYTVYENKO, L. REBEZYUK

EXPERT INFORMATION PROCESSING SYSTEM FOR TRAFFIC LIGHT SYSTEM DECISION-MAKING WITH ADAPTIVE CONDITIONAL TRAM PRIORITY

The article considers the problem of taking into account the dynamics of a tram with a dedicated right of way as it approaches an intersection to provide conditional traffic signal priority with respect to the private transport flow saturation before the intersection. The expert information processing system for the intersection traffic lights system is developed. Decision-making rules regarding the adaptation of the traffic signal plan time parameters based on the interoperability of the signals and their importance have been proposed. The information processing system is implemented and validated using the SUMO urban mobility simulation tool for moderate and saturated transport demands for an artificial intersection.

1. Introduction

Recently, with the growing prevalence of personal transport in densely populated regions and the need to preserve depletable energy resources, the problem of efficiently moving large amounts of people is becoming more and more urgent. An urban transport network operates in a balance between private (PT) and public transport or mass transit (MT). In the highest demand hours, due to the limited spatial resources in the urban environment, PT is sometimes no longer able to meet the mobility needs of an urban agglomeration due to congestion [1], therefore, the minimization of personal costs (time, monetary, etc.) of moving those involved in economic processes is associated with a decrease in commute time using alternative transport links that do not depend on the conditions of private vehicles traffic. This effect, known as the Downs-Thomson paradox [2], [3], is particularly noticeable in areas with a significant share of public transport with dedicated or separate right of way (ROW) in urban commuting [4].

One of the main differences between MT with dedicated ROW and PT is that higher crowding is not associated with an increase in commute time using MT, on the other hand, it is the attractiveness of this commuting mode that may decrease (the substitutability of the modes varies according to social, geographical, etc. factors [5]). Conversely, reducing commute time, hence increasing the frequency of transport service can contribute to the attractiveness of MT, reducing the share of PT in urban mobility. Among the associated benefits of strengthening the role of MT in the transport network, in particular, via the implementation of transit signal priority, it is worth noting the growth of economic activity, improvement of the quality of the urban environment, reduction of the negative impact on the environmental conditions of an urban agglomeration, etc. [6].

In the case of intersecting traffic flows at a signalized at-grade crossing, a decision-making model for a traffic light system with a priority for public transport, specifically a tram, is intended to reduce its delay at the signals [7]. In addition to passive and active transit signal priority (TSP) [8], for a few years a distinctive real-time or adaptive priority has been considered. It iteratively computes signals sequence and their duration (traffic light plan), relying on both system-wide traffic characteristics such as time losses, halts, etc., and local MT vehicle characteristics [9]. A typical decision-making model in a traffic lights system with adaptive TSP interacts with the following components:

- detectors and interactors for MT vehicles;
- general traffic vehicles detection system;
- priority request servers and generators, control system with adaptive traffic signal plans.

The model itself implements a decision-making algorithm for traffic lights that considers the impact on general traffic and ensures pedestrians' safety. By definition, a TSP-capable traffic light will not adversely affect the coordination of traffic signals [10].

Most modern systems implementing TSP operate in a coordinated way as part of urban traffic control (UTC) systems in real time. Existing control systems with real-time TSP can be viewed as two categories [11]:

- with a constant length of traffic light cycle (rule-based) [12], [13], time parameters are gradually adapted to fluctuations in traffic conditions in real time;
- with a variable cycle length (optimization-based) [14]–[16], adaptive commands continuously optimize the traffic signal plan using the rolling horizon method [17].

Since the traffic flow characteristics definition is naturally associated with uncertainty and some degree of imprecision, and traffic at a particular intersection is controlled according to certain rules, fuzzy control may be the most natural way to solve this problem. In addition, based on expert knowledge rather than modeling a directly controlled process [18], decision-making for a traffic lights system provides control reducing computational costs. Existing studies mainly consider the application of fuzzy computing-based expert systems to adaptive decision-making in a traffic lights system in general, without a detailed focus on the TSP problem [19], [20]. The existing methods explicitly implement TSP-oriented decision-making [21], [22] based on soft computing [23] using a fixed traffic light cycle duration and/or predefined traffic light stages (phases). This inevitably imposes limitations on decision-making adaptiveness. In the papers mentioned above, the MT mainly considered does not have a dedicated ROW, therefore the use of MT vehicle dynamics data is not appropriate in these studies.

The paper [24] proposes the expert system for a signalized intersection with an unconditional tram preferential treatment that utilizes the model of a tram dynamics approaching an intersection [25], however under a high transport flow intensity, the application of the unconditional priority may cause secondary delays of MT as a result of a waiting queue length increase and impracticability of its safe discharge interruption later. The study aims to develop an expert information processing system implementing a conditional tram priority on a signalized intersection. To achieve the defined goal, it is necessary to:

- formulate the rules base for fuzzification and decision making in the expert system;
- ensure the adaptation of the system to the diversity of possible intersections, traffic flow intensities, and tram routes configurations;
- implement the conditional minimization of tram passengers' time losses and schedule deviance with respect to traffic light signals controlling the private transport traffic flow;
- ensure the system's resistance to temporary non-standard traffic conditions.

2. Decision-making model for traffic lights system with conditional tram priority

The fuzzy inference-based expert system with a tram priority consists of two levels [24]:

- level of active signals time parameters inference;
- level of the signals activations order inference.

According to the transit signal priority system definition [10], the model's second level must ensure the coordination of traffic signals and not allow simultaneous traffic flows in conflicting directions. This study utilizes a flexible, signal groups-based coordination approach [26], where

interoperability of signal groups (SG) is defined by a conflict matrix (Fig. 1), and non-zero values are visually highlighted. The values in this matrix denote the minimum required clearance interval between the pair of signal groups and are identical to those used in [24]. Such an approach allows the real-time generation of permissive signals sets depending on transport demand, minimizing unused signals which may occur during operation under the fixed, pre-defined signals sets (stage-based approach).

| ID | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|----|---|---|---|---|---|---|---|---|---|---|----|----|----|
| 0 | 0 | 2 | 1 | 2 | 2 | 0 | 1 | 0 | 2 | 1 | 1 | 0 | 2 |
| 1 | 5 | 0 | 5 | 2 | 2 | 2 | 2 | 0 | 0 | 2 | 2 | 2 | 3 |
| 2 | 1 | 2 | 0 | 2 | 2 | 1 | 1 | 0 | 2 | 0 | 1 | 2 | 2 |
| 3 | 5 | 3 | 5 | 0 | 0 | 5 | 0 | 0 | 3 | 0 | 0 | 2 | 2 |
| 4 | 5 | 3 | 5 | 0 | 0 | 5 | 0 | 0 | 3 | 0 | 5 | 2 | 0 |
| 5 | 0 | 2 | 1 | 2 | 2 | 0 | 0 | 0 | 2 | 0 | 1 | 0 | 2 |
| 6 | 1 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 1 | 2 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 5 | 0 | 5 | 3 | 3 | 5 | 5 | 0 | 0 | 5 | 5 | 3 | 3 |
| 9 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 0 |
| 10 | 1 | 2 | 1 | 2 | 2 | 1 | 1 | 0 | 2 | 0 | 0 | 2 | 2 |
| 11 | 0 | 3 | 5 | 0 | 2 | 0 | 5 | 0 | 3 | 5 | 5 | 0 | 0 |
| 12 | 5 | 4 | 5 | 2 | 0 | 5 | 0 | 0 | 4 | 0 | 5 | 0 | 0 |

Fig. 1. Conflict matrix (in seconds)

Figure 2 describes the information processing framework by the production model of decisions $s1..s3(t)$,

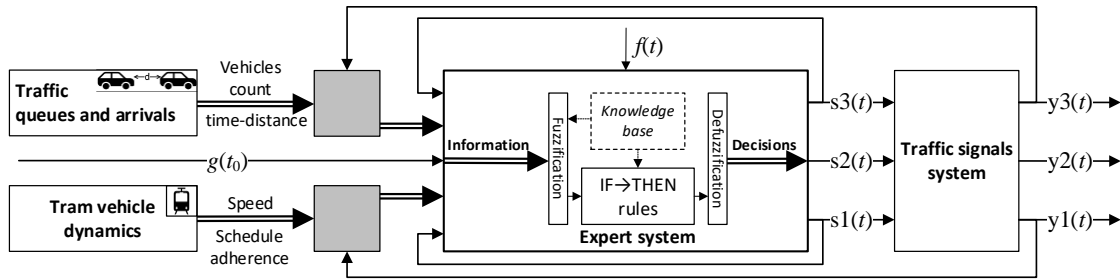


Fig. 2. Information processing framework for the expert system of traffic signal control

then passed to the traffic lights system. The expert system activates the visual signals controlling the dynamics of tram movement, PT, and pedestrian flow with separate type of visual signals provided for each category of road users denoted respectively as $y1(t)$, $y3(t)$, and $y2(t)$ while $s1(t)$, $s3(t)$, and $s2(t)$ are the outputs of the expert system. The initial conditions of the traffic lights system are defined by a parameter $g(t_0)$. In general, the expert decision being received by the traffic lights system is a set $U s_i$ of SG identifiers, treated as an active stage S_j . The decision-making model considers the PT flow and tram(s) in an active way for each s_i , and passively considers the accidental disturbance $f(t)$ caused by a pedestrian briefly affecting the dynamics of a tram.

The SG activation order is defined by their importance (weight) w only, which is inferred on the model's second level [24] using the Mamdani algorithm. Most of the time signal stages are filled up according uniquely to their weights and conflict matrix CM . Each SG can be assigned to one stage only:

$$s_i \in S_j : S_{a=k} \cap \left(\bigcup_{b \neq k} S_b \right) = \emptyset.$$

Only those SG having zero values at the corresponding rows and columns intersection of the conflict matrix are allowed to be added to the same stage:

$$\forall s_x, s_y \in S_j : CM[x, y] = 0.$$

The next stage is the one with the highest normalized sum of weights of constituent SGs:

$$S_A = \arg \max_i \left(\left\| \sum_{s_i \in S_j} w[s_i] \right\| \right).$$

The model's first level (Fig. 3) also utilizes the Mamdani inference algorithm and decides on

the SG time parameters for previously defined stages. The fuzzy production rules for the intermediate Extend or Terminate decision concerning permissive $s3_{i+}$ are defined as in the FUSICO project [27] and they depend on prohibitive SG weights sum, permissive signal duration $t(s3_{i+})$, and time gaps between PT vehicles at the entry and the traffic lane's stop line detectors $dN(s3_{i+})$ and $dT(s3_{i+})$ respectively.

To minimize tram time losses, the intermediate decision is produced both for prohibitive and permissive SG. If the number of stages exceeds 2, which is always the case except for the simplest intersections, the decision concerning action type is inferred depending on the tram SG weight as well as the current stage weight. The action types are defined as follows: run the Rapid Cycle, keep the Basic Plan, add an Extra Phase to only serve the waiting tram, perform a Green Recall to terminate the conflicting signal group(s), or Extend already running Green signal to allow the tram to proceed without a halt. If the intermediate decision being generated has the value GR or GE, this decision confidence degree is further produced by taking into consideration tram dynamics and SG weights (Fig. 4) [24].

The potential conflicts between priority requests from prohibitive SG $s1_{i-}$ during an active $s1_{i+}$ are resolved according to the rules in Table 1. For the case of a conflict with an active $s3_{i+}$ we propose the corresponding rules (Table 2).

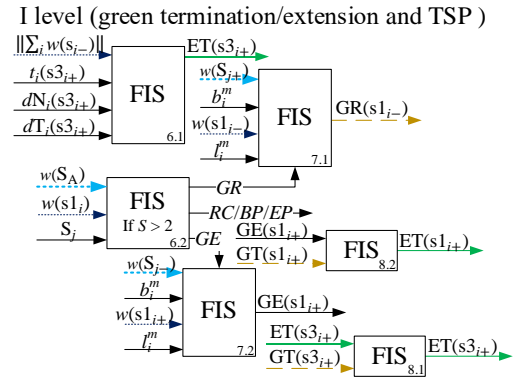


Fig. 3. Diagram of the first level of the model

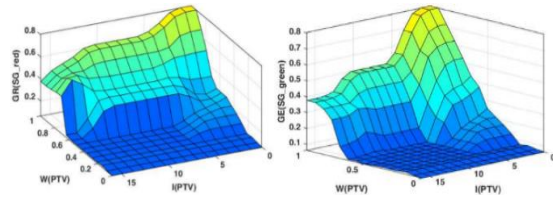


Fig. 4. Reaction functions for SG $s1_{i-}$ and $s1_{i+}$ decision-making (FIS 7.1 and 7.2)

Table 1

| GE($s1_{i+}$) | GT($s1_{i+}$) | | | | |
|-----------------|-----------------|-----------------|-----------------|-----------------|--------------|
| | Wait | RatherWait | NotSure | RatherTerminate | Terminate |
| Wait | Terminate | Terminate | Terminate | Terminate | Terminate |
| RatherWait | RatherTerminate | RatherTerminate | RatherTerminate | Terminate | Terminate |
| NotSure | NotSure | NotSure | NotSure | RatherTerminate | Terminate |
| RatherExtend | RatherExtend | RatherExtend | RatherExtend | RatherExtend | RatherExtend |
| Extend | Extend | Extend | Extend | Extend | Extend |

Table 2

| ET($s3_{i+}$) | GT($s1_{i+}$) | | | | |
|-----------------|-----------------|-----------------|-----------------|-----------------|--------------|
| | Wait | RatherWait | NotSure | RatherTerminate | Terminate |
| Terminate | Terminate | Terminate | Terminate | Terminate | Terminate |
| RatherTerminate | RatherTerminate | RatherTerminate | RatherTerminate | Terminate | Terminate |
| NotSure | NotSure | NotSure | NotSure | RatherExtend | Terminate |
| RatherExtend | RatherExtend | RatherExtend | RatherExtend | NotSure | Terminate |
| Extend | Extend | Extend | Extend | Extend | RatherExtend |

Figure 5 specifies the membership functions for fuzzy decisions regarding the extension or termination of a signal group. The threshold of decision application is defined at 0.5.

3. Modeling results analysis

For the applicability study, the decision-making system providing unconditional adaptive preferential tram treatment proposed in [24] is used as a baseline. The modeling was performed for the same intersection using the SUMO tool [28] and the demand parameters remain unchanged. More details on the experiment itself and its conditions are provided in [24].

Comparing tram time loss distributions, the application of conditional priority significantly increases the delay experienced by tram vehicles during off-peak periods (Fig. 6). However, the difference between passenger delays is not so significant (using a non-parametric test yields a p-value of 0.12). The following Table 3 generalizes the characteristics of tram traffic in non-peak conditions.

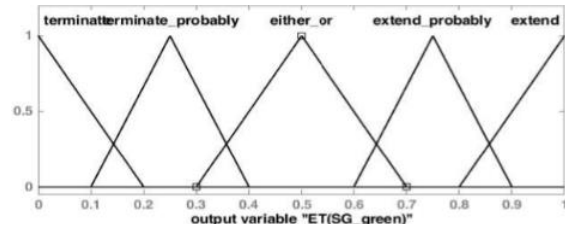


Fig. 5. Membership functions of fuzzy decision on S_{i+} .

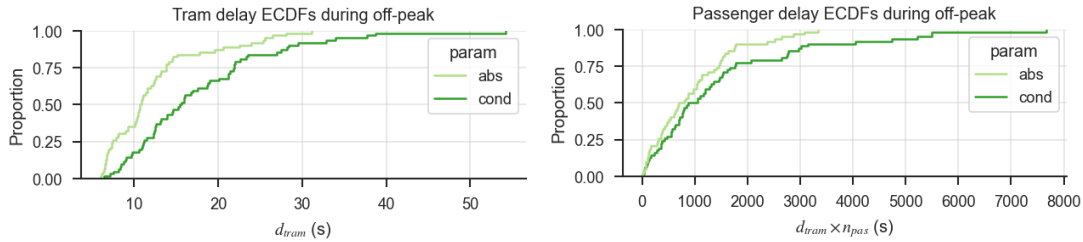


Fig. 6. Comparison of non-peak trams and passengers time losses under different priority types

The switch to conditional priority is associated with approximately a 50 % tram time loss increase, while the time loss experienced by all passengers in it increased only by 35 %. The number of complete halts is higher by 1.5 times and the crossing speed is dropped by 15 %.

The characteristics of private transport flow are compared as a difference between the vehicle queue size under unconditional (abs) and conditional priority types. To reduce minor fluctuations and have a clearer view of the general trend, the data was downsampled to 3 minutes (Fig. 7). In both cases under the off-peak demand, the difference in queue size is negligible and the mean value is around 0.

The following Table 4 is a summary of the queue length on the busiest lanes of the intersection in non-peak conditions. The conditional priority has almost no effect on the queue lengths for off-peak demand.

Table 3

| line | TramTimeLoss (s) | | PasTimeLoss (s) | | HaltsPerTram | | Speed (kph) | | priority |
|------|------------------|-------|-----------------|---------|--------------|------|-------------|------|----------|
| | median | std | median | std | mean | std | mean | std | |
| T1 | 15.89 | 8.71 | 1496.79 | 1764.84 | 0.54 | 0.58 | 18.21 | 4.01 | cond |
| T2 | 13.16 | 11.43 | 558.78 | 683.54 | 0.47 | 0.51 | 20.28 | 5.01 | cond |
| T3 | 14.62 | 6.72 | 1058.0 | 1686.0 | 0.63 | 0.5 | 20.37 | 4.2 | cond |
| T1 | 11.58 | 4.6 | 1070.16 | 868.34 | 0.19 | 0.4 | 21.21 | 2.83 | abs |
| T2 | 10.09 | 7.95 | 382.08 | 478.16 | 0.24 | 0.44 | 22.85 | 5.49 | abs |
| T3 | 7.4 | 5.3 | 821.06 | 812.94 | 0.21 | 0.42 | 24.91 | 3.67 | abs |

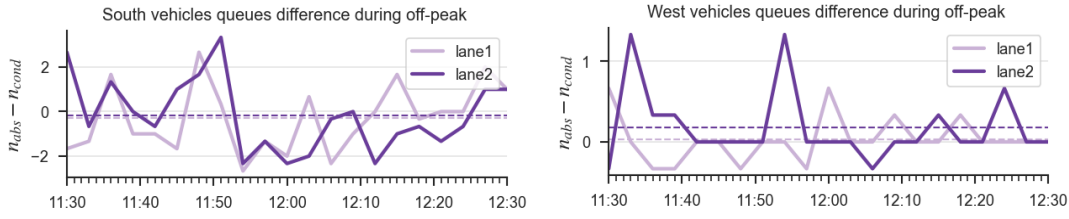


Fig. 7. General traffic time characteristics difference between different priority types under off-peak conditions

Compared to the off-peak conditions, the distribution of tram delay during the peak demand under the conditional priority remains almost the same (Fig. 8), while passenger delay ECDFs follow each other very closely and the difference is still not significant with a p-value close to 0.10.

Table 4

| origin | lane | nVehWithin | | priority | nVehWithin | | priority |
|--------|------|------------|------|----------|------------|------|----------|
| | | mean | std | | mean | std | |
| S | 1 | 2.1 | 1.47 | cond | 1.75 | 1.57 | abs |
| S | 2 | 3.08 | 2.56 | cond | 2.87 | 2.35 | abs |
| W | 1 | 0.49 | 0.72 | cond | 0.52 | 0.85 | abs |
| W | 2 | 0.34 | 0.66 | cond | 0.52 | 0.81 | abs |

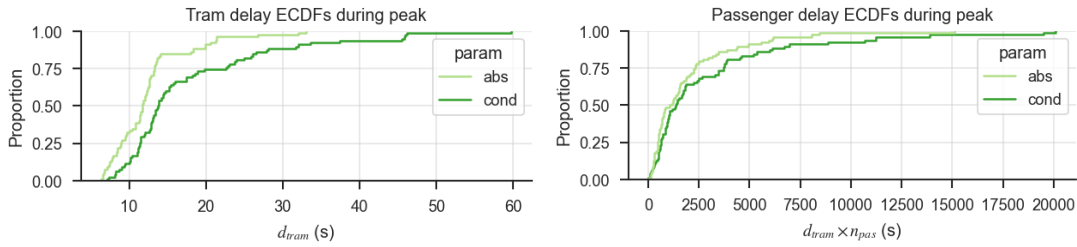


Fig. 8. Comparison of peak trams and passengers time losses under different priority types

Table 5 contains a summary of the tram traffic characteristics during the peak demand. Unlike unconditional priority, the conditional way manages to decrease the delay of trams compared to the off-peak conditions. The average increase of tram delays after switching to

Table 5

| line | TramTimeLoss (s) | | PasTimeLoss (s) | | HaltsPerTram | | Speed (kph) | | priority |
|------|------------------|-------|-----------------|---------|--------------|-----|-------------|-------|----------|
| | median | std | median | std | mean | std | median | std | |
| T1 | 14.06 | 12.19 | 1748.84 | 5394.21 | 0.38 | T1 | 14.06 | 12.19 | cond |
| T2 | 13.83 | 10.92 | 1450.8 | 2979.62 | 0.52 | T2 | 13.83 | 10.92 | cond |
| T3 | 11.48 | 6.54 | 1079.5 | 1252.35 | 0.48 | T3 | 11.48 | 6.54 | cond |
| T1 | 13.02 | 5.79 | 1785.32 | 3274.74 | 0.28 | T1 | 13.02 | 5.79 | abs |
| T2 | 10.87 | 4.2 | 820.88 | 1244.44 | 0.14 | T2 | 10.87 | 4.2 | abs |
| T3 | 9.0 | 3.39 | 698.03 | 956.78 | 0.08 | T3 | 9.0 | 3.39 | abs |

the conditional priority is less than 20%, and respective passenger time loss is increased by 43%. Compared to unconditional priority, the reduction in the speed at which trams run through the intersection decreases by 12%. Even with a slight increase in time losses relative to unconditional priority, the difference with non-adaptive reference methods presented in [24] is still considerable, e.g., compared to vehicle-actuated decision-making decrease of median tram losses is expected to be 40% and time losses of tram passengers decrease by 25%.

As for the private vehicle traffic, under the peak demand, for the busiest southern lanes, the conditional priority has a clear advantage as it reduces queue length significantly (Fig. 9). This can be seen as the average difference being above zero and in the middle of the simulation thus in the highest traffic the difference rises above 3 vehicles. For the western lanes, the difference is more centered around zero for the whole simulation.

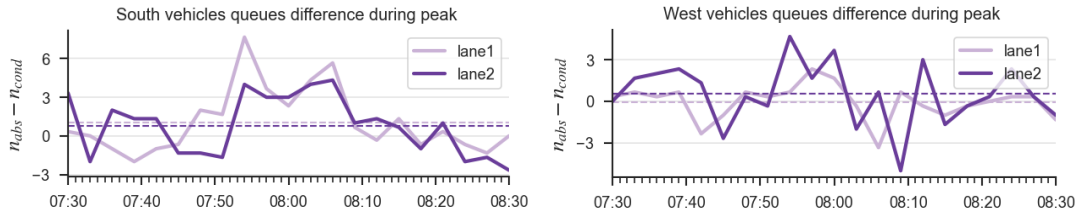


Fig. 9. General traffic time characteristics difference between different priority types under peak conditions

As shown above, the only significant decrease in average queue length and its spread is observed for the southern lanes (Table 6), which makes the application of suggested conditional priority only reasonable in very busy traffic, so that the negative impact on it is less than under the unconditional priority with comparable tram and passengers time losses.

Table 6

| origin | lane | nVehWithin | | priority | nVehWithin | | priority |
|--------|------|------------|------|----------|------------|------|----------|
| | | mean | std | | mean | std | |
| S | 1 | 3.0 | 2.28 | cond | 4.21 | 3.61 | abs |
| S | 2 | 3.53 | 2.7 | cond | 4.49 | 3.47 | abs |
| W | 1 | 1.11 | 1.72 | cond | 1.08 | 1.69 | abs |
| W | 2 | 2.59 | 2.35 | cond | 3.0 | 2.55 | abs |

Conclusions

The application of the proposed expert information processing system for a traffic light system decision-making with an adaptive conditional priority of the tram as MT with dedicated ROW is appropriate for saturated traffic conditions and allows the reduction of PT vehicles queues at traffic lights with a slight increase in tram passengers time losses, which, however, are lower than in the case of the non-adaptive decision-making.

References:

- [1] F. Zhang, R. Lindsey, and H. Yang, "The Downs–Thomson paradox with imperfect mode substitutes and alternative transit administration regimes," *Transp. Res. Part B Methodol.*, vol. 86, pp. 104–127, Apr. 2016, doi: 10.1016/j.trb.2016.01.013.
- [2] A. Downs, "The law of peak-hour expressway congestion," *Traffic Q.*, vol. 16, no. 3, 1962.
- [3] J. M. Thomson, *Great cities and their traffic*. 1978.
- [4] M. J. H. MOGRIDGE, D. J. HOLDEN, J. BIRD, and G. C. TERZIS, "The Downs/Thomson Paradox and the Transportation Planning Process," *Int. J. Transp. Econ. Riv. Internazionale Econ. Dei Tras.*, vol. 14, no. 3, pp. 283–311, 1987.
- [5] T. Litman, "Transit Price Elasticities and Cross - Elasticities," *J. Public Transp.*, vol. 7, no. 2, pp. 37–58, Jun. 2004, doi: 10.5038/2375-0901.7.2.3.
- [6] T. Litman, *Evaluating Public Transit Benefits and Costs*. Victoria Transport Policy Institute Victoria, BC, Canada, 2021. [Online]. Available: <https://www.vtpi.org/tranben.pdf>.
- [7] M. Lytvynenko, O. Shkil, I. Filippenko, and L. Rebezyuk, "Model and Means of Timed Automata-based Real-time Adaptive Transit Signal Control," presented at the 2020 IEEE East-West Design and Test Symposium, EWDTS 2020 - Proceedings, 2020. doi: 10.1109/EWDTS50664.2020.9225116.
- [8] Transportation Research Board and E. National Academies of Sciences and Medicine, "Transit Signal Priority: Current State of the Practice," The National Academies Press, Washington, DC, 2020. doi: 10.17226/25816.
- [9] H. R. Smith, B. Hemily, and M. Ivanovic, *Transit Signal Priority (TSP): A Planning and Implementation Handbook*. Washington, DC, USA: ITS America, 2005.

- [10] American Association of State Highway and Transportation Officials (AASHTO), Institute of Transportation Engineers (ITE), and National Electrical Manufacturers Association (NEMA), *NTCIP 1211 Object Definitions for Signal Control and Prioritization (SCP)*. 2014. [Online]. Available: <https://www.ntcip.org/signal-control-and-prioritization/>
- [11] N. Bhourri, F. J. Mayorano, P. A. Lotito, H. Haj Salem, and J. P. Lebacque, "Public Transport Priority for Multimodal Urban Traffic Control," *Cybern. Inf. Technol.*, vol. 15, no. 5, pp. 17–36, 2015, doi: 10.1515/cait-2015-0014.
- [12] P. P. Jiang, A. Poschinger, and T. Y. Qi, "MOTION - A Developing Urban Adaptive Traffic Signal Control System," *Adv. Mater. Res.*, vol. 779–780, Sep. 2013, doi: 10.4028/www.scientific.net/AMR.779-780.788.
- [13] F. M. Oliveira-Neto, C. F. G. Loureiro, and L. D. Han, "Active and Passive Bus Priority Strategies in Mixed Traffic Arterials Controlled by SCOOT Adaptive Signal System," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2128, no. 1, Jan. 2009, doi: 10.3141/2128-06.
- [14] J.-L. Farges and J.-J. Henry, "PT priority and PRODYN," presented at the Towards an intelligent transport system : 1st World congress on applications of transport telematics and intelligent vehicle-highway systems : Papers, Artech House, 1994, pp. 3086–3093.
- [15] P. Mirchandani, A. Knyazyan, L. Head, and W. Wu, "An Approach Towards the Integration of Bus Priority, Traffic Adaptive Signal Control, and Bus Information/Scheduling Systems," in *Lecture Notes in Economics and Mathematical Systems*, 2001, pp. 319–334. doi: 10.1007/978-3-642-56423-9_18.
- [16] J. Niittymäki and M. Mäenpää, "The Role of Fuzzy Logic Public Transport Priority in Traffic Signal Control," *Traffic Eng. Control*, vol. 42, no. 1, pp. 22–26, Jan. 2001.
- [17] N. H. Gartner, P. J. Tarnoff, and C. M. Andrews, "Evaluation of optimized policies for adaptive control strategy," *Transp. Res. Rec.*, no. 1324, pp. 105–114, 1991.
- [18] J. Niittymäki, "Fuzzy Logic Application to Public Transport Priorities at Signalized Intersections," *Math. Transp. Plan. Control*, pp. 47–57, Dec. 1998, doi: 10.1108/9780585474182-005.
- [19] S. M. Odeh, A. M. Mora, M. N. Moreno, and J. J. Merelo, "A Hybrid Fuzzy Genetic Algorithm for an Adaptive Traffic Signal System," *Adv. Fuzzy Syst.*, vol. 2015, p. e378156, Sep. 2015, doi: 10.1155/2015/378156.
- [20] A. Atta, S. Abbas, M. A. Khan, G. Ahmed, and U. Farooq, "An adaptive approach: Smart traffic congestion control system," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 32, no. 9, pp. 1012–1019, Nov. 2020, doi: 10.1016/j.jksuci.2018.10.011.
- [21] M. S. Ghanim and G. Abu-Lebdeh, "Real-Time Dynamic Transit Signal Priority Optimization for Coordinated Traffic Networks Using Genetic Algorithms and Artificial Neural Networks," *J. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 327–338, Oct. 2015, doi: 10.1080/15472450.2014.936292.
- [22] X. Kuang and L. Xu, "Real-Time Traffic Signal Intelligent Control with Transit-Priority," *J. Softw.*, vol. 7, no. 8, pp. 1738–1743, Aug. 2012, doi: 10.4304/jsw.7.8.1738-1743.
- [23] "A Definition of Soft Computing - adapted from L.A. Zadeh." <http://www.soft-computing.de/def.html>
- [24] M. Lytvynenko and L. Rebezyuk, "Mathematical model of acyclic fuzzy control for traffic signal system with adaptive unconditional tram priority," *Manag. Inf. Syst. Devices*, no. 178, pp. 23–32, 2022.
- [25] L. Do, I. Herman, and Z. Hurák, "Onboard Model-based Prediction of Tram Braking Distance," *IFAC-Pap.*, vol. 53, no. 2, pp. 15047–15052, Jan. 2020, doi: 10.1016/J.IFACOL.2020.12.2006.
- [26] B. G. Heydecker, "A decomposition approach for signal optimization in road networks," *Transp. Res. Part B Methodol.*, vol. 30, no. 2, pp. 99–114, Apr. 1996, doi: 10.1016/0191-2615(95)00025-9.
- [27] J. Niittymäki and M. Pursula, "Signal control using fuzzy logic," *Fuzzy Sets Syst.*, vol. 116, no. 1, pp. 11–22, Nov. 2000, doi: 10.1016/S0165-0114(99)00034-2.
- [28] P. A. Lopez *et al.*, "Microscopic Traffic Simulation using SUMO," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2018, pp. 2575–2582. doi: 10.1109/ITSC.2018.8569938.

Надійшла до редколегії 21.08.2022

Lytvynenko Mykhailo, Master of Automation and Computer-integrated Technologies (Systems Engineering Department) NURE. Address: Nauky Ave., 14, Kharkiv, Ukraine, 61166; tel. +380 98 915 14 95; email: mykhailo.lytvynenko1@nure.ua

Rebezyuk Leonid, Associate Professor of Systems Engineering Department NURE, Candidate of Technical Sciences, Associate Professor. Address: Nauky Ave., 14, Kharkiv, Ukraine, 61166; tel. +380 57 702 10 06, +380 68 961 36 76, email: leonid.rebezyuk@nure.ua.

МОДЕРНІЗАЦІЯ СПЕЦІАЛІЗОВАНИХ ІНФОРМАЦІЙНИХ СИСТЕМ СЛУЖБИ КРОВІ

На основі аналізу складових спеціалізованих медичних інформаційних систем служби крові, показників діяльності служби крові; складових інформаційного супроводу бізнес-процесів служби крові; витратних ресурсів на модернізацію спеціалізованих медичних інформаційних систем запропоновано математичний апарат пошуку оптимального варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем служби крові. Формалізоване представлення задачі пошуку оптимального варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем служби крові дозволяє визначити варіант максимального покращення показників діяльності закладу служби крові за рахунок зменшення частки ручних операцій при введенні даних під час виконання бізнес-процесів за допомогою спеціалізованої медичної інформаційної системи з урахуванням обмежень витратних ресурсів на модернізацію спеціалізованої медичної інформаційної системи.

1. Вступ

Інформаційні системи служби крові можуть бути віднесені до спеціалізованих медичних інформаційних систем (СМІС), ефективність функціонування яких безпосередньо впливає на ключові соціально-значущі показники діяльності (КСЗПД) підприємства. В першу чергу, до КСЗПД служби крові відносять: кількість випадків ускладнень, заражень пацієнта інфекціями, що передаються через донорську кров та її компоненти (гепатит В, гепатит С, ВІЛ 1\2, сифіліс); кількість виявлених випадків побічних реакцій у донорів після здачі крові; кількість списаних та утилізованих доз компонентів крові. На різних етапах бізнес-процесу (БП) служби крові значний вплив на ці показники мають помилки співробітників на робочих місцях при внесенні інформації до СМІС, а саме: помилки введення результатів лабораторних досліджень, ідентифікації донора та компонентів крові; помилки контролю допущення донора до здачі крові; помилки введення даних під час виконання дій від забору крові до видачі компонентів крові медичним закладам. Зменшення кількості помилок і, як наслідок, поліпшення наведених КСЗПД, є метою вдосконалення функціонування СМІС служби крові. Таке вдосконалення здійснюється в результаті модернізації СМІС як одного з найбільш потужних способів підвищення ефективності виробничої діяльності підприємства.

2. Аналіз літературних джерел та визначення проблеми дослідження

Модернізація СМІС служби крові може відбуватися за однією з відомих схем: заміною існуючої системи на нову, досконалішу; поліпшенням та/або додаванням функціональних можливостей існуючої системи; зміною (повною або частковою) структур програмного забезпечення, технічного забезпечення, бази даних; переробкою інтерфейсу користувача [1–2]. БП служби крові, який супроводжується СМІС, є достатньо складним і потребує втручання професійного персоналу впродовж усього ланцюга створення та розподілення компонентів крові – від реєстрації донора та оцінювання його можливості здавати кров до зберігання та передачі компонентів крові в лікувальні заклади [3]. Таким чином, достовірність інформації, що надається СМІС під час виконання БП персоналом служби крові, залежить як від стану персоналу і його професійних навичок, так і від частки ручних операцій на кожному робочому

місці. Зменшення частки ручних операцій безпосередньо залежить від рівня автоматизації БП на робочих місцях і може розглядатись як задача модернізації СМІС.

Причинами виникнення потреби модернізації СМІС служби крові можуть стати такі фактори:

- ітераційний процес проектування, при якому відбувається повернення до попереднього етапу проектування для внесення змін, уточнень або виправлення допущених помилок при експлуатації системи;
- використання різних програмно-апаратних засобів для реалізації окремих частин СМІС служби крові;
- необхідність взаємодії з іншими ІС, що вимагає створення додаткових інтерфейсів;
- розвиток програмно-апаратних засобів, що приводить до оновлення вже створених компонентів СМІС служби крові;
- зміна потреб користувача в ході побудови або експлуатації СМІС служби крові;
- зміни в правовій сфері служби крові [4–5].

Основним способом спрощення процесу модернізації СМІС та скорочення витрат на модернізацію СМІС є декомпозиція СМІС на складові [6–7]. Це в подальшому дає можливість визначати складові СМІС, що потребують першочергової модернізації з максимальним ефектом. СМІС може бути декомпозована на підсистеми, модулі, компоненти або об'єкти. Поширені такі методи декомпозиції:

- декомпозиція на модулі, застосовується при структурному підході;
- декомпозиція на об'єкти, застосовується при побудові об'єктно-орієнтованих інформаційних систем (ІС) [8].

При моделюванні СМІС служби крові пропонується декомпозиція СМІС на організаційно-технічні компоненти (ОТК). Кожен ОТК укомплектовано відповідними елементами організаційного, інформаційного, технічного, програмного та інших видів забезпечень, які визначають ступінь автоматизації введення даних під час виконання певних дій БП, і, відповідно, визначають ступінь автоматизації самого ОТК. Часткова або повна відсутність належних елементів організаційного, інформаційного, технічного, програмного забезпечення призводить до випадків введення даних з помилками під час ручних операцій з введення та обробки даних [9].

Підвищення ступеня автоматизації введення даних в ОТК дозволяє знизити ймовірність помилок при введенні даних персоналом, знизити ризики впливу людського фактору на достовірність даних, що може впливати на підвищення ефективності СМІС, що використовується, та, відповідно, покращити фактичні значення КСЗПД служби крові.

3. Мета і задачі дослідження

У статті будуть розглянуті актуальні питання модернізації СМІС служби крові, пов'язані з вдосконаленням апаратно-програмного забезпечення СМІС служби крові.

Метою дослідження є формалізоване представлення вирішення задачі пошуку оптимального варіанту проектного рішення модернізації СМІС служби крові для подальшого використання як для пошуку рішення з модернізації всієї СМІС, так і для модернізації її частини (ОТК), що забезпечить максимальне покращення КСЗПД центра служби крові в межах визначених витратних ресурсів на реалізацію модернізації.

СМІС доцільно розглядати в розрізі процесно-орієнтованої організації з декомпозицією на ОТК з урахуванням принципів структурної організації спеціалізованої ІС. При цьому мають бути взяті до уваги такі питання:

- параметри інформаційного супроводу БП спеціалізованого підприємства, що здійснюють вплив на КСЗПД підприємства потребують формалізації;
- математичний апарат моделювання СМІС потребує розвитку відомого процесно-

орієнтованого підходу вибору оптимального варіанту проектного рішення модернізації СМІС;
 – математичний апарат визначення оптимального варіанту проектного рішення модернізації СМІС потребує уточнення з урахуванням загальновідомого підходу критеріального оцінювання з обмеженнями.

4. Матеріали і методи дослідження

Для вирішення задачі модернізації СМІС служби крові необхідно проведення аналізу поточного стану її функціонування, визначення вимог до модернізації СМІС, визначення варіанту проектного рішення. Схема вирішення задачі модернізації СМІС служби крові наведена на рис. 1.

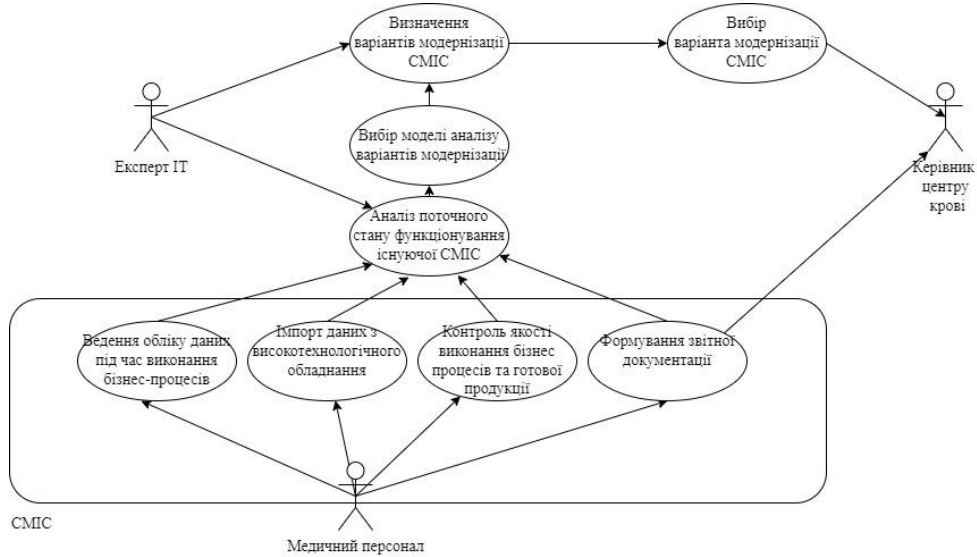


Рис. 1. Схема вирішення задачі модернізації СМІС служби крові

Потрібно знайти такий варіант v_{opt} проектного рішення з модернізації СМІС серед потенційно можливих, який, по-перше відповідає вимогам щодо модернізації, а по-друге, враховує обмеження бюджетування. Під v_{opt} будемо розуміти варіант модернізації спеціалізованої ІС, для якого може бути забезпечено максимальне поліпшення КСЗПД та сукупного показника діяльності підприємства. Поліпшення КСЗПД можливе як за рахунок вдосконалення БП, так і за рахунок збільшення достовірності даних, що використовує СМІС при інформаційному супроводі БП. Числові показники достовірності даних, що вводяться до СМІС за допомогою ОТК, уможливають облік впливу цих даних на КСЗПД шляхом розрахунку відповідного інтегрованого показника функціонування СМІС FI^{v_n} , що корелюється з сукупним показником діяльності спеціалізованого підприємства.

Таким чином, на підставі відомої інформації, а саме, даних про КСЗПД; складових БП; даних інформаційного супроводу БП; складових СМІС; вузьких критичних дій БП; обмежень витратних ресурсів на розвиток СМІС формально задачу пошуку оптимального варіанту проектного рішення модернізації СМІС v_{opt} може бути задано у такому вигляді:

$$v_{opt} \rightarrow \max \left(\sum_{\varepsilon=1}^E \tau_{\varepsilon} \text{ind}_{\varepsilon} \right), \quad (1)$$

де V_{opt} – шуканий варіант проєктного рішення модернізації СМІС; τ_ε – коефіцієнт значності ε -го КСЗПД підприємства; $\varepsilon = \overline{1, E}$; ind_ε – фактичний ε -й КСЗПД підприємства.

Оптимальний варіант проєктного рішення модернізації СМІС V_{opt} можна представити як:

$$V_{opt} \in V, \quad (2)$$

$$V = \{v_1, \dots, v_n, \dots, v_N\},$$

де V – множина всіх можливих варіантів проєктних рішень модернізації СМІС; v_n – n -й варіант проєктного рішення модернізації СМІС; n – індекс порядкового номеру варіанта проєктного рішення модернізації СМІС, $n = \overline{1, N}$.

При цьому

$$v_n = \langle \Delta WP, c^{v_n} \rangle, \quad (3)$$

де ΔWP – зміна елементів організаційного, інформаційного, програмного, технічного забезпечення ОТК СМІС, яку, в свою чергу, можна представити як $\Delta WP = \{\Delta wp_1, \dots, \Delta wp_\omega, \dots, \Delta wp_\Omega\}$ – множина змін елементів організаційного, інформаційного, програмного, технічного забезпечень на кожному ОТК СМІС.

ОТК представлено як

$$wp_\omega = \langle F_\omega, S_\omega, DS_\omega, Sp_\omega \rangle, \quad (4)$$

де wp_ω – ω -те ОТК; ω – індекс порядкового номеру ОТК, $\omega = \overline{1, \Omega}$; F_ω – множина функцій СМІС, що забезпечують ω -й ОТК СМІС; S_ω – множина елементів організаційного, програмного, технічного забезпечення для реалізації множини функцій F_ω за допомогою ω -го ОТК СМІС; DS_ω – дані, що вводяться до СМІС за допомогою ω -го ОТК; Sp_ω – частковий показник функціонування ω -го ОТК СМІС.

Досліджено, що фактичні значення КСЗПД підприємства корелюють з частковими показниками функціонування СМІС [10]:

$$\begin{aligned} \sum_{\varepsilon=1}^E \tau_\varepsilon ind_\varepsilon &\rightarrow \sum_{\varepsilon=1}^E \tau_\varepsilon Sp_\varepsilon, \\ \sum_{\varepsilon=1}^E \tau_\varepsilon Sp_\varepsilon &= \sum_{\varepsilon=1}^E \tau_\varepsilon \sum_{\omega=1}^{\Omega} Sp_\omega = Fl^{v_n}, \\ Fl^{v_{opt}} &= \max \{ Fl^{v_n} \}, \end{aligned} \quad (5)$$

де Sp_ε – частковий показник функціонування СМІС з урахуванням впливу на ε -й КСЗПД підприємства; Fl^{v_n} – інтегрований показник функціонування v_n варіанта СМІС.

При цьому необхідно забезпечити виконання таких умов:

$$0 < \sum_{\omega=1}^{\Omega} Sp_{\omega} < 1;$$

$$0 < \sum_{\varepsilon=1}^E \tau_{\varepsilon} Sp_{\varepsilon} < 1; \quad (6)$$

$$\sum_{\omega=1}^{\Omega} Sp_{\omega} V_n \geq \sum_{\omega=1}^{\Omega} Sp_{\omega} V_{cur};$$

$$C^{V_n} \leq C_{lim},$$

де C^{V_n} – витратні ресурси на реалізацію v_n варіанта проектного рішення модернізації СМІС; C_{lim} – обмеження витратних ресурсів на модернізацію СМІС; v_{cur} – існуючий варіант конфігурації діючої СМІС, що визначає поточний стан автоматизації.

5. Результати дослідження

Кожен ОТК має поточний ступінь автоматизації. Відповідно, для кожного ОТК необхідно розглянути можливі конфігурації організаційного, програмного, технічного забезпечення, що забезпечать кожен наступний за поточним ступінь автоматизації введення даних. Таким чином, оптимальний варіант модернізації СМІС треба обирати з сукупності всіх ОТК з відповідними комбінаціями підвищення ступенів автоматизації кожного з них (див. рис.2).

В свою чергу, ступінь автоматизації кожного ОТК може бути змінено за рахунок відповідної зміни елементів організаційного, інформаційного, програмного, технічного забезпечення ОТК таким чином, щоб забезпечити максимальний інтегрований

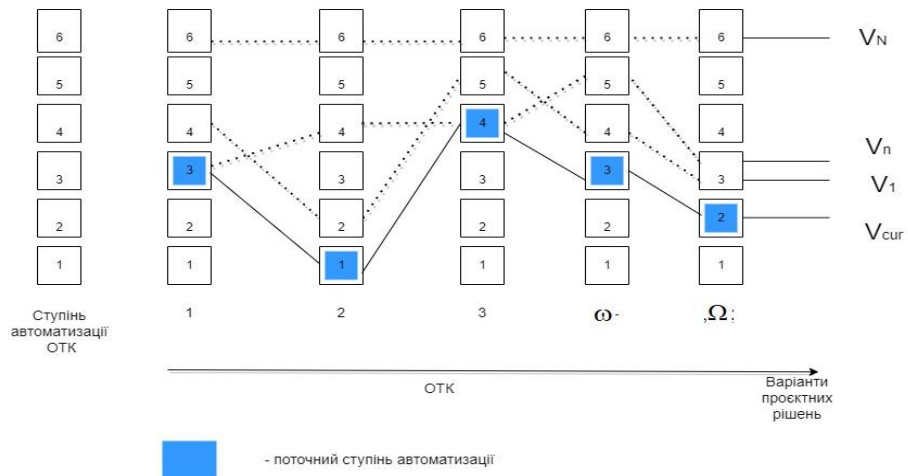


Рис. 2. Варіанти проектних рішень модернізації СМІС

показник функціонування СМІС Fl^{V_n} , що корелює з фактичними КСЗПД підприємства ind_{ε} .

ER-модель модернізації СМІС наведена на рис.3.

Для практичної реалізації пошуку оптимального варіанту модернізації СМІС може бути застосовано один з відомих алгоритмів, наприклад, модуль «Пошук рішення» (MS Excel), що дозволяє вирішувати лінійні задачі оптимізації [11–12].

6. Обговорення результатів дослідження

Отриманий математичний апарат модернізації СМІС ґрунтується на попередньо проведених дослідженнях з аналізу структури БП підприємства, процесно-орієнтованої структури СМІС з декомпозицією на ОТК, впливу ступеню автоматизації введення даних на КСЗПД діяльності підприємства. Запропонована ER-модель модернізації СМІС дає можливість формалізовано представити всі складові для побудови варіантів проектних рішень модернізації СМІС.

медичної сестри операційної може бути підвищено з першого до шостого; ступінь автоматизації ОТК техніка – з другого до четвертого; ступінь автоматизації ОТК медичної сестри фракціонування може бути підвищено з першого до шостого ступеня; ступінь автоматизації ОТК медичного реєстратора апробації може бути підвищено з третього до шостого. Така модернізація відповідних ОТК СМІС, в свою чергу, дала можливість покращення значень КСЗПД, а саме: скорочення кількості випадків інфікування пацієнтів гепатитом В, гепатитом С, сифілісом, випадків ускладнень з причини несумісності групи крові в порівнянні з попереднім роком.

7. Висновки та перспективи подальшого розвитку

Наданий математичний апарат модернізації СМІС в першу чергу може бути застосовано на підприємствах з БП, де передбачається ручне введення важливих поточних даних з безпосереднім впливом на якість послуг, що надаються, або продукції, та дозволяє удосконалювати СМІС підприємства завдяки підвищенню ступеню автоматизації при пошуку оптимального варіанту модернізації ОТК ІС.

Важливими питаннями при реалізації задачі пошуку оптимального варіанта модернізації СМІС є формування експертних груп при оцінюванні як поточного стану СМІС в цілому, так і визначенні можливих варіантів модернізації.

Перелік посилань

1. Реінжиніринг програмного забезпечення інформаційних систем: монографія/ С. С. Великодний, О. С. Тимофєєва. Одеса: Видавничий дім «Гельветика», 2020. 160 с.
2. Моделі та методи проактивного управління проектами з розвитку програмних систем і продуктів: монографія. Одеса: Одеський державний екологічний університет, 2021. 322 с.
3. WHO. Global status report on blood safety and availability 2021. Geneva: World Health Organization; 2022. 184 p.
4. ДСТУ ISO 9001–2015. Системи управління якістю. Вимоги. К.: Держстандарт України. 22 с.
5. Примак С.В., Любінець О.В., Новак В.Л. Структурно-процесний підхід до покращення якості та ефективності гематологічної служби в Україні. *Здоров'я нації*. 2022. № 3 (69) С. 53–59.
6. Палагін В. В., Палагіна О. А., Гаген В. А. Визначення критеріїв ефективності при розробці медичних інформаційних систем. *Herald of Khmelnytskyi national university*. 2021. Issue 3. (297) С. 116–123.
7. Perceptions and experiences with district health information system software to collect and utilize health data in Bangladesh: a qualitative exploratory study. *BMC Health Services Research*. 2020. 465 P. 1 – 6.
8. Єганов О. Ю., Ажищев В. Ф., Погорелова О. В., Баланенко О. Г. Інформаційні системи і технології обліку: монографія. 2-е видання. Варшава: RS Global Sp. z O.O., 2020. 131 с.
9. Міхнова А.В. Міхнов, Д.К., Чиркова К.С. Модель спеціалізованої медичної інформаційної системи служби крові. *Науково-технічний журнал «Вісник Кременчуцького національного університету імені Михайла Остроградського*. 2019. № 5 (118). С. 75–82.
10. Mikhnova A., Mikhnov D., Chyrkova K. Development the Technology of Reengineering Specialized Information Systems. *International Academy Journal Web of Scholar*. 2021. №1 (51). P. 1–6.
11. Mikhnova A., Mikhnov D., Chyrkova K. Method for evaluating the efficiency of upgrading specialized information systems. *Науковий журнал «Сучасний стан наукових досліджень та технологій в промисловості»*. 2019. № 4 (10). С. 69–76.
12. Чеботарьова Д. В. Автоматизація вибору оптимальних проектних варіантів систем зв'язку на основі методів багатокритеріальної оптимізації. *Інфокомунікаційні технології та електронна інженерія*. 2021. № 2 (2). С. 54–61.

Надійшла до редколегії 10.08.2023 р.

Міхнова Аліна Володимирівна, кандидат технічних наук, доцент, доцент кафедри ІУС ХНУРЕ, м. Харків, Україна; e-mail: alina.mikhnova@nure.ua; ORCID: <https://orcid.org/0000-0001-9877-4298>.

Міхнов Дмитро Кіндратович, кандидат технічних наук, доцент, професор кафедри ІУС ХНУРЕ, м. Харків, Україна; e-mail: dmytro.mikhnov@nure.ua; ORCID: <https://orcid.org/0000-0002-9940-8553>.

Чиркова Катерина Сергіївна, кандидат технічних наук, старший викладач кафедри ІУС ХНУРЕ, м. Харків, Україна; e-mail: katernyna.chyrkova@nure.ua; ORCID: <https://orcid.org/0000-0002-3749-3043>.

П.Е. СИТНИКОВА, М.О. ГРЕБЕНЮК

РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ КОМПАКТНОЇ ГІБРИДНОЇ МОДЕЛІ КОРИСТУВАЧА

Запропоновано компактну гібридну модель користувача для методу спільної фільтрації інформації. Ця модель долає обмеження, які часто виникають при використанні традиційних підходів, і дозволяє ефективніше генерувати персоналізовані рекомендації. Для наочного прикладу цієї моделі було взято рекомендаційну систему кінофільмів. Запропонована модель пов'язує оцінки користувачів з деяким описом вмісту об'єктів. Для цього введено поняття міри цікавості жанру, що являє собою гібридну ознаку, яка поєднує оцінки користувачів і жанри фільмів, та представляє уподобання користувача на рівні моделі. Таким чином, запропонована модель зберігає точність спільної фільтрації на основі пам'яті та масштабованість на основі моделі. Це сприяє значному зменшенню складності системи, її розрідженості та забезпечує виконання взаємовідношення транзитивності між сусідами.

1 Вступ

У сучасному цифровому світі, коли обсяг інформації та доступного матеріалу надзвичайно великий, рекомендаційні системи грають важливу роль у полегшенні процесу вибору продуктів, послуг та контенту для користувачів. Вони допомагають надавати індивідуальні рекомендації, враховуючи уподобання та інтереси користувачів, що робить процес прийняття рішень зручнішим та ефективнішим. Рекомендаційні системи використовують різні алгоритми та методи для аналізу даних і надання персоналізованих рекомендацій. Такі системи діють в різних галузях, включаючи електронну комерцію, музику, фільми, книги та інші області. Вони можуть базуватися на аналізі історії оцінок користувача, на основі схожості переглянутих об'єктів, які знаходяться у системі, та на інших даних.

У цьому контексті однією з важливих складових вподобань користувачів є така ознака класифікації об'єктів, як жанри фільмів, музичних композицій, літератури тощо. Знання жанру допомагає користувачам знаходити той контент, який їх цікавить, і зробити вибір серед маси доступних альтернатив. Проте, для ефективної роботи рекомендаційних систем необхідно враховувати не лише жанр, але й інші фактори, такі як історія взаємодій користувачів, рейтинги, відгуки та інші параметри.

Тут на допомогу приходять гібридні рекомендаційні системи, які поєднують різні підходи та методи, включаючи інформацію про жанри, для надання більш точних та персоналізованих рекомендацій. Гібридні системи дозволяють враховувати індивідуальні смаки та вподобання користувачів, а також забезпечують більш масштабовані та ефективні рекомендації, навіть у випадках, коли об'єм доступного контенту дуже великий.

Таким чином, гібридні рекомендаційні системи важливі для рекомендацій об'єктів з різними жанрами, оскільки вони допомагають підвищити точність, рівень персоналізації та релевантність рекомендацій для користувачів, що робить процес вибору контенту більш зручним і задовольняє індивідуальні потреби кожного користувача. Тому розробка саме таких систем є актуальною і важливою задачею та має практичну значущість.

2. Аналіз існуючих методів і постановка проблеми

Для керування обчисленням подібності на основі доступних вхідних даних існує багато методів фільтрації інформації. В загальному випадку кожен метод фільтрації можна віднести до одного з таких класів: демографічний, на основі вмісту, спільний або гібридний [2]. Далі розглянемо різні методи фільтрації інформації з відповідними вхідними даними (табл. 1).

Таблиця 1

Методи фільтрації інформації

| Метод фільтрації інформації | Демографічна інформація | Опис вмісту | Рейтинг |
|-----------------------------------|-------------------------|-------------|---------|
| Демографічна фільтрація (DF) | ✓ | | |
| Фільтрація на основі вмісту (CBF) | | ✓ | |
| Спільна фільтрація (CF) | | | ✓ |
| Гібридна фільтрація | DF/CBF | ✓ | ✓ |
| | DF/CF | ✓ | ✓ |
| | CBF/CF | | ✓ |
| | DF/CBF/CF | ✓ | ✓ |

Демографічна фільтрація (DF) [1] використовує описи користувачів для того, щоб визначити взаємозв'язок між конкретним об'єктом і типом користувачів, яким він подобається. Відповідно, користувачеві будуть рекомендовані товари, подібні до тих, які вподобують інші люди з тієї ж демографічної групи. У [3] розглядається саме цей підхід для генерації рекомендацій. Перевага демографічного підходу полягає в тому, що він може не вимагати історії рейтингів, необхідної для спільної фільтрації та фільтрації на основі вмісту. Основні недоліки демографічної фільтрації такі:

– труднощі отримання даних. Важко отримати особисту інформацію від користувачів. Вони зазвичай залишаються анонімними або не розголошують свої особисті дані [4]. Здебільшого це пов'язано з недовірою до політики конфіденційності сучасних веб-сайтів. Таким чином, користувачі або приховують персональні дані, або надають неправдиві дані;

– проблема сірої вівці. Це стосується користувачів з унікальними вподобаннями та смаками, які ускладнюють розробку точних профілів. Вирішення саме цієї проблеми запропонували у [5].

Фільтрація на основі вмісту (CBF) використовує описи вмісту об'єктів [1], які сподобалися користувачеві в минулому, щоб зробити для нього нові рекомендації. Тому йому будуть рекомендовані об'єкти, схожі на ті, яким він віддавав перевагу в минулому. Цей метод визначає об'єкти за пов'язаними з ними ознаками. Наприклад, у [6] для рекомендацій використовуються теги та жанр музики. Рекомендаційні системи на основі вмісту не підходять для динамічних і дуже великих середовищ, де об'єктів мільйони і вони часто додаються в систему. Основними недоліками фільтрації на основі вмісту є:

– обмеження вмісту. Можна надати лише дуже поверхневий аналіз певних видів контенту;

– надмірна спеціалізація [2]. Користувач може бачити лише предмети, подібні до вже оцінених ним. Це обмеження не дозволяє користувачеві отримувати випадкові рекомендації;

– проблема нового користувача. Система повинна дізнатися уподобання користувача з описів об'єктів, які йому подобалися раніше. Тому користувач повинен оцінити деякі об'єкти, щоб система могла дійсно зрозуміти вподобання користувача і дати надійні та точні рекомендації.

Спільна фільтрація (CF) [1] стимулює соціальний процес пошуку у сусідів рекомендацій щодо об'єктів, які раніше не були розглянуті користувачем. Таким чином, будуть рекомендовані предмети, які подобалися людям зі схожими смаками та уподобаннями в минулому. Велика сила спільної фільтрації по відношенню до фільтрації на основі вмісту полягає в тому, що вона не потребує апріорної інформації про властивості об'єктів (наприклад, жанр фільму чи тип продукту) і здатна пропонувати рекомендації, які можуть бути неочевидними чи "нестандартними". Більше того, спільна фільтрація є повністю незалежною від будь-якого машиночитаного представлення рекомендованих об'єктів. На відміну від фільтрації на основі вмісту, спільна та демографічна фільтрація можуть надавати випадкові рекомендації. Однак спільна фільтрація окрім вже згаданих проблеми нового користувача та проблеми сірої вівці має ще такі слабкі сторони:

– проблема нового об'єкта. Новий об'єкт, який має дуже малу кількість оцінок, не можна легко рекомендувати. Ця проблема також відома як проблема «холодного старту», оскільки перший користувач, який оцінить об'єкт, отримує від цього незначну користь. Такі ранні оцінки не дозволяють користувачам ефективно порівнювати свої вподобання з вподобаннями інших користувачами;

– розрідженість. Зазвичай кожен користувач оцінює лише дуже обмежений відсоток об'єктів у порівнянні з наявною загальною кількістю. Це призводить до розріджених матриць "користувач-об'єкт", і тому можуть бути створені слабкі рекомендації, оскільки успішні сусіди не можуть бути знайдені. Тому було запропоновано багато методів зменшення розмірів для зменшення ефекту розрідженості (наприклад, у [7]);

– втрата транзитивності сусідів. Пряма опора на рейтинги об'єктів призводить до втрати транзитивності сусідів. Припустимо, що ми маємо трьох користувачів u_i , u_j і u_k . Користувачі u_i і u_j дуже корелюють, також u_j і u_k дуже корелюють. Через транзитивність існує ймовірність того, що користувачі u_i та u_k також сильно корелюють. Такий перехідний зв'язок не відображається в традиційних методах спільної фільтрації, якщо користувачі u_i та u_k не оцінили багато спільних об'єктів.

Гібридні системи [1] поєднують два або більше методів фільтрації, які доповнюють один одного для досягнення кращої продуктивності.

У таблицях 2, 3, 4 наведено зразки профілів для користувача [8] відповідно до демографічної фільтрації, фільтрації на основі вмісту та спільної фільтрації.

Таблиця 2

Профіль користувача для демографічної фільтрації

| № | Пол | Вік | Професія | Освіта | ... |
|-----|---------|-----|------------|---------|-----|
| 1 | Чоловік | 23 | Програміст | Магістр | ... |
| ... | ... | ... | ... | ... | ... |

Таблиця 3

Профіль користувача для фільтрації на основі вмісту

| № | Режисер | Актор | Рік | Країна | ... |
|-----|-----------------|------------------|------|--------|-----|
| 1 | Крістофер Нолан | Меттью Макконехі | 2014 | США | ... |
| ... | ... | ... | ... | ... | ... |

Таблиця 4

Профіль користувача для спільної фільтрації

| № | Титанік | Зоряні війни | Месники | Форсаж | ... |
|-----|---------|--------------|---------|--------|-----|
| 1 | 5 | 4 | 2 | 3 | ... |
| ... | ... | ... | ... | ... | ... |

Різниця між профілем користувача та моделлю користувача полягає в різному рівні складності. Профіль користувача — це просто сукупність інформації, зібраної від користувача, яку можна описати простою моделлю. Залежно від змісту та кількості інформації про користувача, яка зберігається в профілі, можна побудувати модель. Точніше, модель користувача — це уявлення про знання та особистісні характеристики, якими, на думку системи, володіє користувач. Відповідно, рекомендаційні системи можна класифікувати на системи на основі пам'яті або на основі моделі [2]. Для того, щоб побудувати модель користувача, необхідно вирішити, які конкретні ознаки або параметри будуть включені до моделі користувача для того, щоб рекомендаційна система могла краще розуміти і передбачати інтереси та потреби користувача. Від цього залежить якість та ефективність системи рекомендацій. Рекомендаційні системи на основі пам'яті використовують просту модель та весь набір даних для процесу рекомендацій. Хоча система, що базується на пам'яті, є простою і надає рекомендації високої точності, вона є дорогою з точки зору обчислень, оскільки розмір вхідного набору даних збільшується. З іншого боку, рекомендаційна система на основі моделі отримує модель з офлайн-даних, які будуть використовуватися для онлайн-рекомендацій. Після формування моделі користувача матриця об'єктів користувача більше не потрібна для генерації рекомендацій. Це призведе до систем із нижчими вимогами до пам'яті та швидкого генерування рекомендацій. Хоча такі системи дозволяють зменшити витрати на обробку в режимі онлайн, часто це супроводжується втратою точності в рекомендаціях.

Аналіз існуючих методів підкреслює актуальність підвищення ефективності та точності систем рекомендацій в умовах зростаючого обсягу інформації та складності задачі забезпечення користувачів персоналізованими рекомендаціями. З ростом обсягу даних та різноманітності інтересів користувачів традиційні методи рекомендаційних систем стикаються з численними обмеженнями, такими як розрідженість даних, недостатній обсяг оцінок користувачів та обмеженість інформації про вміст об'єктів тощо. Ці обмеження можуть призводити до низької точності та обмеженої масштабованості систем рекомендацій, ускладнюючи надання користувачам рекомендацій, які відповідають їхнім індивідуальним потребам та інтересам.

3. Мета і задачі дослідження

Метою даного дослідження є розробка покращеної моделі користувача для рекомендаційних систем для поліпшення точності та ефективності у ситуаціях, коли

інформація стає дедалі об'ємнішою, і завдання забезпечення персоналізованих рекомендацій стають складнішими.

Для досягнення цієї мети пропонується вирішити такі задачі:

1. Розробити компактну гібридну модель користувача, яка буде долати проблеми існуючих методів, такі як розрідженість даних, недостатній обсяг оцінок користувачів та обмеженість інформації про вміст об'єктів. Для цього ввести гібридну ознаку, яка буде точно показувати інтерес користувача до різних жанрів і поєднає оцінки користувачів та інформацію про фільми, а саме про їх жанр;

2. Провести обчислення, спрямовані на демонстрацію та підтвердження ефективності розробленої моделі користувача та її ключової характеристики, а саме гібридної ознаки.

4. Компактна гібридна модель користувача

Спільна фільтрація на основі пам'яті є точнішою, але її масштабованість, порівняно з рекомендаційною системою на основі моделі, гірша. Крім того, фактичні уподобання користувача не завжди можуть бути охоплені лише оцінками, і тому потрібні деякі описи вмісту об'єктів. Цього можна досягти, якщо ми побудуємо гібридну модель користувача. Ідея об'єднання або інтеграції багатьох джерел інформації може значною мірою полегшити проблеми розрідженості та масштабованості. Під час пошуку інформації ознаки з відповідних джерел часто додаються до базового представлення. Це зазвичай називають розширенням запиту. Цей спосіб не допоможе в нашому випадку, тому що якщо ми подолаємо проблему розрідженості, ефект проблеми масштабованості збільшиться. Однак ми хочемо одночасно зменшити вплив цих проблем. Для цього запропоновано набір функцій для обчислення гібридних ознак, що поєднують деякі властивості користувачів і фільмів.

Для наочного прикладу моделі, що пропонується, було взято рекомендаційну систему, яка пропонує користувачам кінофільми. Система рекомендацій повинна отримувати ідентифікатор користувача як вхідні дані і повертати список фільмів та інформацію про те, наскільки цікавими вони будуть для користувача. Ця інформація, фактично, є прогнозом системи на основі її алгоритму і може бути інтерпретована, наприклад, як повідомлення користувачу про топ-10 фільмів для нього. Система рекомендацій також повинна відповідно обчислювати рейтинги для всіх відомих фільмів, які користувач не дивився, і для яких отримано унікальний номер як вхідні дані. Для створення такої системи рекомендацій потрібна база даних, яка повинна складатися з двох частин. Перша частина містить список усіх користувачів системи та інформацію про їх взаємодію з фільмами (рейтинги та перегляди). Друга частина - це детальна та обширна база даних про самі фільми. Отже, перед створенням системи рекомендацій повинні бути доступні такі вхідні дані:

– вся доступна інформація про фільми: назва, короткий сюжет, жанр, ключові слова, акторський склад, режисер та ін.;

– інформація про користувачів: дані про користувача, для кожного користувача – конкретний набір деякої кількості рейтингів, кожен з яких пов'язаний з певним фільмом.

Формально маємо множину користувачів $U = \{u_1 \dots, u_M\}$, де M – кількість користувачів, які мають явні або неявні оцінки множини об'єктів (фільмів) $S = \{s_1 \dots, s_K\}$, де K – кількість об'єктів.

Формально маємо множину користувачів $U = \{u_1 \dots, u_M\}$, де M – кількість користувачів, які мають явні або неявні оцінки множини об'єктів (фільмів) $S = \{s_1 \dots, s_K\}$, де K – кількість об'єктів. Множини S та U є великими, а у деяких випадках можуть бути величезними. Кожен

користувач $u_i, i = 1, \dots, M$, оцінив підмножину об'єктів S_i . Оцінку користувача u_c для об'єкту $s_k, k = 1, \dots, K$, позначено як $r_{c,k}$. Усі доступні оцінки збираються в $(M \times K)$ матрицю "користувач-об'єкт", позначену як R . Архітектура різних систем рекомендацій може бути централізованою або розподіленою. У цій роботі представлена централізована архітектура, де система рекомендацій розташована в одному конкретному місці. Під час розробки системи рекомендацій можна виділити такі п'ять етапів [1]:

- збір даних;
- формування моделі користувача;
- обчислення схожості;
- вибір сусідів;
- прогнози та рекомендації.

Взаємодію цих етапів можна описати схемою, наведеною на рис. 1.



Рис. 1. Схема етапів процесу рекомендації

На основі рейтингів користувача для набору високо оцінених фільмів та описів жанрів, які відповідають цьому набору фільмів, за допомогою формул (1)-(8) виведено «Міру цікавості жанру» (Genre Interestingness Indicator - GII) (9) - гібридну ознаку, яку детально описано у [9] і яка поєднує оцінки користувачів та жанри фільмів. Нижче описано процес виведення «Міри цікавості жанру».

Для певного користувача u_i і певного жанру, G_i набір фільмів з високою оцінкою H_i є:

$$H_i = \{r_{i,k} : s_k \in S_i, r_{i,k} \geq 3\}, \quad (1)$$

Тут S_i — набір фільмів, оцінених u_i .

Загальна оцінка $TR(u_i)$ користувача u_i :

$$TR(u_i) = \sum_{s_k \in S_i} r_{i,k}. \quad (2)$$

Жанровий рейтинг $GR_{u_i}(G_j)$ (відповідно частота жанру, $GF_{u_i}(G_j)$) для високо оцінених фільмів жанру G_j , що відповідають користувачу u_i :

$$GR_{u_i}(G_j) = \sum_{s_k \in G_j \subset H_i} r_{i,k}, \quad (3)$$

$$GF_{u_i}(G_j) = \sum_{s_k \in G_j \subset H_i} \delta_z(r_{i,k}), \quad z \in \{3,4,5\}, \quad (4)$$

де

$$\delta_z(r_{i,k}) = \begin{cases} 1 & z = r_{i,k} \\ 0 & z \neq r_{i,k} \end{cases}. \quad (5)$$

Слід зазначити, що ми розглядаємо лише фільми, які мають рейтинг $r_{i,k} \geq 3$, тобто фільми, оцінені як "добре" – 3, "дуже добре" – 4 або "відмінно" – 5. Нарешті, відносний жанровий рейтинг (RGR) (6) є відношенням жанрового рейтингу (3) користувача u_i до загальної оцінки користувача (2). Відповідно, відносна частота жанру (RGF) (7) є відношенням частоти жанру (4) користувача u_i до загальної частоти оцінок користувача.

$$RGR_{u_i}(G_j) = \frac{GR_{u_i}(G_j)}{TR(u_i)}, \quad (6)$$

$$RGF_{u_i}(G_j) = \frac{GF_{u_i}(G_j)}{TF(u_i)}, \quad (7)$$

де загальна частота $TF(u_i) = |S_i|$ є потужністю S_i . Схема для запропонованої моделі користувача наведена на рис. 2. Перш за все, для створення гібридних ознак використовуються жанри та явні оцінки.

Як простий приклад розглянемо табл. 5 лише для трьох користувачів, які оцінили фільми, що належать до чотирьох жанрів. У стовпцях $G_j, j = 1,2,3,4$ значення "1" вказує на належність даного фільму до G_j , а "0" –

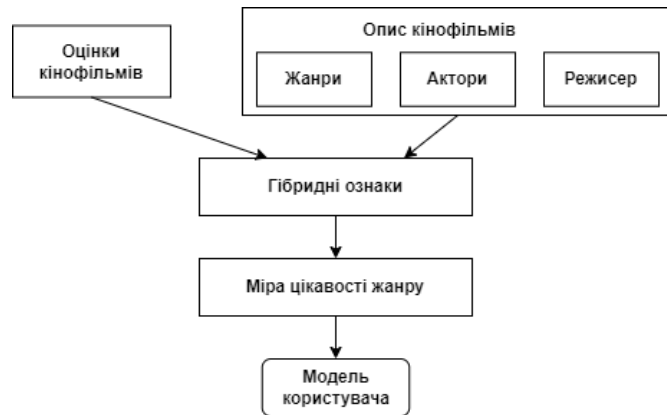


Рис. 2. Схема компактної гібридної моделі користувача

в іншому випадку. Також ненульове значення в стовпцях оцінок користувачів вказує на те, що фільм був оцінений, а нуль вказує, що фільм ще не оцінений.

У табл. 6 наведено рейтинги, пов'язані з різними жанрами. Слід зазначити, що оцінка конкретного фільму пов'язується з усіма жанрами, до яких належить фільм. Наприклад, оцінки User-3 для G_4 — це 4, 5 і 3, де вони відповідають фільму-5, фільму-8 та фільму-11 (табл. 5) відповідно. Як обговорювалося раніше, гібридні ознаки використовуються як основа для вимірювання цікавості жанру, тому користувача більше цікавить G_i , якщо він має вищі оцінки, тобто "добре", "дуже добре" або "відмінно". Кількість гібридних ознак залежить від кількості жанрів. На основі цього можна зробити висновок, що User-1, User-2 і User-3 більше цікавляться G_3 , G_1 і G_4 відповідно (табл. 6). Хоча деякі жанри мають низькі рейтинги

Таблиця 5

Оцінки користувачів

| Фільм | Жанри фільмів | | | | Рейтинги користувачів | | |
|-------|---------------|-------|-------|-------|-----------------------|--------|--------|
| | G_1 | G_2 | G_3 | G_4 | User-1 | User-2 | User-3 |
| 1 | 1 | 0 | 1 | 0 | 1 | 5 | 0 |
| 2 | 1 | 0 | 1 | 1 | 3 | 0 | 0 |
| 3 | 1 | 1 | 0 | 0 | 1 | 5 | 3 |
| 4 | 0 | 1 | 1 | 0 | 5 | 1 | 0 |
| 5 | 0 | 1 | 1 | 1 | 4 | 0 | 4 |
| 6 | 0 | 1 | 0 | 0 | 0 | 2 | 0 |
| 7 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 8 | 0 | 0 | 1 | 1 | 0 | 0 | 5 |
| 9 | 0 | 1 | 1 | 0 | 0 | 2 | 0 |
| 10 | 1 | 1 | 1 | 0 | 0 | 4 | 1 |
| 11 | 1 | 1 | 0 | 1 | 0 | 3 | 3 |
| 12 | 1 | 0 | 0 | 0 | 0 | 0 | 3 |

Таблиця 6

Оцінки користувачів по жанрах

| User | TF | TR | G_1 | G_2 | G_3 | G_4 |
|------|----|----|---------|-------------|---------|-------|
| 1 | 5 | 14 | 1,3,1 | 1,5,4 | 1,3,5,4 | 3,4 |
| 2 | 7 | 22 | 5,5,4,3 | 5,1,2,2,4,3 | 5,1,2,4 | 3 |
| 3 | 6 | 19 | 3,1,3,3 | 3,4,1,3 | 4,5,1 | 4,5,3 |

($r_{i,k} < 3$), їх можна відфільтрувати на етапі спільної фільтрації. Необхідно знайти компактну модель користувача, щоб виявити найближчих сусідів для цього користувача. Після цього отриманий набір сусідів використовується для спільної фільтрації, щоб рекомендувати певні фільми, які можуть сподобатися користувачеві. Тому не можна рекомендувати активному користувачеві будь-який фільм із прогнозованим рейтингом, меншим за "добре", тобто 3.

Для того, щоб дослідити можливість вибору формули для GII , ми розглянули формули (6) і (7). Формула для RGR (6) добре працює для User-1 та User-2, але для User-3 вона дає G_1 і G_3 однакові значення (табл. 7), тоді як оцінки користувача для фільмів цих двох жанрів досить різні. G_1 має три оцінки "добре" – 3, а G_3 має оцінки "дуже добре" – 4 і "відмінно" – 5. Це тому, що кількість фільмів для кожного жанру ігнорується в цій формулі. Формула для RGF (7) добре працює для User-2, але для User-3 вона дає G_1 , G_2 і G_4 однакові значення. Крім того, значення G_3 нижче, ніж у G_1 і G_2 , тоді як оцінки для G_3 – "дуже добре" та "відмінно". Це пов'язано з тим, що враховується лише кількість фільмів із високим рейтингом, і всі високі оцінки вносяться в цю формулу однаково. Очевидно, що RGR і RGF є формулами GII з деякими недоліками. Насправді, користувач представляє своє вподобання до фільму за допомогою надання оцінки цьому фільму, наприклад "добре", "дуже добре" або "відмінно". Цю детальність уподобань неможливо відобразити, якщо ми візьмемо просте усереднення частот, оскільки воно надає всім високим рейтингам однакову вагу. Тому введено нову версію формули RGF , яка намагається відобразити точне уподобання до фільмів з високими

рейтингами (схожий підхід наведено у [10]). Тобто модифікована відносна частота (*MRGF*) жанру G_j для користувача u_i визначається як:

$$MRGF_{u_i}(G_j) = \frac{\sum_{s_k \in G_j \subset H_i} \delta_3(r_{i,k}) + 2 \times \delta_4(r_{i,k}) + 3 \times \delta_5(r_{i,k})}{3 \times TF(u_i)} \quad (8)$$

Результати обчислень, наведені у табл. 7, показують, що, як і очікувалося, *MRGF* долає недоліки *RGF*.

Таблиця 7

Результати обчислень

| <i>GII</i> формула | User | G_1 | G_2 | G_3 | G_4 |
|--------------------|------|-------|-------|-------|-------|
| <i>RGR</i> | 1 | 0.214 | 0.643 | 0.857 | 0.500 |
| | 2 | 0.773 | 0.545 | 0.409 | 0.136 |
| | 3 | 0.474 | 0.526 | 0.474 | 0.632 |
| <i>RGF</i> | 1 | 0.200 | 0.400 | 0.600 | 0.400 |
| | 2 | 0.571 | 0.429 | 0.286 | 0.143 |
| | 3 | 0.500 | 0.500 | 0.333 | 0.500 |
| <i>MRGF</i> | 1 | 0.067 | 0.333 | 0.400 | 0.200 |
| | 2 | 0.429 | 0.286 | 0.238 | 0.048 |
| | 3 | 0.167 | 0.222 | 0.278 | 0.333 |

Для того, щоб розробити точніший *GII*, необхідно також враховувати відносний жанровий рейтинг. Поєднання формул *RGR* і *MRGF* допоможе взаємно усунути недоліки, які притаманні окремим формулам. Для випадку жанру G_j для користувача u_i міра цікавості жанру (*GII*) визначається як:

$$GII_{u_i}(G_j) = \frac{2 \times nf \times RGR_{u_i}(G_j) \times MRGF_{u_i}(G_j)}{RGR_{u_i}(G_j) + MRGF_{u_i}(G_j)}, \quad (9)$$

де *nf* - коефіцієнт нормалізації для даної системи.

Формула (9) дає гармонічне середнє значення $RGR_{u_i}(G_j)$ і $MRGF_{u_i}(G_j)$, помножене на *nf*. Діапазон $GII_{u_i}(G_j)$ становить [0, MAX] Це узгоджується зі структурою рейтингу системи, тобто 1 – "дуже погано", ..., MAX – "відмінно". Тут MAX - найвищий можливий рейтинг для даної системи. Коефіцієнт нормалізації *nf* приймає значення MAX або глобального середнього рейтингу користувача ($TR(u_i)/TF(u_i)$). Результати обчислень на основі формули (9) з *nf* = MAX показані в табл. 8. Ці результати відображають саме ступінь цікавості для кожного

Таблиця 8

Результати обчислень *GII*

| Користувач | $GII_{u_i}(G_j)$ | | | |
|------------|------------------|-------|-------|-------|
| | G_1 | G_2 | G_3 | G_4 |
| 1 | 0.510 | 2.194 | 2.727 | 1.429 |
| 2 | 2.759 | 1.876 | 1.505 | 0.355 |
| 3 | 1.235 | 1.561 | 1.752 | 2.181 |

жанру: User-1 більше зацікавлений у G_3 , User-2 – G_1 , User-3 – в G_4 . За допомогою формули (9) отримуються три основні переваги. По-перше, уподобання користувачів у жанрах легко обчислити. По-друге, отримується компактна модель користувача. По-третє, GII гарантує транзитивність сусіда, тому користувачі, які мають близький GII , будуть корелювати між собою (мати схожі смаки).

Дана компактна модель користувача вводить гібридизацію на двох різних рівнях, а саме, на рівні пам'яті і на рівні моделі. Гібридні функції використовують як оцінки користувачів для фільмів з високим рейтингом, так і деякі описи вмісту фільмів (жанрів). На рівні моделі компактна модель користувача використовується для пошуку набору односторонніх, серед яких здійснюється пошук на основі пам'яті. Цей набір набагато менший за розміром, ніж оригінальний набір, що робить систему масштабованою. Очевидно, що $GII_{u_i}(G_j)$ представляє інтерес u_i користувача до жанру G_j на основі оцінок фільмів у жанрі G_j , що були надані іншими користувачами.

5. Висновки та перспективи подальших досліджень

Розглянуто та проаналізовано різні методи фільтрації інформації для генерації персоналізованих рекомендацій. Розроблено компактну гібридну модель користувача для методу спільної фільтрації. Така модель пов'язує оцінки користувачів з деяким описом вмісту об'єктів. Для цього введено поняття міри цікавості жанру, що являє собою гібридну ознаку, яка поєднує оцінки користувачів і жанри фільмів, та представляє уподобання користувача на рівні моделі. Такий підхід дозволяє зберігати точність спільної фільтрації на основі пам'яті та водночас забезпечує масштабованість завдяки моделі користувача. Це робить процес рекомендації компактнішим та швидшим у порівнянні із стандартними методами. Важливо відзначити, що ця модель допомагає вирішити проблеми розрідженості даних та обмеженої інформації про об'єкт, які пов'язані з розглянутими у статті підходами, і водночас зберігає транзитивність відносин між сусідніми користувачами. Для можливого поліпшення результатів у майбутніх дослідженнях можна розглянути впровадження модифікованого методу фільтрації з використанням алгоритмів нечіткої логіки, що допоможе уникнути недоліків при побудові моделі з ознаками, яким притаманна нечіткість та невизначеність.

Перелік посилань:

1. Charu C. Aggarwal. Recommender Systems: The Textbook, 2016, http://pzs.dstu.dp.ua/DataMining/recom/bibl/1aggarwal_c_c_recommender_systems_the_textbook.pdf (Last accessed: 10.10.2023).
2. Francesco Ricci, Lior Rokach, Bracha Shapira. Recommender Systems Handbook, 2015, https://www.cse.iitk.ac.in/users/nsrivast/HCC/Recommender_systems_handbook.pdf (Last accessed: 10.10.2023).
3. M.Sridevi, Dr .R.Rajeswara Rao, DECORS: A Simple and Efficient Demographic Collaborative Recommender System for Movie Recommendation, 2017, ISSN 0973-6107 Volume 10, Number 7, pp. 1969-1979. https://www.ripublication.com/acst17/acstv10n7_01.pdf (Last accessed: 20.10.2023).
4. Ruogu Kang, Stephanie BrownSara, KieslerSara Kiesler, Why do people seek anonymity on the Internet? Informing policy and design, 2013 https://www.researchgate.net/publication/262273589_Why_do_people_seek_anonymity_on_the_Internet_Informing_policy_and_design (Last accessed: 20.10.2023).
5. Abdellah El Fazziki, Ouafae El Aissaoui, Yasser EL Madani El Alami, Youssouf El Alloui, Mohammed Benbrahim, A new collaborative approach to solve the gray-sheep users problem in recommender systems, 2019, https://www.researchgate.net/publication/338361794_A_new_collaborative_approach_to_solve_the_gray-sheep_users_problem_in_recommender_systems (Last accessed: 20.10.2023).

6. Juuso Kaitila, A content-based music recommender system, 2017, <https://www.cs.rit.edu/usr/local/pub/GraduateProjects/2161/kxd8041/Report.pdf> (Last accessed: 20.10.2023).
7. Guan, Xin, On reducing the data sparsity in collaborative filtering recommender systems., 2017, https://wrap.warwick.ac.uk/97978/1/WRAP_Theses_Guan_2017.pdf (Last accessed: 20.10.2023).
8. Wanvimol Nadee, Modelling user profiles for recommender systems, 2016, https://eprints.qut.edu.au/93723/1/Wanvimol_Nadee_Thesis.pdf (Last accessed: 20.10.2023).
9. Andreu Vall, Matthias Dorfer, Hamid Eghbal-zadeh, Markus Schedl, Keki Burjorjee & Gerhard Widmer. Feature-combination hybrid recommender systems for automated music playlist continuation, 2019, <https://link.springer.com/article/10.1007/s11257-018-9215-8> (Last accessed: 10.10.2023).
10. Sebastien Fremal, Fabian Lecron. Weighting Strategies for a Recommender System Using Item Clustering Based on Genres, 2017, p. 6-11, <https://www.sciencedirect.com/science/article/abs/pii/S0957417417300404> (Last accessed: 10.10.2023).

Надійшла до редколегії 13.10.2023 р

Ситнікова Поліна Едуардівна, канд. техн. наук, доцент, доцент кафедри системотехніки ХНУРЕ, м. Харків, Україна, e-mail: polina.sytnikova@nure.ua, ORCID: <https://orcid.org/0000-0002-6688-4641>.

Гребенюк Микита Олександрович, аспірант кафедри системотехніки ХНУРЕ, м Харків, Україна, e-mail: mykyta.hrebeniuk@nure.ua. ORCID: <https://orcid.org/0009-0008-0989-7957>.

УДК 004.627

DOI: 10.30837/0135-1710.2023.179.042

І.Г. ПЕРОВА, Н.С. МІРОШНИЧЕНКО

ОГЛЯД ІСНУЮЧИХ МЕТОДІВ ЗМЕНШЕННЯ РОЗМІРНОСТІ ТА КЛАСИФІКАЦІЇ ВЕЛИКИХ ВИБІРОК ДАНИХ

Аналіз великих вибірок даних, який проводиться з метою виявлення прихованих закономірностей і тенденцій, за останні роки стає все важливішим і кориснішим. Такі великі вибірки на поточний час характеризуються загальнодоступністю, складністю структури і великими розмірами.

Для вирішення проблеми великої розмірності даних пропонується ознайомлення з існуючими методами зменшення розмірності великих вибірок даних та порівняння ефективності цих методів на репозиторних вибірках. Розглядаються такі методи, як аналіз головних компонент (Principal Component Analysis), лінійний дискримінантний аналіз (Linear Discriminant Analysis), аналіз головних компонент ядра (Kernel Principal Component Analysis), багатовимірне масштабування (MDS), метод t-розподільного стохастичного вбудовування сусідів (t-SNE) та аналіз незалежних компонент (Independent Component Analysis). Як приклади великих вибірок даних використовуються набір даних ініціативи з нейровізуалізації хвороби Альцгеймера (ADNI) та набір даних про щитоподібну залозу, який є одним з декількох баз даних про щитоподібну залозу, доступних в репозиторії UCI.

1. Вступ

Для роботи з великими вибірками даних попередньо необхідно зменшити кількість параметрів вибірки. Цей процес називається зменшенням розмірності вибірки.

Зменшення розмірності як етап попередньої обробки машинного навчання є ефективним для видалення нерелевантних і надлишкових даних, підвищення точності навчання та покращення зрозумілості результату за допомогою візуалізації розмірності [1]. Дуже важливо зменшити розмірність набору даних без втрати будь-якої інформації, що міститься в них.

2. Мета і задачі дослідження

Метою даної роботи є порівняльний аналіз методів та підходів до зменшення розмірності великих вибірок даних у сфері машинного навчання.

Задачами дослідження є:

- ознайомлення з існуючими методами зменшення розмірності великих вибірок даних;
- опис головної ідеї кожного із методів;
- дослідження особливостей застосування кожного з методів на репозиторних вибірках даних;
- порівняння отриманих результатів.

3. Матеріали і методи дослідження

На даний момент найпопулярнішим методом для задач високої розмірності вважають метод аналізу головних компонент (Principal Component Analysis (PCA)) – метод вилучення ознак, який використовується для аналізу статистичних даних шляхом перетворення початкового набору даних у різноманітний набір лінійних комбінацій. Такі комбінації відомі як головні компоненти, що мають певні властивості щодо дисперсій. Виділення лінійних комбінацій робить розмірність системи щільнішою і в той же час зберігає інформацію про змінні зв'язки. Обчислення виконуються на наборі даних шляхом розрахунку власних значень коваріаційної матриці, розташованих у систематичному порядку за спаданням. Ця методика забезпечує максимальну реалістичність довільних рішень у просторі великої розмірності [2].

Як приклад розглянемо набір даних, який складається з переліку пацієнтів та ознак, що їх характеризують. Припустимо, що цей набір даних являє собою матрицю $X = \{x_{mn}\}$ розмірністю $K \times k$, де K – кількість пацієнтів, а k – кількість ознак, які характеризують кожного пацієнта. Тоді кожен пацієнт описується $(k \times 1)$ -вектором ознак

$$x(a) = (x_1(a), x_2(a), \dots, x_j(a), \dots, x_k(a))^T, \quad (1)$$

де $a=1, \dots, K$.

Перша головна компонента PC_1 є лінійною комбінацією вихідних змінних X_1, X_2, \dots, X_k

$$PC_1 = c_{11}X_1 + c_{12}X_2 + c_{13}X_3 \dots \dots c_{1p}X_k = \sum_{j=1}^p c_{ij}X_j \quad (2)$$

Умова

$$c_{11}^2 + c_{12}^2 + c_{13}^2 \dots \dots c_{1p}^2 = 1, \quad (3)$$

вказує на те, що коефіцієнти c_{1p} мають бути нормованими, тобто квадрати суми їх коефіцієнтів дорівнюють 1. Ця умова гарантує, що власне значення PC_1 є найбільшим, враховуючи обмеження (3) на константу c . Обмеження має бути накладено для того, щоб уникнути збільшення власного значення PC_1 простим збільшенням одного або декількох значень вихідних змінних.

Проте, незважаючи на те, що PCA є дуже ефективним, він є і повністю неконтрольованою технікою зменшення розмірності. Це означає, що цей метод не використовує мітки класів під час навчання низьковимірною представлення, що призводить до субоптимальної продуктивності під час вирішення надалі задачі класифікації.

Ще одним з найважливіших фундаментальних методів є метод лінійного дискримінантного аналізу Фішера (Linear Discriminant Analysis (LDA)). Цей метод має дуже важливі властивості для класифікації даних. По-перше, він базується на простих геометричних принципах: необхідно максимізувати відстань між середніми значеннями класів (центрами) та мінімізувати внутрішньокласову дисперсію. Це означає, що LDA намагається знайти лінійну комбінацію ознак, яка найкращим чином розділяє класи, збільшуючи одночасно роздільну здатність. По-друге, метод LDA дозволяє зменшити розмірність даних, обираючи нові змінні (дискримінантні ознаки), які відображають основні відмінності між класами. Це корисно для подальшого аналізу та класифікації, оскільки він дозволяє працювати з меншими кількостями ознак, що може зменшити обчислювальні витрати і запобігти перенаванчання [3].

LDA зосереджується головним чином на проєкції функцій з простору вищих вимірів у простір нижчих вимірів. Для цього необхідно обчислити відстань між середніми значеннями різних класів, яка називається міжкласовою дисперсією

$$S_B = \sum_{k=1}^g N_k (m_k - m) (m_k - m)^T, \quad (4)$$

де g – кількість класів; m – загальне середнє; m_k – середнє для відповідного класу; N_k – обсяг вибірки для k -го класу.

Наступним кроком розраховується внутрішньокласова дисперсія (S_w)

$$S_w = \sum_{i=1}^c S_i, \quad (5)$$

де c – кількість класів; S_i – матриця розсіювання окремого i -го класу, яка обчислюється за формулою

$$S_i = \sum_{x \in D_i}^{n_i} (x - m_i) (x - m_i)^T, \quad (6)$$

де D_i – множина прикладів з i -го класу; n_i – обсяг вибірки для i -го класу. m_i – середній вектор для i -го класу. Формула (6) показує, як внутрішньокласова дисперсія (S_w) обчислюється для кожного класу окремо. Матриця S_i – відображає розсіювання даних всередині конкретного класу i .

Останнім кроком необхідно побудувати простір нижчих розмірів, який максимізує дисперсію між класами та мінімізує дисперсію всередині класу.

$$P_{LDA} = \arg \max \left| \frac{P^T S_B P}{P^T S_w P} \right| \quad (7)$$

де P – критерій Фішера, або так звана проєкція простору нижчої вимірності.

Проте даний метод має недоліки, основним з яких є неможливість надійної оцінки параметрів моделі без достатньої кількості зразків для кожного класу. Якщо деякі класи мають обмежену кількість прикладів, то LDA може давати неточні результати. Застосування LDA при малій кількості зразків в одному з класів схоже на спробу підігнати пряму до точки. Отже, без додаткових обмежень ці алгоритми будуть перенавчатися, тобто обиратимуть

класифікатор на основі шуму в даних, а не відкидатимуть шум на користь потрібного сигналу. Це означає, що даний метод можливо використовувати тільки для виконання контрольованого зменшення розмірності шляхом проєкції вхідних даних у лінійний підпростір, що складається з напрямків, які максимізують поділ між класами. Розмірність результату зазвичай повинна бути меншою за кількість класів, тому це загалом досить сильне зменшення розмірності, яке має сенс лише у випадку багатокласовості [4].

Ще одним методом зменшення розмірності є аналіз головних компонент ядра (Kernel Principal Component Analysis (KPCA)). Стандартний PCA дозволяє зменшувати лише лінійну розмірність. Однак, якщо дані мають складнішу структуру, яка не може бути добре представлена в лінійному підпросторі, стандартний PCA не буде дуже ефективним. На щастя, ядро PCA дозволяє узагальнити стандартний PCA до нелінійного зменшення розмірності, крім того, даний метод дозволяє спростувати обчислення. KPCA може обробляти багатовимірні набори даних із багатьма функціями, зменшуючи розмірність даних, зберігаючи найважливішу інформацію [5].

Порівняльний аналіз корисності функцій лінійного та нелінійного ядер PCA для розпізнавання за допомогою лінійного класифікатора виявив дві переваги нелінійного ядра PCA: по-перше, нелінійні головні компоненти мають кращі показники розпізнавання ніж відповідні показники лінійних головних компонент. По-друге, продуктивність для нелінійних компонентів можна додатково покращити, використовуючи більше компонентів, ніж це можливо в лінійному випадку [6].

Проте KPCA має і свої недоліки. Даний метод має деякі обмеження, наприклад, необхідність вибору відповідної функції ядра та її параметрів, що може бути складним і трудомістким. KPCA також може бути дорогим в обчислювальному плані для великих наборів даних, оскільки він вимагає обчислення матриці ядра для всіх пар точок даних.

Багатовимірне масштабування (Multidimensional Scaling (MDS)) – це ще одна методика зменшення та візуалізації, аналізу і звітності результатів. Даний метод на першому кроці працює з інформацією про певну форму несхожості між елементами набору об'єктів. Потім на основі цієї інформації створюється геометричне зображення, у якому відмінності представлені як відстані, які можуть бути використані для визначення ступеня схожості і відмінностей між парами об'єктів, кожна з яких представляє реальну точку даних. Об'єкти з невеликими відмінностями демонструють високий ступінь подібності, тоді як об'єкти з великими відмінностями вважаються несхожими [7].

Математичною основою MDS є функція напруження, яка вимірює різницю між відстанями у вихідному просторі та відстанями у просторі нижчої розмірності

$$stress = \sqrt{\frac{1}{2n^2 \sum_{i=1}^n \sum_{j=1}^n (d_{ij} - \widehat{d}_{ij})^2}}, \quad (8)$$

де d_{ij} – відстань між точками даних i та j у вихідному просторі; \widehat{d}_{ij} – відстань між точками даних i та j у просторі нижньої розмірності; n - кількість точок даних. Функція напруження є мірою відхилення відстаней у нижньовимірному просторі від відстаней у вихідному просторі і використовується для оцінки якості проєкції.

Проте MDS є описовим методом, в якому практично повністю відсутнє поняття статистичного висновку [8].

До групи методів нелінійного зменшення розмірності можна також віднести метод t-розподіленого стохастичного вбудовування сусідів (t-SNE). Даний метод добре підходить для вбудовування даних великої розмірності з метою подальшої їх візуалізації в дво- або тривимірний простір низької розмірності. t-SNE знаходить закономірності в даних на основі схожості точок даних, схожість точок обчислюється як умовна ймовірність того, що точка A вибере точку B як свого сусіда. Умовна ймовірність для високої розмірності визначає, наскільки близько x_j знаходиться від x_i з урахуванням гауссівського розподілу навколо x_i із заданою дисперсією σ_i^2 , і розраховується за формулою:

$$p_{i|j} = \frac{\exp(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2})}{\sum_{k \neq i} \exp(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2})}, \quad (9)$$

Дисперсія σ_i^2 є різною для кожної точки; вона обрана таким чином, що точки в щільних областях мають меншу дисперсію, ніж точки в розріджених областях.

Для низькорозмірних аналогів y_i та y_j високорозмірних точок даних x_i і x_j можна обчислити подібну умовну ймовірність за формулою:

$$q_{j|i} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq i} \exp(-\|y_i - y_k\|^2)}. \quad (10)$$

Вимірювання попарних відстаней у високо- та низьковимірному просторах з використанням різних розподілів ймовірностей дозволяє оптимальніше візуалізувати внутрішньокластерні деталі [9]. Метод є стохастичним, тому багаторазове його виконання з різними випадковими значеннями дасть різні результати. Допустимо запустити алгоритм кілька разів і вибрати вкладення з найменшою розбіжністю KL

$$KL = \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}}, \quad (11)$$

де p_{ij} та q_{ij} – парні ймовірності у високовимірному та низьковимірному просторі відповідно. Причиною точнішого визначення локальної структури є те, що функція витрат пропорційна парній ймовірності у високовимірному просторі. Відносно віддалені точки мають набагато менший вплив на функцію витрат.

Недоліком даного методу можна вважати великі кількість часу і обсяг простору, необхідні для обчислення. Це пов'язано з тим, що алгоритм обчислює попарно умовні ймовірності з метою мінімізувати суму різниць ймовірностей у вищих і нижчих вимірах, що потребує багато обчислень. t-SNE має квадратичну часову і просторову складність в залежності від кількості точок даних [10].

Наближенням до t-SNE є метод рівномірної багатовимірної апроксимації та проєкції (UMAP), який використовується для візуалізації. Метод базується на трьох припущеннях:

– дані рівномірно розподілені на Рімановому просторі;

- Ріманова метрика є локально сталою (або її можна наближено вважати такою);
- простір є локально зв'язним.

Виходячи з цих припущень, можна змодельовати множину з нечіткою топологічною структурою. Вкладення знаходять шляхом пошуку низьковимірної проекції даних, яка має найближчу еквівалентну нечітку топологічну структуру.

Цей метод дуже добре працює з різними даними, від розпізнавання зображень до кластеризації та класифікації даних. Результати UMAP дуже якісні, та є конкурентоспроможними відносно методу t-SNE, завдяки своїм перевагам у швидкості та стабільності [11].

Аналіз незалежних компонент (Independent Component Analysis (ICA)) – це статистичний метод для виявлення прихованих факторів, які лежать в основі наборів випадкових величин, вимірювань або сигналів. ICA визначає генеративну модель для спостережуваних багатовимірних даних, які зазвичай подаються у вигляді великої бази даних зразків. У моделі передбачається, що змінні даних є лінійними сумішами деяких невідомих латентних змінних, і система змішування також невідома [12]. Приховані змінні вважаються негауссівськими та взаємно незалежними, і їх називають незалежними компонентами спостережуваних даних. Ці незалежні компоненти, які також називаються джерелами або факторами, можна знайти за допомогою ICA

$$x_i(t) = \sum_{j=1}^m a_{ij}s_j(t), \quad \text{for all } i = 1, \dots, n, \quad (12)$$

де $x_i(t)$ – спостережувані змінні даних; i - індекс спостережуваної змінної даних, $i=1, \dots, n$; t – індекс часу або інший індекс різних спостережень, $t=1, \dots, T$. Припустимо, що $x_i(t)$ змодельовані як лінійні комбінації прихованих (латентних) змінних $s_j(t)$, $j=1, \dots, m$, з деякими невідомими коефіцієнтами a_{ij} .

ICA поверхнево пов'язаний з PCA. Однак ICA є набагато потужнішою технікою, здатною знайти основні фактори або джерела, коли класичні методи повністю не спрацьовують. Дані, що аналізуються за допомогою ICA, можуть походити з різних сфер застосування, включаючи цифрові зображення, бази даних документів, економічні показники та психометричні вимірювання. У багатьох випадках вимірювання подаються у вигляді набору паралельних сигналів або часових рядів; для характеристики цієї задачі використовується термін "сліпе розділення джерел". Типовими прикладами є суміші одночасних мовних сигналів, які були вловлені кількома мікрофонами, мозкові хвилі, записані кількома датчиками, радіосигнали, що заважають, які надходять на мобільний телефон, або паралельні часові ряди, отримані в результаті якогось промислового процесу.

4. Результати дослідження та їх обговорення

Розглянуті вище методи було досліджено на наборі даних ініціативи з нейровізуалізації хвороби Альцгеймера (ADNI) [13]. Основною метою ADNI було перевірити, чи можна комбінувати для вимірювання прогресування легкого ступеня когнітивних порушень (MCI) і ранньої хвороби Альцгеймера (AD) результати магнітно-резонансної томографії (МРТ), позитронно-емісійної томографії (ПЕТ) та інших біологічних маркерів, а також клінічних та нейропсихологічних оцінок.

Набір даних містить записи про 90 ознак для 500 пацієнтів, які були відібрані випадковим чином (по 100 пацієнтів у кожному класі) для оцінки моделей. П'ять отриманих класів:

когнітивно нормальні (CN), хвороба Альцгеймера (AD), ранні легкі когнітивні порушення (EMCI), пізні легкі когнітивні порушення (LMCI) та значні порушення пам'яті (SMC).

На рис. 1 показано тривимірне відображення результатів застосування розглянутих методів зменшення розмірності. AD, CN, EMCI, LMCI та SMC позначені червоним, блакитним, зеленим, пурпуровим та помаранчевим кольорами відповідно.

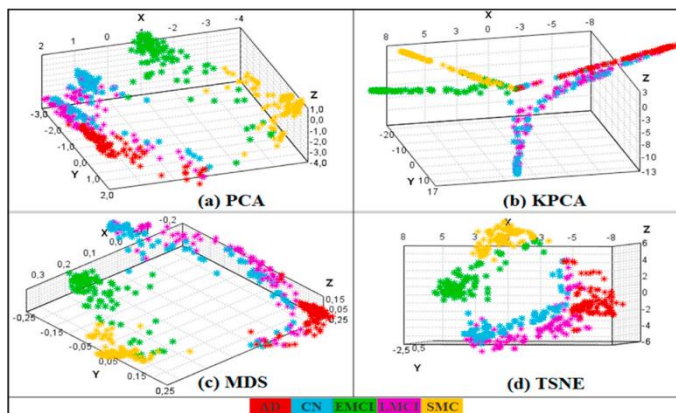


Рис. 1. 3D-візуалізація зменшень: а – PCA; б – KPCA; с – MDS; д – t-SNE.

Дослідження показали, що метод зменшення розмірності t-SNE показав відмінну класифікаційну оцінку даних. Даний метод є більш значущим і кращим для використання класифікаторами за інші методи.

Дослідження розглянутих методів було також проведено на наборі даних про захворювання щитовидної залози [14], який є одним з декількох баз даних про захворювання щитовидної залози, доступних в репозиторії UCI. У наборі даних зареєстровано 215 пацієнтів з трьох категорій: 1 – в нормі, 2 – страждають на гіпертиреоз, 3 – страждають на гіпотиреоз. Кожна вибірка містить п'ять характеристик: Т3 реверсивний, тироксин, трийодтиронін, тиреотропний гормон, ТТГ.

| | Normal | Suffer | Hyperthyroidism |
|-----------------|--------|--------|-----------------|
| Normal | 136 | 5 | 9 |
| Suffer | 9 | 18 | 3 |
| Hyperthyroidism | 7 | 4 | 24 |

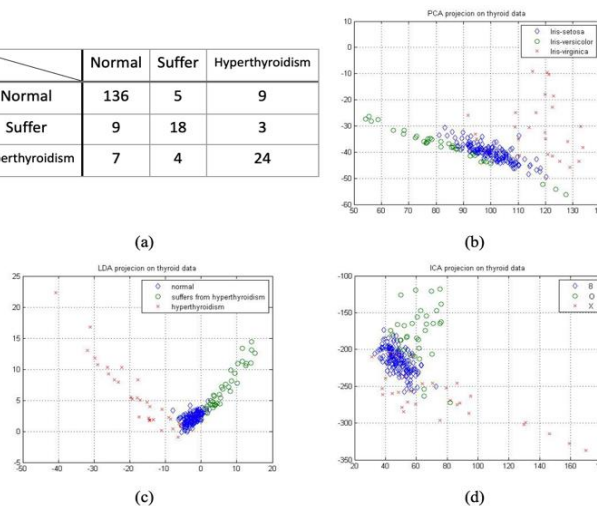


Рис. 2. Результати застосування методів зменшення розмірності для наборів даних щодо щитовидної залози: а – K-середніх; б – PCA; с –LDA; д – ICA

Рис. 2 ілюструє результати застосування методів PCA, LDA та ICA, отриманих для наборів даних про захворювання щитовидної залози.

Результати дослідження показують, що LDA має кращу продуктивність візуалізації, ніж PCA та ICA, оскільки використовує інформацію мітки. В алгоритмі ICA є випадковий фактор,

це означає, що якщо ми виберемо різні вектори для обчислення незалежних компонентів, продуктивність ІСА може змінюватись.

5. Висновки

Ознайомившись із головними ідеями розглянутих у дослідженні методів, автори дійшли висновку, що кожен з цих методів має свої унікальні властивості і використовується в залежності від конкретних завдань та властивостей даних. РСА спрямований на зменшення розмірності шляхом проєкції даних на нові ортогональні власні вектори (головні компоненти), які відображають напрямки найбільшої дисперсії в даних. LDA є методом зменшення розмірності, який спрямований на максимізацію віддільності класів даних шляхом проєкції даних на нові відомі вектори (власні вектори Фішера). ІСА намагається розкласти спостереження на лінійні комбінації незалежних компонентів, які максимально незалежні один від одного. КРСА є розширенням РСА, яке використовує ядерні функції для перетворення даних в вищу вимірність перед застосуванням РСА. Це дозволяє виявити нелінійні залежності в даних. t-SNE створений для візуалізації вимірюваних даних шляхом зменшення вимірності та збереження глобальних та локальних структур даних. Він базується на ймовірнісних розподілах схожості між об'єктами.

Проте, попри свою унікальність, ці методи мають спільні характеристики. Вони допомагають перетворити високорозмірні дані у низькорозмірний простір, зберігаючи при цьому важливу інформацію або структуру даних. Ці методи також є важливими інструментами в машинному навчанні та аналізі даних, оскільки вони допомагають зрозуміти та обробляти великі обсяги даних, зменшуючи їх розмірність або покращуючи їхню структуру.

Результати аналізу розглянутих методів дозволяють зробити висновок про відсутність на даний час досконалішого методу зменшення розмірності, який зміг би класифікувати багатокласові дані з мільйонами ознак, отримуючи при цьому сильні теоретичні гарантії, сприятливі та інтерпретовані емпіричні результати, а також гнучку, надійну та масштабовану реалізацію.

Перелік посилань:

- [1] Ayesha, S., Hanif, M. K., & Talib, R. (2020, July). Overview and comparative study of dimensionality reduction techniques for high dimensional data. *Information Fusion*, 59, 44–58. <https://doi.org/10.1016/j.inffus.2020.01.005>.
- [2] B. COY, “DIMENSION REDUCTION FOR ANALYSIS OF UNSTABLE PERIODIC ORBITS USING LOCALLY LINEAR EMBEDDING,” *International Journal of Bifurcation and Chaos*, vol. 22, no. 01, p. 1230001, Jan. 2019, doi: 10.1142/s0218127412300017.
- [3] Badaoui, F., Amar, A., Ait Hassou, L., Zoglat, A., & Okou, C. G. (2017, October 10). Dimensionality reduction and class prediction algorithm with application to microarray Big Data. *Journal of Big Data*, 4(1). <https://doi.org/10.1186/s40537-017-0093-4>.
- [4] S. Vannatta, “The Return of the Repressed (and Oppressed): A Freudo-Marxian Analysis of Jordan Peele’s Us,” *Popular Culture Review*, vol. 31, no. 2, 2020, doi: 10.18278/pcr.31.2.10.
- [5] Essa, A. M., & Ghalib Alrawi, A. (2019, September 1). Comparison Between The Method of Principal Component Analysis And Principal Component Analysis Kernel For Imaging Dimensionality Reduction. *IRAQI JOURNAL OF STATISTICAL SCIENCES*, 16(29), 11–24. <https://doi.org/10.33899/ijqjoss.2019.164189>.
- [6] Jiang, J. L., Li, S. Y., Liao, M. L., & Jiang, Y. (2019). Application in Disease Classification based on KPCA-IBA-LSSVM. *Procedia Computer Science*, 154, 109–116. <https://doi.org/10.1016/j.procs.2019.06.017>.
- [7] Dzemyda, G., Sabaliauskas, M., & Medvedev, V. (2022). Geometric MDS Performance for Large Data Dimensionality Reduction and Visualization. *Informatica*, 299–320. <https://doi.org/10.15388/22-infor491>.
- [8] T. Li, Q. Yin, R. Song, M. Gao, and Y. Chen, “Multidimensional scaling method for prediction of lysine glycation sites,” *Computing*, vol. 101, no. 6, pp. 705–724, Mar. 2019, doi: 10.1007/s00607-019-00710-x.

[9] Spiwok, V., & Kříž, P. (2020, June 30). Time-Lagged t-Distributed Stochastic Neighbor Embedding (t-SNE) of Molecular Simulation Trajectories. *Frontiers in Molecular Biosciences*, 7. <https://doi.org/10.3389/fmolb.2020.00132>.

[10] Ayyappa, T and S. Kurse, "Fault Detection of Bearing using XGBoost Algorithm and Data Visualization using t-distributed stochastic neighbor embedding (t-SNE) Method," *SSRN Electronic Journal*, 2021, Published, doi: 10.2139/ssrn.3834976.

[11] J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning," *Genetic Programming and Evolvable Machines*, vol. 19, no. 1–2, pp. 305–307, Oct. 2018, doi: 10.1007/s10710-017-9314-z.

[12] Hong, S. E. (2019, December 31). Exploring Independent Component Analysis Based on Ball Covariance. *The Korean Data Analysis Society*, 21(6), 2721–2735. <https://doi.org/10.37727/jkdas.2019.21.6.2721>.

[13] ADNI | Study Documents. (n.d.). <https://adni.loni.usc.edu/methods/documents/>;

[14] UCI Machine Learning Repository. (n.d.). <https://archive.ics.uci.edu/dataset/102/thyroid+disease>.

Надійшла до редколегії 02.08.2023

Перова Ірина Геннадіївна, доктор технічних наук, професор, професор кафедри системотехніки ХНУРЕ, м. Харків, Україна, e-mail: iryna.perova@nure.ua. ORCID: <https://orcid.org/0000-0003-2089-5609>.

Мірошниченко Неля Сергіївна, аспірант кафедри системотехніки ХНУРЕ, м. Харків, Україна, e-mail: nelia.miroshnychenko@nure.ua. ORCID: <https://orcid.org/0000-0002-3846-1668>.

УДК 004.89:61

DOI: 10.30837/0135-1710.2023.179.050

І.Ю. ПАНФЬОРОВА, А.С. БУЦЬКА

ДОСЛІДЖЕННЯ АНСАМБЛЮВАННЯ МОДЕЛЕЙ MACHINE LEARNING В МЕДИЧНІЙ ДІАГНОСТИЦІ

Розглянуто застосування моделей машинного навчання (machine learning – ML) в медичній діагностиці. Представлено основні виклики та цілі останніх досліджень у сфері медичного прогнозування. Основну увагу зосереджено на порівнянні існуючих моделей ML. Проведено аналіз вже існуючих рішень для розробки ансамблю моделей ML. Розраховано ключові характеристики моделей ML: точність, чутливість, специфічність та AUC-ROC. Запропоновано варіант об'єднання цих моделей в ансамбль для покращення точності – основної характеристики прогнозування діагнозу.

1. Вступ

Медична діагностика є критично важливим аспектом охорони здоров'я, оскільки її рівень безпосередньо впливає на точність результатів діагностування і, як наслідок, на вірність рішення щодо лікування пацієнтів. Нова ера в медичній діагностиці розпочалася з появою машинного навчання (machine learning – ML), яке надало розширені інструменти для аналізу великих обсягів медичних даних, включаючи історії хвороби пацієнтів, зображення та генетичну інформацію. Використання моделей ML може сприяти виявленню прихованих закономірностей і аномалій, а також прогнозуванню ризиків розвитку того чи іншого захворювання, що робить їх цінними інструментом для медичних працівників.

Один з напрямів сучасних досліджень в галузі застосування ML у медичній діагностиці полягає у ліквідації недоліків існуючих моделей ML шляхом їх комбінації. Цей різновид ML, який забезпечує покращення продуктивності і точності прогнозів за рахунок комбінації кількох моделей ML, отримав назву «ансамблеве навчання» [1]. Мета досліджень за цим напрямом полягає у порівнянні існуючих моделей ML, пропозиції множини варіантів та виборі найкращого варіанту вирішення задачі створення ансамблю для прогнозування діагнозу.

Ансамблеве навчання, на жаль, не вільне від деяких недоліків. Так непростою задачею є інтеграція моделей ML, адже до того, як впровадити моделі ML у вже існуючі медичні

інформаційні системи (МІС), слід провести значну роботу з даними, починаючи від їх збору, попередньої обробки та закінчуючи інтерпретацією отриманих результатів. Ансамблі моделей ML потребують більшого об'єму даних, аніж окремі моделі.

Головна проблема ансамблевого навчання полягає у виборі реалізації ансамблю з окремих моделей ML, оскільки не всі моделі добре поєднуються між собою, а невірний вибір може погіршити продуктивність ансамблю. У конкретних МІС використання окремих засобів ML показало їх перспективність, проте поєднання сильних сторін МІС та засобів ML може призвести не тільки до підвищення надійності прогнозів діагностики, а й до зниження швидкодії МІС та збільшення витрат для отримання бажаного результату. Ця проблема також породжує проблему розрахунку обчислювальної складності та ресурсів, необхідних для реалізації відразу декількох моделей ML. Тому проведення досліджень, спрямованих на вирішення зазначених вище проблем визнається необхідним як з теоретичної, так і з практичної точок зору.

2. Аналіз літературних джерел та визначення проблеми дослідження

У 2019 році було проведено огляд тенденцій у галузі використання моделей ML для діагностики [2]. Цей огляд розділив моделі на дві великі категорії: моделі, керовані даними, і моделі, керовані знаннями. Кожна з цих категорій додатково поділяється на контрольоване та неконтрольоване навчання. Моделі, керовані даними, зосереджені на аналізі та використанні великих обсягів даних для навчання моделей, в той час, як моделі, керовані знаннями, використовують збагачені бази експертних знань для вирішення конкретних завдань. Обидва підходи можуть взаємодіяти. Ансамблювання може використовувати моделі керовані даними для автоматичного вивчення залежності у великих обсягах історичних даних, а потім використовувати експертні знання для уточнення та покращення результатів діагностування.

Дослідження, опубліковані в [2], продемонстрували потенціал ML в медичній діагностиці. Моделі, керовані даними, такі, як згорткові нейронні мережі (CNN) продемонстрували виняткову продуктивність в аналізі медичних зображень, включаючи виявлення пухлин на радіологічних зображеннях. Ключова перевага моделі CNN перед своїми попередниками полягає в тому, що вона визначає основні функції без потреби втручання людини [3]. Зазвичай CNN використовує зображення як вхідні дані. Така модель призначає вагові коефіцієнти різним частинам зображення для отримання диференціації пікселів та їх просторового розташування, що дозволяє виявляти різноманітні аспекти та характеристики, такі як границі об'єктів, текстури та інші ключові елементи. CNN застосовують операцію згортки принаймні на одному рівні замість простого множення матриць [4], що відрізняє CNN від інших моделей.

Контрольована модель навчання «Машина векторів підтримки» (Support Vector Machine –SVM) принесла користь багатьом напрямкам прогнозування, таким, наприклад, як прогнозування захворювань на основі даних, отриманих за допомогою магнітно-резонансної томографії (МРТ). Дослідники використовували SVM для діагностики нервово-м'язових розладів [5]. Обробка природної мови (NLP) була застосована для вилучення цінної інформації з неструктурованих клінічних записів, що допомагає класифікувати захворювання та оцінювати ризик. Рекурентні нейронні мережі (RNN) здатні розпізнавати послідовності. Існує багато варіантів реалізації RNN, таких як LSTM, BLSTM, MDLSTM і HLSTM. Недоліком RNN є виникнення значних ускладнень у його застосуванні у випадках зникнення градієнта. Крім того, RNN для своєї експлуатації потребують великих наборів даних, зокрема, наборів даних медичних зображень. Але такі набори занадто обмежені в порівнянні з наборами даних для загальних проблем комп'ютерного зору, які зазвичай коливаються від кількох сотень тисяч до мільйонів анотованих фотографій [6].

Загальним недоліком моделей ML є те, що через складність структур даних навчання моделей може стати дороговартісним завданням. Іноді для обробки цих структур даних

необхідні потужні графічні процесори та сотні комп'ютерів, що призводить до збільшення вартості для кінцевих користувачів [7].

Оскільки при використанні декількох моделей у ансамблевому навчанні складність обчислень підвищується, виникає високе обчислювальне навантаження, що негативно впливає на продуктивність процесу навчання моделей. Для подолання проблеми зникання градієнта і перетримки використовуються вдосконалені функції активації, архітектура функції вартості та методи видалення, описані в [8]. Вирішити проблему великої обчислювальної навантаженості допомагає використання ресурсів такого формату як графічні процесори та пакетна нормалізація. ML добуває інформацію з великої кількості даних і генерує результати, які можна використовувати для індивідуального прогнозування та прийняття клінічних рішень [9]. Цей підхід відкриває шлях до розвитку персоналізованої медицини, де враховуються генетичні фактори та спосіб життя людини для профілактики захворювань, лікування та прогнозування розвитку захворювання [10].

3. Мета і задачі дослідження

Метою даного дослідження є розробка ансамблевої моделі для вирішення задачі прогнозування діагнозу за результатами медичних дослідів та порівняння ансамблевої моделі з існуючими одиничними моделями ML, котрі вже активно використовуються для прогнозування в медицині. Досягнення цієї мети дає поштовх для покращення існуючих рішень в медичній діагностиці та подальшого розвитку МІС шляхом вдосконалення існуючих рішень задачі прогнозування діагнозу.

Для досягнення поставленої мети пропонується вирішити такі задачі:

- розробити схематичне зображення роботи ансамблевої моделі;
- зібрати достатній обсяг даних з медичних досліджень;
- провести дослідження на окремих моделях ML та запропонованій ансамблевій моделі;
- порівняти отримані результати.

4. Матеріали і методи дослідження

Як зазначено в [1] у загальному вигляді схему ансамблевого навчання моделей ML може бути представлено у вигляді комбінації кроків, у якій дані можна тренувати на різних базових класифікаторах, а вихідні дані об'єднуються для отримання остаточного прогнозу (див. рис. 1).

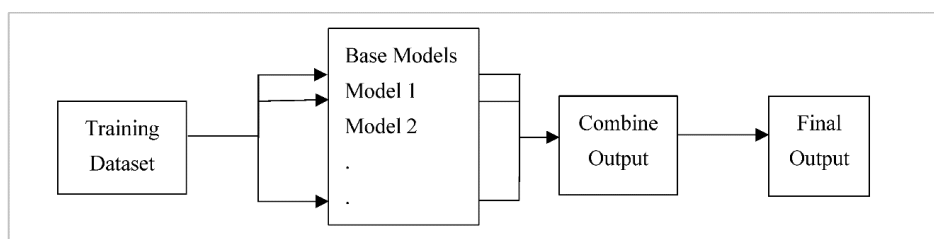


Рис. 1. Схема ансамблевого навчання моделей ML

Рис. 1 ілюструє базове представлення реалізації ансамблювання моделей в ML, де різні моделі можуть навчатися на вхідних даних, а на виході отримується об'єднаний результат. Існують різні підходи до створення ансамблевих моделей. Одними з найпопулярніших таких підходів є stacking, bagging, boosting та voting.

Stacking передбачає використання декількох різнорідних слабких моделей, які навчаються незалежно одна від одної, а потім об'єднуються для створення прогнозу на основі результатів кожної з моделей. Bagging, з іншого боку, передбачає навчання однорідних

моделей на різних наборах даних і їх об'єднання. Прогноз отримується шляхом усереднення прогнозів кожної з моделей. Boosting – це підхід послідовного навчання декількох однорідних моделей, при якому кожна наступна модель виправляє помилки попередніх моделей. Voting – це підхід, який агрегує прогнози з численних незалежних моделей (базових оцінювачів), щоб зробити остаточний прогноз [11].

Як наслідок, в результаті ансамблювання кілька моделей ML поєднуються для створення точніших прогнозів, ніж ті, що реалізовані за допомогою одного класифікатора [12]. Кожен тип ансамблювання детально розглянуто в [1]

Для моделі ансамблю запропоновано обрати три моделі (SVM, CNN і LSTM), оскільки вони продемонстрували високу продуктивність та мають попит в МІС (рис. 2).

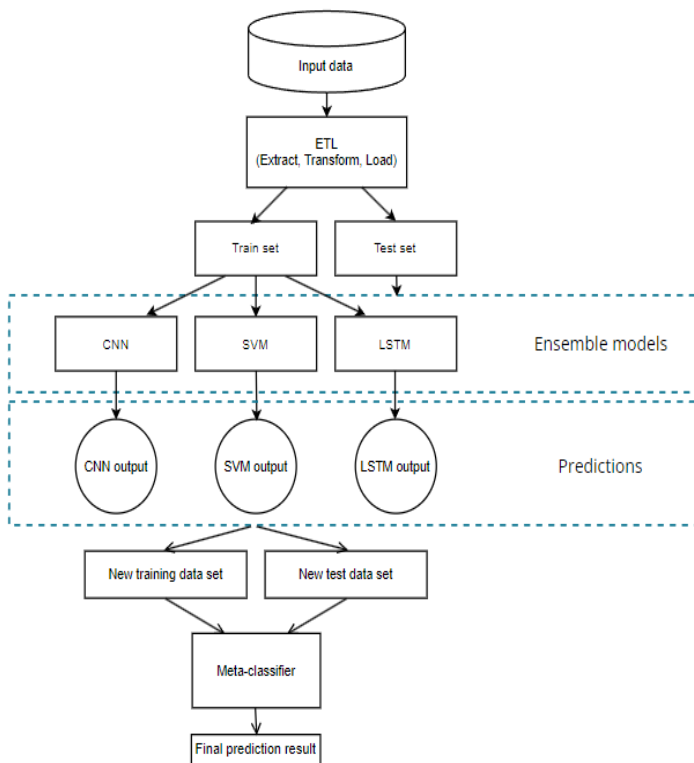


Рис. 2. Схема архітектури моделі ансамблевого навчання з використанням підходу stacking на основі моделей SVM, CNN та LSTM

акцент на діагностуванні раку шкіри, а саме, створенні класифікатора меланому, який зможе з достатньою точністю розрізнати доброякісні (неракові) та злоякісні (ракові) ділянки шкіри.

Для того, щоб дослідити запропоновану модель ансамблю, було використано тестові та навчальні зразки зображень. Для доповнення існуючих зображень, пов'язаних із захворюваннями шкіри, була використана текстова інформація щодо атрибутів цих зображень, наведена у файлі формату .csv [13]. Досліди проводилися на портативному комп'ютері з процесором Intel Core(TM) i5-6300HQ @ 2,30 ГГц x 4, 16 Гб оперативної пам'яті DDR3 і графічним процесором NVIDIA GeForce GTX 960M 4 Гб DDR5.

У цьому дослідженні для вирішення проблеми, пов'язаної з тим, що деякі вхідні дані були неповними або необробленими, використано додаткову інформацію з матриці стохастичних

Дослідження передбачає навчання та оцінку обраних моделей на наборах даних, зібраних з реальних медичних закладів. Ці набори складаються з радіологічних зображень, клінічних записів, генетичних даних та історій пацієнтів, охоплюють широкий спектр захворювань і забезпечують необхідну різноманітність для комплексного аналізу.

Тому дослідження пропонується проводити за допомогою даних із тестового набору даних з відкритого репозиторію даних Kaggle, що належить корпорації Google [13].

5. Результати дослідження

Серед поширених захворювань, які можуть значно вплинути на здоров'я пацієнтів, зазвичай виділяють діабет, рак шкіри, захворювання нирок, печінки та серцеві захворювання. У цьому дослідженні зроблено

пертурбацій. Матриця стохастичних пертурбацій, кожен елемент якої являє собою випадкове число, використовується для введення випадкових значень в дані під час тренування моделі. Тому матриця доповнення формується шляхом додавання до вхідних даних матриці стохастичних пертурбацій:

$$E = x_{i \times j} + D_{i \times j}, \quad (1)$$

де E – розширена матриця; x – вхідні дані; D – матриця стохастичних пертурбацій.

В даному випадку значення елементів матриці стохастичних пертурбацій мають бути значно меншими за вихідні дані, щоб уникнути зміни характеристик необроблених даних. У той же час розмірність стохастичної матриці пертурбацій має відповідати розмірності матриці вхідних даних.

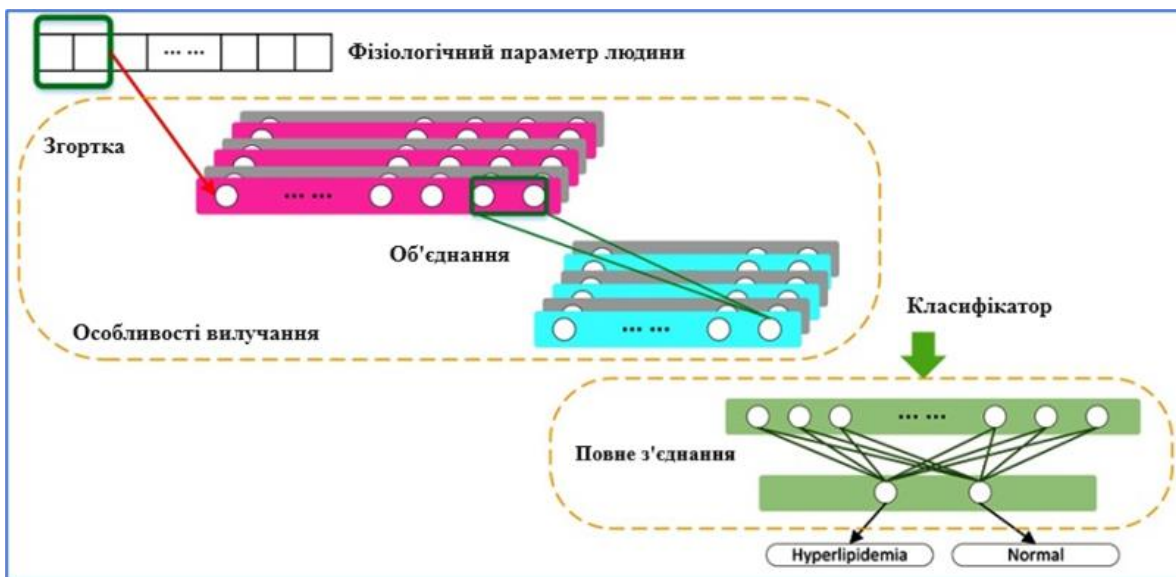


Рис. 3. Розширення алгоритму навчання

Було також проведено дослідження впливу кількості і точності інформації на результативність моделі в досліді. На етапі кореляції даних значення фізіологічних параметрів з негативними результатами були замінені значеннями, близькими до нуля, але не рівними нулю.

Для одночасного навчання моделей ML у дослідженні було використано необроблені та розширені дані. Рис. 3 демонструє основний принцип розширення алгоритму навчання. Експеримент проводився у середовищі Ubuntu 16.04 з використанням програмного забезпечення Keras, яке базується на Tensorflow.

Для порівняння результатів моделювання було використано такі характеристики.

Точність *Accuracy* характеризує загальну правильність передбачень як відношення кількості правильних прогнозів до загальної кількості передбачень.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}, \quad (2)$$

де TP (True Positives) – кількість правильно передбачених позитивних класів; TN (True Negatives) – кількість правильно передбачених негативних класів; FP (False Positives) – кількість хибнопозитивних передбачень; FN (False Negatives) – кількість хибнонегативних передбачень.

Чутливість *Recall* характеризує частку справді позитивних випадків, правильно ідентифікованих, як відношення справжніх позитивних випадків до суми справжніх позитивних та помилкових негативних випадків.

$$Recall = \frac{TP}{(TP + FN)}. \quad (3)$$

Специфічність *Specificity* характеризує частку справді негативних випадків, правильно визначених, як відношення справжніх негативних випадків до суми справжніх негативних та помилкових позитивних випадків.

$$Specificity = \frac{TN}{(TN + FP)}. \quad (4)$$

AUC-ROC характеризує здатність моделі розрізняти класи, причому вищі значення вказують на кращу продуктивність. Точний розрахунок AUC-ROC вимагає побудови кривої ROC та інтегрування площі під нею, що зазвичай виконується за допомогою чисельних методів або спеціалізованого програмного забезпечення.

З результатами досліджень можна ознайомитись в табл. 1.

Таблиця 1

Результати, отримані в ході дослідження різних моделей

| Модель | Точність | Чутливість | Специфічність | AUC-ROC |
|--------------|----------|------------|---------------|---------|
| LSTM | 0.88 | 0.92 | 0.86 | 0.91 |
| SVM | 0.82 | 0.88 | 0.78 | 0.86 |
| CNN | 0.89 | 0.93 | 0.87 | 0.92 |
| CNN-SVM-LSTM | 0.95 | 0.96 | 0.93 | 0.95 |

Значення в табл. 1 було обчислено за допомогою формул (2)-(4) на основі прогнозів моделі та справжніх міток класу в тестовому наборі даних.

Аналіз наведених в табл.1 результатів, отриманих в ході дослідження моделей ML, показав, що окремі моделі (LSTM, SVM, CNN) демонструють добрі показники для характеристик, котрі наведені в табл. 1, але інтегрована модель, яка об'єднує CNN, SVM і LSTM, показує загальне покращення точності діагностики. Використання ансамблювання моделей дозволило досягти значення оцінки AUC-ROC в 0.95, що перевершує значення оцінки індивідуальної моделі діагностування за допомогою тільки CNN моделі ML, яке дорівнює 0.89.

Запропонована ансамблева модель також надала інтерпретовані пояснення своїх рішень, що сприяло клінічному прийняттю результатів моделювання. Таке сприйняття важливе для медичних працівників, щоб зрозуміти прогнози та довіряти їм.

6. Обговорення результатів дослідження та висновки

Проведені дослідження показали, що впровадження ансамблевих моделей в МІС сприяє подоланню розриву між теорією ML та практичним застосуванням цих моделей в сфері

охорони здоров'я. Адаже поєднання кількох моделей ML забезпечує кращу продуктивність не лише при вирішенні задачі автоматичного діагностування, а й в інших сферах повсякденного життя, про що свідчить, наприклад, робота [14]. Для вирішення задачі прогнозування діагнозу за результатами медичних дослідів запропоновано об'єднати моделі, які показали добрі результати при вирішенні задачі класифікації, обробки візуальної інформації та обробки тексту.

В роботі було запропоновано ансамблеву модель, яка є результатом об'єднання моделей CNN, SVM та LSTM. Розроблено схему архітектури моделі ансамблевого навчання з використанням підходу *stacking* на основі моделей SVM, CNN та LSTM. Застосування цієї схеми сприяє підвищенню точності діагностики, що може сприяти ранньому виявленню захворювання та підвищенню ефективності лікування.

Для експериментальної перевірки отриманих рішень використаний набір даних з медичних досліджень, з яким можна ознайомитися в [13]. Проведено попередню обробку даних для забезпечення статистичної значущості та забезпечення репрезентативності вибірки.

Проведено дослідження на окремих моделях ML та запропонованій ансамблевій моделі та виконано порівняння отриманих результатів. Підтверджено перспективність поєднання існуючих моделей ML для підвищення точності медичних діагнозів. Ансамблева модель CNN-SVM-LSTM випереджає окремі моделі за точністю, чутливістю, специфічністю, а також забезпечує можливість інтерпретації результатів прогнозування.

В результаті дослідження було отримано порівняльну оцінку обраних моделей ML. За результатами оцінювання можна зробити висновок, що запропонована ансамблева модель досягла задовільних результатів і може бути запропонована до використання при автоматизації процесу діагностування, адже точність ансамблевої моделі досягла значення в 95%, що свідчить про надійність моделі.

Основними проблемами залишаються: відсутність стандартизованих протоколів збору даних і анотацій, що може призвести до упередженості та обмежити можливість узагальнення моделей; складність у отриманні достатнього обсягу даних про стан пацієнта; проблеми з конфіденційністю даних; проблеми з інтеграцією ML в клінічні робочі процеси; неоднорідність даних з різних джерел, що може вимагати різних етапів попередньої обробки та інтеграції даних для ефективного поєднання.

У цілому ансамблювання моделей машинного навчання є перспективним напрямком досліджень, і в майбутньому буде використовуватися для вирішення складніших завдань та забезпечувати точніші прогнози і класифікації в різних галузях. Майбутні дослідження в цьому напрямку повинні бути зосереджені на зазначених вище проблемах. Однак слід враховувати, що ансамблювання може бути більш обчислювально вимогливим, оскільки вимагає навчання та прогнозування більшої кількості моделей, що може збільшити час та ресурси, необхідні для обробки даних.

Перелік посилань

1. Mahajan P., Uddin S., Hajati F., Moni M. A. Ensemble learning for disease prediction: A Review. *Healthcare*. 2023. Vol. 11. No. 12. P 1808. doi:10.3390/healthcare11121808
2. Ali R., Hardie R. C., Narayanan Narayanan B., De Silva S. Deep learning ensemble methods for skin lesion analysis towards melanoma detection. *2019 IEEE National Aerospace and Electronics Conference (NAECON)*. 2019. doi:10.1109/naecon46414.2019.9058245.
3. Jain G., Mittal D., Thakur D., Mittal M. K. A deep learning approach to detect covid-19 coronavirus with X-ray images. *Biocybernetics and Biomedical Engineering*. 2020. Vol. 40. No. 4. P. 1391–1405. doi:10.1016/j.bbe.2020.08.008.
4. Noor M.B. et al. Application of deep learning in detecting neurological disorders from Magnetic Resonance Images: A survey on the detection of alzheimer's disease, parkinson's disease and schizophrenia. 2020. P. 11. doi:10.1186/s40708-020-00112-2.

5. Jeena R.S., Kumar S. Stroke prediction using SVM. *2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*. 2016. doi:10.1109/iccicct.2016.7988020.
6. Apostolopoulos I. D., Mpesiana T. A. COVID-19: Automatic detection from X-ray images utilizing transfer learning with Convolutional Neural Networks. *Physical and Engineering Sciences in Medicine*. 2020. Vol. 43. No. 2. P. 635–640. doi:10.1007/s13246-020-00865-4.
7. Ballin A, Karlinsky L, Alpert S, Hasoul S, Ari R, Barkan E. A region based convolutional network for tumor detection and classification in Breast Mammography. *Deep Learning and Data Labeling for Medical Applications*. 2016. P. 197–205. doi:10.1007/978-3-319-46976-8_21.
8. Anavi Y., Kogan I., Gelbart E., Geva O., Greenspan H. Visualizing and enhancing a deep learning framework using patients age and gender for chest X-ray image retrieval. *Medical Imaging 2016: Computer-Aided Diagnosis*. 2016. doi:10.1117/12.2217587.
9. Hassan M., Ali S., Alquhayz H., Safdar K.. Developing Intelligent Medical Image Modality Classification system using Deep Transfer Learning and LDA. *Scientific Reports*. 2020. Vol. 10. No.1. doi:10.1038/s41598-020-69813-2.
10. Abbas A., Abdelsamea M. M., Gaber M. M. Classification of covid-19 in chest x-ray images using DeTraC deep convolutional neural network. *Applied Intelligence*. 2020. Vol. 51. No. 2. P. 854–864. doi:10.1007/s10489-020-01829-7.
11. Jani R., Shariful Islam Shanto Md., Mohsin Kabir Md., Saifur Rahman Md., Mridha M. F. Heart disease prediction and analysis using ensemble architecture. *2022 International Conference on Decision Aid Sciences and Applications (DASA)*. 2022. doi:10.1109/dasa54658.2022.9765237.
12. Suganyadevi S., Seethalakshmi V., Balasamy K. A review on deep learning in Medical Image Analysis. *International Journal of Multimedia Information Retrieval*. SpringerLink. <https://link.springer.com/article/10.1007/s13735-021-00218-1> (accessed Oct. 16, 2023).
13. Siim-ISIC melanoma classification. Kaggle. <https://www.kaggle.com/competitions/siim-isic-melanoma-classification/discussion/162486> (accessed Oct. 16, 2023).
14. Livieris I. E., Pintelas E., Stavroyiannis S., Pintelas P. Ensemble deep learning models for forecasting cryptocurrency time-series. *Algorithms*. 2020. Vol. 13. No. 5. P. 121. doi:10.3390/a13050121.

Надійшла до редколегії 17.10.2023

Панфорова Ірина Юрївна, кандидат технічних наук, доцент, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: iryna.panforova@nure.ua. ORCID: <https://orcid.org/0000-0001-7032-9109>.
Буцька Анастасія Сергіївна, здобувачка вищої освіти гр. ІУСТМ-22-1, факультет комп'ютерних наук ХНУРЕ, м. Харків, Україна, e-mail: anastasiia.levchenko@nure.ua.

А.Є. КОЖАНОВ

ДОСЛІДЖЕННЯ МОДЕЛЕЙ ФОРМАЛЬНОГО ОПИСУ ЗНАНЬ ПРО ФУНКЦІОНАЛЬНІ ВИМОГИ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ EER-ДІАГРАМИ

Розглянуто моделі формального опису знань про функціональні вимоги до інформаційної системи. Досліджено формальний апарат, що використовує фреймову модель для відображення опису функціональних вимог на рівні знань з використанням діаграми класів та ER-діаграми, та запропоновано формальний апарат відображення спільних елементів діаграми класів, ER-діаграми та EER-діаграми. Проаналізовані отримані результати.

1. Вступ

Однією з найважливіших проблем, що виникають при спробах описати вимоги до інформаційної системи (ІС) формальними способами, є практично постійна неповнота вимог. Ця неповнота викликана, насамперед, прагненням Споживача отримати тільки ті ІТ-послуги, експлуатація яких забезпечить якнайшвидшу появу прибутку від процесу, що автоматизується. При цьому Споживач не замислюється про необхідність додаткових ІТ-послуг для забезпечення ефективної злагодженої роботи основних ІТ-послуг. Крім того, слід пам'ятати про можливість змін процесу, що автоматизується, із часом (наприклад, внаслідок реінжинірингу або поступового поліпшення).

Вирішення цієї проблеми стає можливим у тому випадку, якщо припустити, що Постачальник і Споживач у кожному конкретному проєкті створення ІС мають справу не з простою множиною вимог до ІС, повністю або частково впорядкованих певним чином, а з загальнішою сукупністю вимог. При цьому до такої сукупності мають входити не тільки вимоги до ІС, висунуті Споживачем і прийняті до виконання Постачальником, але й так звані «забуті» (невиявлені або несвоєчасно виявлені) вимоги, які Постачальник може висунути до ІС, виходячи зі свого бачення предметної області (Про); усі ці вимоги можуть раніше або пізніше стати елементами множини вимог до ІС у ході чергової ітерації процесів, що безпосередньо працюють із вимогами. Тому перед побудовою системи дуже важливо зафіксувати вимоги. Позначення вимог за допомогою точної й однозначної моделі дозволяє гарантувати, що всі зацікавлені особи розуміють вимоги й згодні з ними.

Множина вимог правовласників є хронологічно першим системним описом створюваної ІС і являє собою систему не як єдиний цілісний продукт, а як набір не пов'язаних один з одним описів окремих елементів системи. Дана множина містить у собі різні вимоги, на основі яких формується множина сценаріїв, що описують окрему функціональну можливість створюваної системи. Вимоги правовласників і сценарії, що ідентифікують функціональні можливості системи, виражаються в термінах моделей (текстових або формальних), які орієнтовані на мету й поведінку системи і описують її в контексті середовища й умов функціонування.

На основі множини вимог правовласників і сценаріїв, яка ідентифікують функціональні можливості системи, у ході виконання процесу аналізу вимог формується системний опис, що являє собою множину системних вимог. Серед цих вимог можна виділити [1]:

- а) функціональні вимоги;

- б) експлуатаційні вимоги;
- в) вимоги, пов'язані з такими властивостями системи, як здоров'я, безпека, захищеність, безвідмовність, готовність;
- г) вимоги, пов'язані із властивостями систем забезпечення;
- д) технічні показники й показники якості під час використання.

Крім того, множина системних вимог доповнюється множиною обмежень, серед яких слід особливо виділити множину функціональних меж системи в термінах її поведінки й властивостей, які повинні бути забезпечені. Рекомендації зі способу вираження системних вимог і обмежень в [1] не наводяться.

Аналіз прийнятих у сучасних стандартах системних описів створюваної ІС на ранніх стадіях її життєвого циклу (ЖЦ) дозволяє зробити такі висновки:

а) рекомендовані [1] системні описи, сформовані на ранніх стадіях ЖЦ ІС, орієнтовані на представлення створюваної системи як множини окремих вимог, функцій і архітектурних елементів, а не єдиного цілісного проєкту;

б) у ході виконання процесів формування вимог правовласників, аналізу вимог і проєктування архітектури задачі системної інтеграції не вирішуються й не ставляться;

в) використання при формуванні системних описів досвіду, накопиченого в ході виконання попередніх проєктів створення аналогічних систем, не розглядається;

г) рішення про доцільність початку ІТ-проєкту створення ІС має ухвалюватися на основі множини вимог правовласників і не базується ані на досвіді створення ІС аналогічного призначення, ані на представленні ІС як єдиного цілісного продукту, що серйозно спотворює оцінки обсягу робіт зі створення ІС.

Дані висновки дозволяють стверджувати, що в ході створення ІС існують серйозні проблеми на ранніх стадіях її ЖЦ, що значно підвищує витрати на створення ІС й збільшує кількість інцидентів, які можуть виникнути в ході розробки частини забезпечення даної ІС. У той самий час практика виконання проєктів (у тому числі – ІТ-проєктів) свідчить [2], що зміни ІС за вимогами зацікавлених сторін найефективніші й вимагають менших витрат переважно на ранніх стадіях її ЖЦ. Тому вирішення зазначених проблем є одним з найперспективніших напрямків науково-прикладних досліджень в ІТ-сфері.

Результати аналізу сучасного представлення ІС показують, що вимоги до ІС є основною вихідною інформацією для проєктування архітектури ІС, на основі якої виконуються роботи з розробки систем забезпечення. Тому необхідно проаналізувати сучасні представлення вимог до ІС із метою виділення їх недоліків, що призводять до збільшення витрат на створення або модернізацію ІС.

Результати сучасних процесів візуального моделювання вимог до ІС, призначених для автоматизації керування підприємствами й організаціями, є одними з найважливіших факторів, що визначають успішність виконання всіх наступних робіт із проєктування й розробки в рамках або створення нової ІС, або модифікації наявної системи. В основі цих процесів лежить формування опису ІС, що створюється або модифікується, як сукупності вимог до системи з подальшим перетворенням цих описів спочатку в опис архітектури ІС, а згодом – в описи специфікацій на розробку елементів комплексів забезпечення.

2. Аналіз систем класифікації вимог і постановка проблеми дослідження

У ході аналізу розглянуті класифікації вимог до ІС, що використовуються в таких методологіях і роботах дослідників:

- а) методологія SSADM [3];
- б) дослідження, виконані Д. Леффінгуеллом і Д. Уїдригом [4], а також К. Вігерсом [5];
- в) методологія, що лежить у основі пакета керування вимогами Rational DOORS [6];
- г) модель класифікації вимог FURPS і її сучасний варіант FURPS+ (компанія «Hewlett-Packard») [7], [8];
- д) звід знань з аналізу бізнес-процесів (БП) сучасних підприємств (Business Analysis Body of Knowledge, BABOK) [9], у тому числі – з метою визначення доцільності автоматизації даних БП.

Дані варіанти описують усі основні точки зору на вимоги до ІС як до технічного ІТ-продукту, необхідного для досягнення економічних або техніко-економічних цілей.

За результатами аналізу можна зробити такі висновки [10]:

- а) єдиної системи класифікації вимог до ІС не існує;
- б) у теперішній час більшість варіантів класифікації вимог розглядають вимоги до системи як проміжну ланку між бізнес-вимогами й вимогами до програмного забезпечення (ПЗ);
- в) нефункціональні вимоги розглядаються як спосіб уточнення інших вимог у ході їх аналізу;
- г) у процесі проєктування архітектури системи увага зосереджується, головним чином, на проєктуванні архітектури ПЗ.

Хоча загальноприйнятні визначення поняття «вимога до системи» вже запропоновані, єдиної системи класифікації вимог досі не існує. Тому способи формального опису вимог до ІС багато в чому визначаються тими методами збору вимог, які застосовуються під час виконання процесу визначення вимог правласників.

У ІТ-спільноті сформована думка, відповідно до якої жодне представлення вимог тільки одним способом не дає їх повної картини. Цю думку обумовлюють використанням ієрархії DIKW (Data, Information, Knowledge and Wisdom) для опису процесів, що безпосередньо працюють із вимогами, а також процесів проєктування архітектури системи. Порівняння представлень вимог, отриманих різними фахівцями під час різноманітних досліджень, допомагає виявити невідповідності, неясності, допущення й недогляди, які важко виявити, коли вимоги представлені в одному форматі. Слід зазначити, що класична модель DIKW припускає чітку спрямованість перетворень даних в інформацію, інформації – у знання, а знань – у мудрість. Однак у процесах, що безпосередньо працюють із вимогами, часто основою для моделювання вимог до ІС є представлення цих вимог на рівні інформації, виражене у вигляді їх текстових описів або візуальних моделей (ВМ). При цьому процес проєктування архітектури системи припускає можливе уточнення ВМ вимог описом цих вимог на рівні даних у вигляді тієї або іншої атрибутивної моделі [11].

Водночас використання атрибутивних моделей, або аналогічних їм, для представлення вимог до ІС на рівні даних ускладнене, насамперед, через прагнення об'єднати в рамках однієї атрибутивної моделі вимог до ІС набори атрибутів, що використовуються у принципово різних цілях. Тому виникає необхідність розгляду кожної вимоги до ІС як первісного різноманіття представлень цієї вимоги у вигляді даних, інформації й знань. Існування такого різноманіття обумовлене такими чинниками [12]:

- а) представлення вимог до ІС на рівні інформації призначене для опису різними способами елементів об'єктів або процесів, що автоматизуються, елементів розроблюваної ІС

або ІС у цілому з метою забезпечення можливості виконання проєкту створення ІС, до якої висунуто вимогу шляхом виконання послідовності перетворень «неформалізований опис вимоги звичайною мовою – частково формалізований опис вимоги – формальний опис вимоги у вигляді набору цільових показників, значення яких характеризують ступінь задоволення вимоги»;

б) представлення вимог на рівні даних призначене для формування описів вимоги до ІС і її окремих представлень, взаємно узгоджених одна з одною, з метою забезпечення можливості здійснення стандартних операцій уведення, відображення, редагування й виділення вимоги до ІС і її окремих представлень у рамках інформаційної технології (ІТ) формування й аналізу вимог до ІС, а також керування окремими вимогами в ході виконання проєкту створення ІС;

в) представлення вимог до ІС на рівні знань призначене для виявлення знань про об'єкти або процеси, що автоматизуються, розроблюваних елементах ІС або ІС у цілому з метою забезпечення можливості повторного використання цих знань у проєктах створення інших ІС.

Основні положення концепції представлення вимог до ІС визначають загальні формалізовані описи вимог. Вони мають містити в собі:

- формалізовані описи вимог до ІС (як відомих, так і невідомих сторонам);
- формалізовані описи методів формування вимог до ІС;
- формалізовані описи представлень вимог до ІС на рівнях даних, інформації й знань.

Крім того, загальні формалізовані описи вимог до ІС додатково мають містити в собі [10]:

– формалізовані описи методів, прийомів і способів перетворення представлень різних вимог, виконаних на тому самому рівні, самих у себе й одна в одну, що дозволить установити й відображати взаємозв'язки між окремими вимогами групи, обумовлені особливостями процесу, що автоматизується;

– формалізовані описи методів, прийомів і способів перетворення представлень вимог до ІС, виконаних на одному рівні, у представлення вимог, виконаних на іншому рівні.

Розглянуті умови створення формалізованих описів вимог до ІС змушують відмовитися від класичних теоретико-множинних описів вимог. Математичний апарат, на основі якого слід розробляти формалізовані описи груп вимог до ІС, має дозволяти:

– розглядати вимоги до ІС як єдину цілісну систему – прообраз розроблюваної ІС – і одночасно як множину пов'язаних один з одним елементів, що перетворюються один в інший, відповідно до методології й технологій формування й аналізу вимог;

– враховувати вплив на представлення вимог до ІС конкретних методів, прийомів і способів формування й перетворення представлень цих вимог.

При формулюванні вимог до ІС усе різноманіття синтаксичних конструкцій і можливих синонімів повинне зводитися до єдиного зрозумілого для всіх сторін представлення цього поняття у вигляді даних, інформації або знань.

На практиці найчастіше зустрічається, що розробники переходять від моделей, що описують БП підприємства, до моделей, що описують ІС керування цим підприємством, а від них, у свою чергу, – до моделей, що описують окремі види забезпечень ІС. Можливі й зворотні переходи – наприклад, у ході реверсінжинірингу (зворотного проєктування) бази даних (БД) ІС.

Із прикладної точки зору з одного боку передбачається формування й реалізація якогось загального алгоритму або бази правил перетворення початкового представлення вимоги до ІС на кінцеве представлення цієї ж вимоги або множини інших вимог. При цьому мається на

увазі, що формат текстових конструкцій або елементів ВМ, які описують ці вимоги, апріорно співпадає з форматом вихідних даних алгоритму або системи правил і практично не змінюється з часом. Тому даний спосіб найбільше підходить для таких описів, що зводять до мінімуму кількість елементів алфавіту мови опису або моделювання вимог до ІС. До подібних описів можна віднести більшість розповсюджених ВМ ІС і її компонентів – діаграми потоків даних, діаграми «сутність – зв'язок» (Entity-Relation Diagram, ERD), діаграми класів (ДК) тощо. З іншого боку передбачається створення й постійний розвиток якоїсь множини алгоритмів або якоїсь системи правил перетворення початкового представлення вимоги до ІС у кінцеве представлення цієї ж вимоги або множини інших вимог. При цьому темпи зміни цієї множини алгоритмів або систем правил визначаються темпами появи нових або зміни існуючих понять тезауруса початкового представлення вимоги до ІС. Поява кожного нового, раніше невідомого поняття в такому представленні вимоги до ІС (у вигляді даних, інформації або знань) вимагає доповнення існуючого варіанта реалізації новими алгоритмами або правилами, що встановлюють спосіб і форму опису такого нового поняття в кінцевому представленні цієї вимоги.

Як впливає з моделей сформульованих вимог до ІС і універсума вимог до ІС, процеси розробки ІС [13], що безпосередньо працюють із вимогами, можуть розглядатися як послідовні перетворення представлень вимог у вигляді даних, інформації й знань під час виявлення й формулювання вимог до ІС відповідно до обраної методології розробки ІС. Однак ця модель залишає відкритим питання про вид представлень окремих вимог до ІС, методах і способах формування цих представлень і перетворень цих представлень самих у себе й одне в одне.

Функціональні вимоги, здебільшого, висуваються до таких елементів ІС, як інформаційне забезпечення (ІЗ) та ПЗ. Зазвичай, для опису архітектур цих частин, використовують ERD та ДК, відповідно. Та під час розробки, у деяких випадках, виникає проблема сумісності ІЗ та ПЗ через те, що діаграми виконані з лише таких точок зору на відповідні види забезпечення, для яких візуальні описи були створені. Ця проблема може бути вирішена уніфікацією цих описів. Існуючі технології розробки й коректування ВМ передбачають конвертацію окремих типів ВМ одна в одну відповідно до особливостей технологічних ланцюжків. Як такі технології і засоби їх реалізації можна відзначити технології SSADM, SADT, мову UML [14] і спеціалізований інструментальний засіб моделювання ARIS. Більшість технологій, що реалізують об'єктно-орієнтовані ВМ, припускають використання як допоміжних моделей структурних ВМ БП об'єкта автоматизації й ВМ потоків даних. Структурні ВМ застосовуються, головним чином, для визначення меж БП досліджуваного об'єкта автоматизації, а також для визначення меж проєктованої ІС.

Моделі слугують корисним інструментом аналізу проблем, обміну інформацією між усіма зацікавленими сторонами (користувачами, аналітиками, проєктувальниками тощо), проєктування ПЗ та підготовки документації. Моделювання сприяє повнішому засвоєнню вимог, поліпшенню якості системи й підвищенню ступені її керованості. Під терміном «моделювання» розуміється процес створення формалізованого опису системи у вигляді сукупності моделей. Під моделлю ПЗ в загальному випадку розуміється формалізований опис системи ПЗ певного рівня абстракції. Кожна модель визначає конкретний аспект системи, використовує набір діаграм і документів заданого формату, а також відбиває точку зору і є об'єктом діяльності різних людей з конкретними інтересами, ролями й задачами.

Графічні моделі являють собою засоби для візуалізації, опису, проєктування й документування архітектури системи. За ISO/IEC/IEEE 42010 «Systems and Software Engineering – Architecture Description», під архітектурою розуміються «фундаментальні поняття та властивості системи в навколишньому середовищі, втілені в її елементах, відносинах, а також у принципах її проєктування та розвитку.

Архітектурне представлення – це спрощений опис системи з конкретної точки зору, що охоплює певне коло інтересів й опускає об'єкти, несуттєві з даної точки зору.

Архітектурно значимі елементи – це елементи, що мають значний вплив на структуру системи та її продуктивність.

Погляд на архітектуру може бути представлений набором архітектурних моделей (Architecture Model). Кожна модель будується відповідно до конвенцій, установлених для даного виду моделі, причому ці конвенції, зазвичай, визначаються як частина архітектурної точки зору. Моделі забезпечують засоби для обміну деталями між поглядами на архітектуру й використанням множинних позначень в одному погляді на архітектуру.

Вид моделі (Model Kind) визначає конвенції, що задають правила побудови архітектурних моделей.

Мова моделювання включає [15]:

- елементи моделі – фундаментальну концепцію моделювання і семантики;
- нотацію (систему позначень) – візуальне представлення елементів моделювання;
- посібник з використання – правила застосування елементів у рамках побудови тих або інших моделей ПЗ.

Методологія візуального моделювання допомагає подолати деякі організаційні труднощі, що виникають перед командами розробників, групами підтримки якості й менеджерами, що беруть участь у створенні складного ПЗ масштабу підприємства.

Візуальне моделювання являє собою застосування для фіксування ескізів нотацій з розвиненою семантикою, графікою й текстовим змістом. Нотація, така як UML, дозволяє збільшити рівень абстракції, забезпечуючи повноцінний синтаксис і семантику. Таким чином вона поліпшує зв'язок у колективі, який працює над ескізом, і забезпечує точно виражену основу для реалізації.

Практика розробки ІС різноманітного призначення показує, що найсерйозніші протиріччя при спільному використанні структурного й об'єктного підходів до моделювання ІС виникають при стикуванні інформаційного й програмного комплексів ІС.

Інформаційний комплекс являє собою сукупність інтегрованих інформаційних, програмних, технічних, організаційних і лінгвістичних рішень, що забезпечують необхідні умови прийому, тривалого зберігання інформації й видачі її користувачам. Основу даного комплексу становлять рішення щодо структури БД ІС, організації запитів на видачу даних і процедур введення даних у базу. Для опису інформаційного комплексу найчастіше використовують ERD. Дані діаграми є достатньо поширеними й присутні в реалізаціях переважної більшості структурних технологій візуального моделювання ІС.

Існує також модель EER [16]. Вона була розроблена для точнішого відображення властивостей і обмежень, що є в складних БД, які застосовуються в галузях інженерного дизайну й виробництва, телекомунікаціях, пошукових системах, data mining, у складних програмних системах і геоінформаційних ІС. Дана модель є розширенням класичної ER-моделі за рахунок введення:

- субтипів й супертипів (інша назва – субкласи й суперкласи);
- спеціалізації та генералізації;
- спадкування атрибутів і зв'язків таблиць;
- типу категорій.

Нотацією EER-моделі є розширена діаграма «сутність – зв'язок» (Enhanced/Extended Entity-Relationship Diagram, EERD).

Програмний комплекс слід розглядати як об'єднання інтерфейсного й розрахункового комплексів. Він являє собою сукупність інтегрованих програмних, інформаційних, технічних і лінгвістичних рішень, що забезпечують необхідні умови взаємодії користувачів із проектованою ІС і реалізацію бізнес-логіки, адекватну реальним БП об'єкта автоматизації. Основу даного комплексу становлять, зазвичай, програмні продукти, що реалізують введення, обробку й відображення необхідної користувачеві інформації. Основні складнощі під час їх стикання виникають, як правило, у ході розробки й коректування за результатами тестування численних локальних реалізацій окремих елементів програмного комплексу, для чого потрібна досить значна кількість часу. Для опису програмного комплексу використовується цілий ряд ВМ, серед яких найзначущою є діаграма класів (ДК). Дана діаграма також є однією з найпоширеніших серед ВМ програмних систем і присутня в більшості об'єктно-орієнтованих технологій візуального моделювання ІС.

3. Мета і задачі дослідження

Метою дослідження є вирішення задачі уніфікації візуальних описів архітектур ІЗ та ПЗ ІС. Така уніфікація дозволить замість декількох ВМ, що описують специфікації вимог для розробників окремих видів забезпечень (ІЗ і ПЗ), використовувати єдину ВМ.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- провести аналіз існуючих підходів до формування вимог і створення описів архітектури складних ІС;
- провести аналіз існуючих ВМ, що використовуються у процесах проектування ІС;
- розробити чи модифікувати модель, яка буде використовуватися для формального опису знань про функціональні вимоги до ІС у процесах проектування архітектури системи та реалізації елементів ІЗ та ПЗ системи;
- розробити формальний апарат синтезу варіантів описів архітектури ІЗ та ПЗ ІС з використанням запропонованих технологій;
- розробити формальний апарат формування опису архітектури ІЗ і ПЗ ІС, заснований на результатах проведеного в роботі дослідження.

4. Моделі, що використовуються для формального опису знань про функціональні вимоги до інформаційної системи

Представленням вимоги до ІС на рівні знань пропонується вважати [10] онтологію елемента керованого об'єкта або процесу, елемента ІС або ІС у цілому, до яких висувається вимога. У той самий час прагнення повторного використання вимог до ІС обумовлює необхідність додаткового виділення онтологій елементів ІС або ІС у цілому, реалізованих у виконаних раніше проектах створення ІС. Загалом пропонується розглядати такі види онтологій:

- онтології ПрО, які являють собою знання про об'єкти або БП, що автоматизуються, добути під час виявлення й аналізу вимог до ІС;

– онтології реалізованих вимог до ІС, які являють собою знання про структури даних і процесах ІС, ІТ-послуг і ІТ-сервісів, створених у рамках попередніх проєктів;

– онтології вимог до створюваної ІС, які являють собою знання про структури даних і процесах ІС, ІТ-послуг і ІТ-сервісів, виокремлені під час формування й аналізу вимог у рамках поточного проєкту створення ІС.

Формування цих онтологій передбачається проводити на основі фреймової моделі знань. Застосування даної моделі обумовлене такими міркуваннями:

– використання фреймової моделі знань дозволяє застосовувати єдиний математичний апарат для опису знань про ПрО для формального представлення вимог до ІС, а також для опису знань, реалізованих в елементах ІС у вигляді моделей ПЗ даної системи;

– використання фреймової моделі дозволяє реалізувати взаємо-однозначне відображення представлень вимог до створюваної ІС на рівні знань в елементи ПЗ цієї ІС.

Для дослідження використовуються існуючі ВМ опису знань про функціональні вимоги, їх формальні описи, дослідження фреймової моделі представлення знань, дослідження формального апарату візуального опису функціональних вимог на рівні знань та формального апарату перетворення візуального опису функціональних вимог на рівні знань у описи ІЗ та ПЗ ІС.

Як ВМ узяті ДК, ERD і EER. Порівняльний аналіз елементів цих моделей дозволяє зробити такий висновок: описи зазначених елементів мають однакову структуру. Підмножина може бути відображена в множини класів. Однак, деталізуючи опис класів і сутностей, можна побачити, що деяка підмножина множини атрибутів ДК не може бути адекватно відображена в множини атрибутів ERD. Це пов'язане з тим, що атрибут в ERD зобов'язаний бути атомарною одиницею, а в ДК атрибут може мати тип і не бути атомарним. Але на рівні знань атомарність можна не враховувати, оскільки архітектори БД при розробці ІЗ враховують атомарність первинних ключів, а розробники ПЗ, у свою чергу, працюють зі своєю специфікою.

Пропонується застосовувати другий з описаних вище підходів. Взаємооднозначне відображення термінів ПрО в таблиці БД ІС вимагає застосування відповідної методики побудови, згідно з якою БД проєктується як набір взаємозалежних вітрин даних, реалізованих у вигляді окремих схем даних. У кожній із цих вітрин таблицею фактів є кореневий елемент конкретної ієрархії фреймів, а консольним таблицям і таблицям вимірів відповідають уточнюючі поняття ПрО. При цьому наслідувані фрейми представляються як слабкі сутності, що залежать від кореневого поняття відповідної ієрархії фреймів онтології ПрО, а кожна окрема вітрина даних являє собою багатомірний куб, який може бути згорнутий до одного універсального відношення, що містить усю множини можливих атрибутів конкретної вітрини даних.

У цьому випадку назви таблиць відповідають фреймам, атрибути таблиць – атрибутам фреймів, а приєднані процедури фреймів відображаються в збережені процедури й функції, а також у тригери реляційної схеми даних. Ієрархічні зв'язки відображаються в набори сутностей, пов'язаних між собою ідентифікуючими зв'язками, а зв'язки-асоціації – у посилальні обмеження цілісності даних, реалізовані із застосуванням первинних і зовнішніх ключів.

Оскільки дослідження та обговорення отриманих результатів проводиться на прикладі використання реляційної БД, треба розглянути особливості реалізації спадкування у подібних

БД докладніше. Для розв'язання даної задачі пропонується виділення абстрактної узагальненої сильної сутності, яка відповідає кореневому елементу конкретної ієрархії фреймів і є універсальною для різних ПрО. Атрибути такої сутності носять загальний характер, а її застосування припускається при реалізації вимог, характерних для декількох ІС. Для фреймів другого й вище рівнів ієрархії створюються слабкі сутності, які повністю залежать від загальної й універсальної сутності попереднього рівня ієрархії й доповнюють її, оскільки містять додаткову інформацію, характерну для конкретнішого класу об'єктів ПрО ІС. Така слабка сутність як ключовий атрибут (РК) містить зовнішній ключ (ФК), що реалізує зв'язок «1:0..1» з узагальнюючим сильним типом сутності. При цьому у випадку, коли відповідний їй фрейм має нащадків, така сутність у свою чергу є сильною для сутностей, відповідних до наступного рівня ієрархії.

Для спрощення роботи з БД ІС реалізуються віртуальні таблиці – представлення, які містять усю інформацію про об'єкт ПрО, поєднуючи дані з декількох таблиць (аналогічно тому, як в об'єктно-орієнтованому програмуванні класи-потокі успадковують усі атрибути класів батьків). Використання представлень дозволяє сховати розгалуженість схеми даних від користувачів і забезпечити відповідність логічного представлення даних конкретній ПрО.

Незважаючи на залежність слабкої сутності від сильної (така залежність виражається в тому, що записи в слабкій сутності не можуть існувати без відповідних записів у сильній сутності), розподіл набору атрибутів одного реального об'єкта ПрО між декількома таблицями, зв'язаними ідентифікуючими зв'язками й відношеннями «1:0..1» і створеними з метою реалізації в реляційній БД ієрархічними зв'язками, утворює необхідність забезпечення цілісності даних на рівні декількох таких таблиць. Подібна цілісність даних нехарактерна для класичної ситуації розміщення таких атрибутів в одній таблиці (на етапі концептуального проєктування таблиці, що мають зв'язки «1:1», завжди поєднуються в одну таблицю).

5. Результати дослідження

Якщо говорити про відображення ДК в EER, то, оскільки мова йде про рівень знань, то при перенесенні опису класу ДК в опис сутності EER можна перенести тільки знання (знання про назви класів відображаються у знання про назви сутностей, їх зв'язки, назви і типи змінних – у назви і типи атрибутів, тощо). Відображення операцій класу з діаграми класів на EER описане нижче. Варто вказати, що деякі досліджувачі зазначають, що в EER використовуються уточнення та узагальнення на основі нотації й термінології UML [17], яка є метамоделлю ДК.

Таким чином, можна говорити про можливість такого відображення:

$$\left\{ \begin{array}{l} at_{C_k}^j \rightarrow at_{E_i}^j \\ mt_{C_k}^b \rightarrow nmt_{E_i}^b, descmt_{E_i}^b \end{array} \right., \quad (1)$$

де $at_{C_k}^j, j=0, \dots, N$ – елемент множини атрибутів класу C_k ; $at_{E_i}^j$ – елемент множини атрибутів Atr_e сутності E_i ; $mt_{C_k}^b, b=0, \dots, M$ – опис b -ї операції класу C_k ; $nmt_{E_i}^b$ – елемент множини атрибутів Atr_e сутності E_i , який зберігає назву операції $mt_{C_k}^b$; $descmt_{E_i}^b$ – елемент множини атрибутів Atr_e сутності E_i , який зберігає текстовий опис операції $mt_{C_k}^b$.

Оскільки робота з EERD йде на рівні знань, то можна говорити, що, дотримуючись задуму її авторів [16], дана діаграма досить широко висвітлює знання про ПрО.

В основу запропонованих моделей покладено багатомірну модель даних, яка дозволяє найкращим способом зберігати й обробляти дані й метадані, що характеризують ПрО, раніше розроблені ІС і створювану ІС. Водночас запропоновані моделі можуть бути трансформовані в класичні схеми реляційних БД для спрощення практичної реалізації інтелектуальної ІТ формування й аналізу вимог до ІС.

Слід зазначити, що представлення моделей структурних патернів проектування вимог до ІС у вигляді кортежів дозволяє розглядати ці патерни як формалізовані висловлення певних вимог до способів організації описів вимог до ІС як об'єктів, автоматизоване оброблюваних у рамках інтелектуальної ІТ формування й аналізу вимог до ІС.

Результати взаємного відображення елементів ДК, EER та ER, які описують ІС на рівні знань, представлено у таблиці 1.

Таблиця 1

Результати взаємного відображення елементів діаграм класів, EER та ER

| Елемент ДК | Елемент EER | Елемент ER |
|------------------------|--|---|
| Змінна (назва, тип) | Атрибут (назва, тип) | Атрибут (назва, тип) |
| Назва класу | Назва типу сутності | Назва типу сутності |
| Операція | Атрибут <i>A</i> [назва_операції] Підатрибут <i>B</i> [опис_операції], <i>B</i> є підатрибутом <i>A</i> | Атрибут <i>A</i> [назва_операції] Атрибут <i>B</i> [назва_операції] [опис_операції] |
| Пакет | Домен | Домен |
| Наслідування | Наслідування | Базовий клас і дочірні класи стають сутностями, пов'язаними між собою зв'язком типу «асоціація» (або асоціативною сутністю) |

Для позначення й подальшого формалізованого опису такої розширеної сукупності вимог до ІС пропонується використовувати існуюче поняття «універсум». Це поняття найчастіше позначає фіксовану систему (або системи) об'єктів, до яких належать твердження (висловлення) якої-небудь теорії. Пропонується така інтерпретація поняття «універсум»: «універсум – це «множина всіх можливих систем», з яких лише одна – досліджувана система – реальна, а всі інші (у тому числі й проєктовану ІС) можливо осмислити тільки логічним шляхом, тобто несуперечливим чином описуючи можливі факти або зв'язки досліджуваної системи». Розгортаючи дану інтерпретацію, поняття «універсум» у теорії й практиці створення, впровадження, експлуатації й модернізації ІС можна сформулювати таким чином: «універсум – сукупність даних, інформації та знань про досліджувану систему, об'єкт або процес, як відомих, так і невідомих дослідникам, у розпорядженні якого є скінченна множина методів отримання й обробки цих даних, інформації та знань». Дане означення породжує цілий

ряд висновків, з яких для формалізованого опису вимог до ІС і процесів проектування архітектури важливі, насамперед [12]:

- точність опису досліджуваної системи, об'єкта або процесу наближується до максимуму в тому випадку, якщо обсяг невідомих дослідникові даних, інформації та знань про цю систему, об'єкт або процес наближується до мінімуму;

- дослідник завжди має припускати, що сукупність невідомих йому даних, інформації та знань про досліджувану систему, об'єкт або процес не є порожньою;

- для практичного застосування Споживачем універсум має мати не максимальну точність, а таку, яка дозволить на підставі сукупності відомих даних, інформації та знань ухвалювати рішення з бажаними для Споживача універсуму характеристиками ефективності і якості.

6. Обговорення отриманих результатів

Одній вимозі фізично відповідають набір класів ПЗ, таблиць БД і мережа фреймів, що описують вимогу у вигляді знань. Отже, фізична реалізація вимоги вимагає її декомпозиції на окремі підвимоги, що фізично подається у вигляді окремих класів, таблиць тощо.

Відомості про особливості механізмів і алгоритмів обробки даних, регламентованих вимогою до ІС, можуть бути витягнуті з візуальних структурних і об'єктних моделей, що описують вимоги на рівні інформації. Їхній опис у вигляді знань може бути реалізований із застосуванням каузальних сценаріїв. У такому випадку інформаційні одиниці мережі фреймів, яку описує вимога, можуть бути доповнені сигнатурами приєднаних процедур фреймів, описаними у вигляді сценаріїв.

Наявність ієрархічних зв'язків між об'єктами ПрО обумовлює утворення фреймами дерев, які додатково можуть бути з'єднані між собою горизонтальними зв'язками. Отримана в такий спосіб мережа фреймів, яка описує вимоги, матиме вигляд «низькорослого лісу» або «чагарнику», у відповідності зі зв'язками об'єктів ПрО, для якої проектується ІС.

Під час аналізу ПрО й формування вимог до ІС здійснюється декомпозиція цих вимог на підвимоги й виділення окремих об'єктів ПрО, представленнями яких має оперувати ІС. Реалізовані вимоги перед виконанням робіт зі створення елементів ІС, як правило, групуються навколо одного кореневого поняття ПрО й можуть включати дерева понять, що деталізують базове дерево й пов'язані з ним горизонтальними зв'язками. При цьому вимога може вважатися новою, а не уточнюючою існуючою, якщо в мережі фреймів вона є кореневим вузлом нового дерева об'єктів ПрО. Таке дерево деталізується під час подальшого аналізу ПрО й формування вимог, що уточнюють це дерево. У ПЗ це відображається ієрархією класів-спадкоємців одного базового (часто абстрактного) класу, в ІЗ – схемами даних типу «зірка» або «сніжинка», у яких консольні таблиці й таблиці вимірів згруповані навколо базової таблиці фактів, яку вони розширюють і уточнюють. Для зручності роботи з такими наборами таблиць вони можуть поєднуватися в одну схему даних (яка належить одному користувачеві).

Характерною властивістю подібної схеми даних є можливість денормалізації й об'єднання всіх таблиць схеми в одне універсальне відношення, що описує базове поняття ПрО, відповідне до таблиці фактів вітрини даних. Наприклад, таке універсальне відношення може бути створене у вигляді представлення (віртуальної таблиці). Застосування універсальних відношень дозволяє масштабувати моделювання вимог до ІС, оперуючи реалізованими вимогами на рівні підвимог, окремих таблиць БД і класів ПЗ, окремих вимог до ІС, фізично представлених ІТ-сервісів або їх компонентів.

Для формування мережі фреймів, відображуваних згодом в елементи ПЗ та ІЗ ІТ-сервісів ІС, потрібне застосування прийомів класичного об'єктно-орієнтованого аналізу й проектування (аналогічно сучасним методологіям розробки ПЗ). Сформована в такий спосіб онтологія ПрО придатна для двостороннього відображення представлення вимог на рівні знань в описи ПЗ та ІЗ ІС.

Застосування об'єктно-орієнтованого аналізу дозволяє виявити об'єкти ПрО та зв'язки між ними, у тому числі сформувати ієрархії класів об'єктів.

Об'єктно-орієнтований аналіз дозволяє формувати ієрархії класів ПрО необмежено великої довжини. Формально в сучасних мовах програмування також немає обмежень на кількість рівнів спадкування класів об'єктів. Однак на практиці при проектуванні й візуальному моделюванні ПЗ не рекомендується застосовувати ієрархії спадкування, що мають більш ніж 5-6 рівнів, бо це утрудняє сприйняття діаграм класів людиною. Крім того, необхідність відображення мережі фреймів у БД ІС також накладає обмеження у вигляді складності реалізації багаторівневих ієрархій у вигляді наборів таблиць БД.

Внаслідок цього найскладнішою задачею при формуванні онтології ПрО у вигляді мережі фреймів є вибір фреймів, що є кореневими елементами дерев онтології, і ухвалення рішення про спадкування фреймів від існуючих або про формування нового кореня дерева фреймів. При роботі з деревами вимог з метою порівняння онтологій різних ПрО й ухвалення рішення про розширення й повторне використання вимоги в створюваній ІС у рамках відповідної ІТ-послуги найскладнішою є задача оцінки подібності об'єктів різних ПрО.

Як один з результатів застосування розробленої технології пропонується розглядати здійснення взаємо-однозначного відображення мережі фреймів, яка описує онтологію ПрО проектованої ІС у вигляді патернів вимог до ІС, в елементи ПЗ та ІЗ ІС для наступного виконання проектних робіт з фізичної реалізації системи. Теорія фреймів є основою сучасних об'єктно-орієнтованих мов програмування, і тому відображення патернів вимог, описаних у вигляді мережі фреймів, у сутності БД і класи ПЗ розроблюваної ІС зводиться до розв'язання задачі об'єктно-реляційного відображення.

Оскільки мережа фреймів і діаграма класів засновані на застосуванні загального математичного апарата, однакових типів зв'язків і структур даних, відображення мережі фреймів онтології ПрО в діаграму класів ПЗ здійснюється природним чином: фрейми відображаються в класи з такими самими назвами й наборами атрибутів і методів, зв'язки між фреймами – в однойменні їм зв'язки генералізації й асоціації класів. Стосовно ІЗ ІС основна складність полягає у відсутності в найпоширенішій у даний момент реляційної моделі даних ієрархічних зв'язків і механізму спадкування. Для реалізації задачі об'єктно-реляційного відображення зараз існує кілька типових підходів, найпоширенішими з яких є відображення кожного дочірнього класу (включаючи наслідувані атрибути) в окремі таблиці БД, а також емуляція механізму спадкування за рахунок застосування ідентифікуючих зв'язків між сильними (батьківськими) і слабкими (дочірніми) сутностями.

Внаслідок цього саме необхідність відображення ієрархії фреймів у реляційні схеми вітрин даних обумовлює необхідність застосування правила класифікації фреймів як корневих самостійних понять і як результат – представлення мережі фреймів у вигляді «низькорослого лісу».

Оскільки ані цілісність сутностей, ані посилальна цілісність повною мірою не забезпечує розв'язання даної задачі, пропонується використання тригерів. Тригери реалізують складні

обмеження цілісності даних, які нереалізовані описовими обмеженнями, що встановлюються при створенні таблиць.

Можна виділити чотири основні типи тригерів, що забезпечують цілісність даних в узагальненій й уточнюючих таблицях:

а) тригери, що забезпечують сюр'єктивне відображення даних у двох таблицях (обов'язкова наявність відповідних записів у головній і залежній таблицях, зв'язок «1:1», оскільки наявність ідентифікуючого зв'язку забезпечує тільки наявність зв'язку «1:0..1»);

б) тригери, що забезпечують унікальність значень комбінацій атрибутів, розміщених у різних таблицях (аналог унікального ключа в одній таблиці);

в) тригери, що забезпечують контроль внесення даних у дочірні таблиці, коли сильна сутність пов'язана з декількома слабкими, а дані в слабких сутностях не мають перетинатися між собою (наприклад, сильна сутність «люди», слабкі – «викладач» і «студент»);

г) тригери, що забезпечують заповнення (аналог *not null option*) необов'язкових полів сильних сутностей, якщо це потрібно для залежних від них слабких сутностей (наприклад, для абстрактної сильної сутності «людей» поле «контактний телефон» може бути необов'язковим, а для наслідуваної від неї сутності «співробітник» це саме поле, фізично розташоване в головній таблиці, є обов'язковим).

Основні етапи побудови мережі фреймів під час аналізу вимог і ПрО створюваної ІС, а також відповідні їм етапи відображення й реалізації вимог в описи елементів ПЗ та ІЗ ІС наведено у [11].

Оскільки EER-діаграма схожа з діаграмою класів та ER-діаграмою у математиці, наведеній у роботах [18-19], то мережа фреймів працює і з EER-діаграмою.

Вимога на рівні знань може бути описана у вигляді фрейму *f*, який може бути розділений на окремі структури даних, описувані інтерфейсами фрейму. Як і в об'єктно-орієнтованому програмуванні, кілька різних фреймів, що описують різні вимоги, можуть реалізовувати той самий інтерфейс, тобто мати спільні структурні елементи. Застосування інтерфейсів фреймів дозволяє застосовувати в різних фреймах єдині описи типової структури даних. При цьому клас або фрейм може одночасно реалізовувати кілька інтерфейсів. Крім того, більшість сучасних мов об'єктно-орієнтованого програмування не надають можливість реалізації множинного спадкування, реалізація якого викликає неоднозначності (наприклад, якщо в наслідуваних класах існують однойменні атрибути). Застосування інтерфейсів дозволяє здійснювати не тільки вертикальний аналіз ієрархічної структури класів або фреймів, але й горизонтальний аналіз структурних елементів окремих класів, що компенсує відсутність механізмів множинного спадкування. Комбінації двох зазначених видів аналізу структур фреймів дозволяє здійснювати відображення фреймів, що описують вимоги, у різні елементи ПЗ.

Головним недоліком отриманих результатів дослідження слід вважати обмеженість застосування отриманих результатів через спирання запропонованого підходу на фреймову модель (що вимагає додаткового аналізу придатності описаного у дослідженні до певних ПрО).

7 Висновки

Проведене дослідження базується на припущенні, що функціональні вимоги, здебільшого, впливають на такі компоненти ІС, як ІЗ та ПЗ. Зазвичай, для опису архітектур ІЗ та ПЗ використовують ER-діаграму та діаграму класів, відповідно. Але під час розробки у деяких випадках виникає проблема сумісності ІЗ та ПЗ через те, що діаграми виконані лише з таких точок зору, для яких були створені візуальні описи. Вирішення цієї проблеми полягає

в уніфікації цих описів, для чого слід застосувати формальний апарат, що використовує фреймову модель для відображення опису функціональних вимог на рівні знань з використанням діаграми класів та ER-діаграми.

У статті досліджені три візуальні нотації функціональних вимог, що описують архітектуру ІЗ (ER-діаграма та EER-діаграма) та ПЗ (ДК). За результатами дослідження спільних елементів діаграм та запропонованої мережі фреймів виявлено, що мережа фреймів може відображати опис функціональних вимог на рівні знань у EER-діаграму. Bazуючись на отриманому результаті, запропоновано формальний опис взаємно-однозначних відображень спільних елементів ДК, ER-діаграми та EER-діаграми. Отримані результати дозволяють визначити EER-діаграму як таку ВМ, яка буде використовуватися для формалізації процесів візуального відображення описів функціональних вимог до ІС на рівні знань, і запропонувати механізм трансформації цієї ВМ у ER-діаграму та ДК, які, відповідно, описують архітектури ІЗ та ПЗ ІС.

Перелік посилань

1. ДСТУ ISO/IEC/IEEE 15288:2016 Інженерія систем і програмного забезпечення. Процеси життєвого циклу систем (ISO/IEC/IEEE 15288:2015, IDT). [Чинний від 2018-10-01]. Вид. офіц. Київ: ДП «УкрНДНЦ», 2018. 80 с.
2. Настанова до зводу знань з управління проєктами (Настанова РМВОК) Сьоме видання. Project Management Institute, PMI. 2021. 370 с.
3. SSADM Version 4 Reference Manual. Oxford: NCC Blackwell, 1990. 1400 p.
4. Leffingwell D., Widrig D. Managing software requirements: a unified approach/ Dean Leffingwell, Don Widrig. Boston: Addison-Wesley Longman Publishing, 1999. 491 p.
5. Wiegers Karl E., Beatty J. Software Requirements, 3rd Edition /Karl E. Wiegers, Joy Beatty. Redmond, WA: Microsoft Press, 2013. 673 p.
6. Hull E., Jackson K., Dick J. Requirements Engineering / Elizabeth Hull, Ken Jackson, Jeremy Dick. Springer London, 2011. 207 p.
7. Watson M. Managing Smaller Projects: A Practical Approach / M. Watson. Multi-Media Publications Inc., 2006. 240 p.
8. Capturing Architectural Requirements. URL: <https://www.ibm.com/developerworks/rational/library/4706-pdf.pdf> (дата звернення: 19.10.2023)
9. Babok: A Guide to the Business Analysis Body of Knowledge, v3. International Institute of Business Analysis, 2015. 512 p.
10. Евланов М.В. Модели, методы и информационная технология разработки архитектуры сложных информационных систем на основе функциональных требований: дис. ... докт. техн. наук: 05.13.06. Харьков, 2017. 429 с.
11. Левыкин В.М. Паттерны проектирования требований к информационной системе: моделирование и применение / В.М. Левыкин, М.В. Евланов, М.А. Керносов: монография. Харьков: ООО «Компанія СМІТ», 2014. 320 с.
12. Евланов М.В. Концепция представления требований к информационной системе / М.В. Евланов // *Вісник національного технічного університету «ХПІ». Збірник наукових праць. Серія «Нові рішення в сучасних технологіях»*. 2012. № 68 (974). С. 32-40.
13. Davis Alan M. 201 Principles of Software Development / Alan M. Davis. New York: McGraw-Hill, 1995. 240 p.
14. Fowler M. UML Distilled. A Brief Guide to the Standard Object Modeling Language 3rd Edition / Martin Fowler. Addison-Wesley Professional, 2004. 175 p.
15. He X., Ma Z., Shao W., Li G. A metamodel for the notation of graphical modeling languages. /He, Xiao; Ma, Zhiyi; Shao, Weizhong; Li, Ge. *31st Annual International Computer Software and Applications Conference (COMPSAC 2007)*, Beijing, China, 2007. Vol. 1. P 219-224.
16. Teorey T. A logical design methodology for relational databases using the extended entity–relationship model. / Toby J. Teorey, Dongqing Yang, James P. Fry // *ACM Computing Surveys*. 1986. Vol. 18, № 2. P. 197–222.
17. Chapter 8. Enhanced EER Model. URL: <http://comet.lehman.cuny.edu/jung/cmp420758/chapter8.pdf> (дата звернення: 12.10.2023).

18. Левыкин В.М. Параллельное проектирование информационного и программного комплексов информационной системы / В.М. Левыкин, М.В. Евланов, В.С. Сугробов // *Радиотехника*. 2006. Вып. 146. С. 89-98.

19. The Enhanced Entity-Relationship Model. URL: <https://www.is.informatik.uni-kiel.de/~thalheim/HERM/HERMindetail.pdf> (дата звернення: 12.10.2023).

Надійшла до редколегії 27.10.2023 р.

Кожанов Адріан Євгенійович, аспірант кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: adrian.kozhanov@nure.ua. ORCID: <https://orcid.org/0009-0002-3586-6886/>

РЕФЕРАТИ

УДК 004.65:005.33

Оцінювання витрат на проведення рефакторингу бази даних інформаційної системи, яка експлуатується / М.В. Євланов, І.О. Юр'єв, Д.О. Мірошніченко // АСУ і прилади автоматики. 2023. № 179. С. 4-16.

Об'єктом дослідження є робота з прийняття рішення щодо доцільності проведення рефакторингу бази даних інформаційної системи, яка експлуатується. Визначено, що основні рекомендації з формування та прийняття подібних рішень базуються на використанні переважно інтуїції та досвіду особи, яка приймає рішення. Сучасні дослідження в галузі рефакторингу баз даних присвячені, головним чином, аналізу та вдосконаленню прикладних методик рефакторингу. Проблема визначення доцільності рефакторингу бази даних та, зокрема, визначення можливості здійснення рефакторингу в умовах діючих обмежень ІТ-проєкту залишається практично невирішеною. Тому проведення досліджень з вирішення цієї проблеми є актуальним з теоретичної і прикладної точок зору.

Як основи моделі оцінювання витрат було обрано моделі СОСОМО II. Визначені особливості використання цих моделей під час розрахунку трудовитрат для ІТ-проєктів модифікації або розвитку програмних систем. Встановлені основні SQL-команди, які найчастіше використовуються під час рефакторингу бази даних. Розроблено моделі оцінювання кількості рядків вихідного коду кожної з цих SQL-команд. Вдосконалено модель оцінювання кількості рядків вихідного коду для ІТ-проєкту модифікації або розвитку систем, окремим випадком якого пропонується вважати рефакторинг бази даних.

Проведено експериментальну перевірку отриманих наукових результатів. Для такої перевірки було обрано базу даних інформаційно-облікової системи однієї з організацій, яка виконує функції Інтернет-провайдера. Для цієї системи було сформульовано два запити на зміну, за якими слід було проводити рефакторинг бази даних. Описана послідовність дій з оцінювання трудовитрат і витрат часу на рефакторинг бази даних за визначеними запитами на зміну. Отримані результати оцінювання дозволяють прийняти рішення про можливість проведення рефакторингу бази даних інформаційно-облікової системи.

Ключові слова: рефакторинг, база даних, СОСОМО II, SQL, запит на зміну.

Табл. 0. Іл. 9. Бібліогр.: 17 назв.

UDC 004.65:005.33

Estimating the costs of refactoring the database of the information system that is being operated / M.V. Yevlanov, I.O. Yur'ev, D.O. Miroshnichenko // Management Information System and Devices. 2023. № 179. P. 4-16.

The object of research is the decision-making work on the expediency of refactoring the database of the information system that is being operated. It was determined that the main recommendations for the formation and adoption of such decisions are based on the use of the intuition and experience of the person making the decision. Modern research in the field of database refactoring is mainly devoted to the analysis and improvement of applied refactoring techniques. The problem of determining the expediency of database refactoring and, in particular, determining the possibility of refactoring in the conditions of the current limitations of the IT project remains practically unsolved. Therefore, conducting research to solve this problem is relevant from a theoretical and applied point of view.

The COCOMO II model was chosen as the basis of the cost estimation model. The specifics of using these models when calculating labor costs for IT projects of modification or development of software systems are determined. Basic SQL commands that are most often used during database refactoring are installed. Models for estimating the number of lines of source code of each of these SQL commands have been developed. The model for estimating the number of lines of source code for an IT project of system modification or development has been improved, a special case of which is proposed to be database refactoring.

Experimental verification of the obtained scientific results was carried out. The database of the information and accounting system of the organization, which performs the functions of an Internet provider, was selected for such verification. For this system, two requests for changes were formulated for which the database should be refactored. The sequence of actions for estimating labor costs and time spent on database refactoring based on specified change requests is described. The obtained evaluation results allow to make a decision about the possibility of refactoring the database of the information and accounting system.

Keywords: refactoring, database, COCOMO II, SQL, request for change.

Tab. 0. Fig. 9. Ref.: 17 items.

УДК 681.5:004.94

Експертна система обробки інформації для прийняття рішень в світлофорній системі з адаптивним умовним пріоритетом трамваєві./ М. Литвиненко, Л. Ребезюк // АСУ і прилади автоматики. 2023. № 179. С. 17-24.

Запропонована експертна система обробки інформації в системі світлофорів мультимодального перехрестя з трьома типами учасників дорожнього руху: маршрутний транспорт (трамвай), пішоходи і загальний транспорт. Основою системи, яка враховує динамічні характеристики трамвая від моменту його виявлення вхідним датчиком перехрестя, є набір нечітких законів прийняття рішень щодо адаптації часових параметрів світлофорного плану на основі сумісності сигналів напрямків руху та значень важливості запитів від транспорту. Для сигналів, що регулюють потоки загального транспорту рішенням є подовження чи припинення дії дозвільного сигналу для певної групи напрямків руху, що керуються цим сигналом. Рішення щодо трамвайних сигналів, залежно від наближеності запланованого дозвільного сигналу і важливості пов'язаного запиту, можуть впливати як на часові параметри безпосереднього світлофорного плану, так і змінювати загальну послідовність сигналів та/або їхню тривалість. Умовний пріоритет полягає в урахуванні можливих конфліктів між рішеннями від сигналів маршрутного транспорту або від маршрутного і загального транспорту, для розв'язання яких в роботі запропоновані відповідні правила продукції. Світлофорний план змінюється ітеративно в ациклічний спосіб з урахуванням ступеня впевненості в рішенні, отриманого дворівневою моделлю нечіткого висновування за алгоритмом Мамдані, другий рівень якої відповідає за визначення послідовності сигналів, а перший – за їхні часові параметри. Модель обробки інформації реалізована та апробована за допомогою засобу моделювання міської мобільності SUMO для помірного та насиченого транспортних попитів на нетривіальному перехресті з синтетичним транспортним попитом. За результатами моделювання продемонстрована доцільність застосування системи в умовах насиченого транспортного попиту, що дозволяє зменшити негативний вплив на потоки загального транспорту без значного зростання часових втрат пасажирів трамваїв у порівнянні з безумовним способом реалізації світлофорного пріоритету трамваю.

Ключові слова: експертна система, прийняття рішень, нечітка логіка, конфлікт, трафік, світлофор.

Табл. 6. Іл. 9. Бібліогр.: 28 назв.

UDC 681.5:004.94

Expert information processing system for traffic light system decision-making with adaptive conditional tram priority/ M. Lytvynenko, L. Rebezyuk // Management Information System and Devices. 2023. № 179. P. 17-24.

The expert information processing system was proposed for a traffic signal system at a multimodal intersection with three types of traffic users: public transport (tram), pedestrians, and general traffic. The system considers the dynamics characteristics of the tram after it is detected by the entry detector of the intersection and is based on a set of fuzzy decision-making rules for decision-making on the time parameters adjustment of the traffic signal plan based on the compatibility of traffic direction signals and the importance of traffic requests. For general traffic signals, the solution is to extend or terminate the permissive signal for a certain group of traffic directions controlled by this signal. Decisions for tram signals, depending on the proximity of the planned

permissive signal and the importance of the associated request, can affect both the time parameters of the immediate traffic signal plan and change the overall signal sequence and/or their duration. The conditional priority involves taking into account possible conflicts between the decisions from the public transport signals or from public transport and general traffic signals, which are handled by the corresponding production rules proposed in the paper. The traffic light plan is changed iteratively in an acyclic manner, considering the degree of confidence of the decision obtained by a two-level fuzzy inference model based on the Mamdani algorithm, the second level of which is responsible for determining the signal sequence and the first level for their time parameters. The information processing model is implemented and validated using SUMO urban mobility simulation tool for moderate and saturated transport demands for a non-trivial intersection with synthetic traffic demand. The experimental results demonstrate the applicability of the system in conditions of a saturated transport demand, allowing to reduce the negative impact on the general traffic flows without a significant increase in the time losses of tram passengers compared to the adaptive unconditional way of implementing tram traffic light priority.

Keywords: expert system, decision making, fuzzy logic, conflict, traffic, traffic light.

Tab. 6. Fig. 9 Ref.: 28 items.

УДК 004.9

Модернізація спеціалізованих інформаційних систем служби крові / А.В. Міхнова, Д.К. Міхнов, К.С. Чиркова // АСУ і прилади автоматики. 2023. № 179. С. 25-31.

Об'єктом дослідження є математичний апарат пошуку оптимального варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем служби крові, який дозволяє визначити шляхи підвищення ефективності діяльності закладів служби крові в межах визначених витратних ресурсів за рахунок зменшення кількості ручних операцій введення даних бізнес-процесів за допомогою відповідних організаційно-технічних компонентів спеціалізованих медичних інформаційних систем.

Метою розробки математичного апарату пошуку варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем служби крові є формалізоване представлення задачі пошуку оптимального варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем служби крові з урахуванням складових бізнес процесів, складових спеціалізованих медичних інформаційних систем, показників діяльності закладів служби крові, часткових та узагальненого показників функціонування спеціалізованих медичних інформаційних систем, витратних ресурсів на реалізацію варіанту проектного рішення модернізації спеціалізованих медичних інформаційних систем або окремого організаційно-технічного компонента.

Для розв'язання поставленої задачі використовувався процесно-орієнтований підхід організації спеціалізованої медичної інформаційної системи з декомпозицією на організаційно-технічні компоненти, на основі чого запропоновано формалізоване представлення оптимального варіанту проектного рішення модернізації спеціалізованої медичної інформаційної системи за рахунок підвищення ступеня автоматизації відповідних організаційно-технічних компонентів, що забезпечує максимальний інтегрований показник функціонування спеціалізованої медичної інформаційної системи, що корелює з фактичними ключовими соціально значущими показниками діяльності центра служби крові. Спроектовано ER-модель модернізації спеціалізованої медичної інформаційної системи, яка дозволяє графічно представити структуру пошуку оптимального варіанту проектного рішення модернізації спеціалізованої медичної інформаційної системи.

Результати дослідження застосовано для пошуку оптимального варіанту проектного рішення модернізації спеціалізованої медичної інформаційної системи КНП ХОР «Обласний центр служби крові», що дозволило покращити ключові соціально значущі показники діяльності закладу за рахунок визначення оптимального ступеня автоматизації введення даних на кожному організаційно-технічному компоненті спеціалізованої медичної інформаційної системи в межах існуючих витратних ресурсів на модернізацію.

Ключові слова: модернізація, спеціалізована, система, дані, компонент, функціональність,

ефективність, медичні, показник, автоматизація

Табл. 0. Лл. 3. Бібліогр.: 12 назв.

UDC 004.9

Modernization blood service's specialized information systems / A. Mikhnova, D. Mikhnov, K. Chyrkova // Management Information System and Devices. 2023. № 179. P. 25-31.

The object of the research is a mathematical apparatus for searching for the optimal variant of the project solution for the modernization blood service's specialized medical information systems, which allows determining ways to increase the efficiency of blood services attached to limits of defined expendable resources by reducing the number of manual data entry operations during business processes with the help of appropriate organizational and technical components of specialized medical information systems.

The purpose of the development of a mathematical apparatus for searching for a variant of a project solution for the modernization blood service's specialized medical information systems is a formalized presentation of the problem finding an optimal variant of a project solution for the modernization blood service's specialized medical information systems, taking into account the components of business processes, components of specialized medical information systems, performance indicators of blood services, partial and generalized indicators of the functioning of specialized medical information systems, expendable resources for the implementation of a variant of the project solution for the modernization specialized medical information systems or a separate organizational and technical component.

To solve the problem, a process-oriented approach to the organization of a specialized medical information system with a decomposition into organizational and technical components was used, on the basis of which a formalized presentation of the optimal version of the project solution for the modernization a specialized medical information system by increasing the degree of automation of the relevant organizational and technical components was proposed, that provides the maximum integrated indicator of the functioning of the specialized medical information system, which correlates with the actual key socially significant indicators of the activity of the blood service. The ER model of the modernization of the specialized medical information system was designed, which allows to graphically present the structure of the search for the optimal variant of the project solution for the modernization of the specialized medical information system.

The results of the study were used to find the optimal project solution for the modernization of the specialized medical information system of the CNE KHRC "Regional Blood Service Center", which made it possible to improve the key socially significant indicators of the blood service 's activity by determining the optimal data entry's degree of automation on each organizational and technical component of the specialized medical information system considering the existing expendable resources for modernization.

Keywords: modernization, specialized, system, data, component, functionality, efficiency, medical, indicator, automation

Tab. 0. Fig. 3. Ref.: 12 items.

УДК 004.67

Рекомендаційна система на основі компактною гібридною моделі користувача / П.Е. Ситнікова, М.О. Гребенюк // АСУ і прилади автоматики. 2023. № 179. С. 32-42.

Об'єктом дослідження є процес рекомендації об'єктів користувачу системи та поліпшення результатів рекомендації. Розглянуто принцип роботи рекомендаційної системи та етапи створення рекомендацій. Описані різні методи фільтрації інформації для генерації персоналізованих рекомендацій. Виявлено переваги та недоліки кожного із методів. Зроблено висновки про те, що з урахуванням як теоретичних, так і практичних аспектів, важливим є проведення досліджень з метою виправлення обмежень, характерних для різних методів фільтрації.

Запропоновано компактною гібридну модель користувача для методу спільної фільтрації інформації. Ця модель долає обмеження, які часто виникають при використанні традиційних підходів, і дозволяє ефективніше генерувати персоналізовані рекомендації. Для наочного прикладу моделі, що

пропонується, було взято рекомендаційну систему кінофільмів. Ця модель поєднує оцінки користувачів з описами вмісту об'єктів і використовує поняття міри цікавості жанру, що було отримано шляхом виведення формул для обчислення гібридної ознаки, тобто показника того, наскільки користувач зацікавлений у тому чи іншому жанрі. Це полегшує формування набору близьких однодумців для активного користувача. Запропонований підхід спрямований на забезпечення високої точності рекомендацій в системі, дозволяє знизити вимоги до обчислювальних ресурсів та ефективно використовувати інформацію про вміст об'єктів. Гібридна модель дозволяє поєднувати переваги спільної фільтрації на основі пам'яті та рекомендаційних систем на основі моделі, що забезпечує точність та масштабованість.

Наведено декілька прикладів з результатами обчислень. Зроблено висновок стосовно потенційного покращення якості рекомендацій з урахуванням можливостей розробленого алгоритму.

Ключові слова: рекомендаційна система, спільна фільтрація, компактна модель, гібридна ознака, міра цікавості жанру.

Табл. 8. Іл. 2. Бібліогр.: 11 назв.

UDC 004.67

Recommendation system based on a compact hybrid user model / P.E. Sytnikova, M.O. Hrebeniuk // Management Information System and Devices. 2023. № 179. P. 32-42.

The object of research is the process of recommending items to the system user and improving the recommendation results. The principle of the recommendation system and the stages of creating recommendations are considered. The article describes various methods of filtering information to generate personalized recommendations. The advantages and disadvantages of each of the methods are revealed. Therefore, taking into account both theoretical and practical aspects, it is important to conduct research in order to correct the limitations characteristic of various filtering methods.

A compact hybrid user model has been proposed for the collaborative filtering method of information. This model overcomes the limitations often encountered when using traditional approaches and allows for more efficient generation of personalized recommendations. For a visual example of the described approach, the recommender system of motion pictures was taken. This model combines user ratings with descriptions of the content of objects and uses the concept of genre interest rate, which was obtained by deriving formulas for calculating a hybrid feature, that is, an indicator of how much the user is interested in a particular genre. This facilitates the formation of a set of close associates for an active user. The proposed approach is aimed at ensuring high accuracy of recommendations in the system, reducing the requirements for computing resources and effectively using information about the content of objects. The hybrid model combines the benefits of memory-based collaborative filtering and model-based recommender systems, providing accuracy and scalability.

Several examples with calculation results are given. A conclusion was drawn regarding the potential improvement of the quality of recommendations, taking into account the capabilities of the developed algorithm.

Keywords: recommendation system, collaborative filtering, compact model, hybrid feature, genre interestingness indicator.

Tab. 8. Fig. 2. Ref.: 11 items.

УДК 004.627

Огляд існуючих методів зменшення розмірності та класифікації великих вибірок даних / І.Г. Перова, Н.С. Мірошніченко // АСУ і прилади автоматики. 2023. № 179. С. 42-50.

На сьогоднішній день аналіз великої кількості даних займає ключове місце у виявленні прихованих закономірностей і тенденцій, які не відразу видно з узагальнених даних. Так як дані мають складні структури та великі розміри, науковці працюють над розробкою методів зменшення розмірності великих вибірок даних. Зменшення розмірності як етап попередньої обробки машинного навчання є

ефективним у видаленні нерелевантних і надлишкових даних, підвищення точності навчання та покращення зрозумілості результату, за допомогою візуалізації розмірності. А також, процес зменшення кількості аналізованих випадкових величин шляхом отримання набору основних змінних. Проте зменшення розмірності має недолік, пов'язаний з втратою даних. Дуже важливо зменшити розмірність набору даних без втрати будь-якої інформації з цих наборів даних. У статті розглянуто існуючі методи зменшення розмірності великих вибірок даних, а саме: аналіз головних компонент (Principal Component Analysis), лінійний дискримінантний аналіз (Linear Discriminant Analysis), аналіз головних компонент ядра (Kernel Principal Component Analysis), багатовимірне масштабування (MDS), t-розподільного стохастичного вбудовування сусідів (t-SNE) та аналіз незалежних компонент (Independent Component Analysis).

Кожен з методів має свої переваги та недоліки, для вибору найбільш оптимального методу зменшення розмірності великих вибірок даних було проведено їх порівняльний аналіз. На наборі даних ініціативи з нейровізуалізації хвороби Альцгеймера та на наборі даних про щитоподібну залозу було протестовано кожен з розглянутих методів.

Результати порівняльного аналізу методів було представлено у вигляді графічних зображень.

Ключові слова: вибірка; дані; розмірність; аналіз; дослідження; класифікація; візуалізація.

Табл. 0. Іл. 02. Бібліогр.: 12 найм.

UDC 004.627

A review of existing methods for dimensionality reduction and classification of large data sets / I.G.

Perova, N.S. Miroshnychenko // Management Information System and Devices. 2023. № 179. P. 42-50.

Today, big data analysis occupies a key place in finding hidden patterns and trends that are not immediately visible from generalized data. Since data has complex structures and large sizes, scientists are working on developing methods to reduce the dimensionality of large data sets. Dimensionality reduction as a pre-processing step in machine learning is effective in removing irrelevant and redundant data, increasing training accuracy, and improving the comprehensibility of the result by visualizing the dimensionality. It is also a process of reducing the number of analyzed random variables by obtaining a set of basic variables. However, dimensionality reduction has the disadvantage of losing data. It is very important to reduce the dimensionality of a dataset without losing any information from these datasets. This article discusses the existing methods for dimensionality reduction of large data sets, namely Principal Component Analysis, Linear Discriminant Analysis, Kernel Principal Component Analysis, Multidimensional Scaling, t-distributed stochastic neighborhood embedding and Independent Component Analysis.

Each of the methods has its advantages and disadvantages, and a comparative analysis was performed to select the most optimal method for reducing the dimensionality of large data sets. Each of the considered methods was tested on the Alzheimer's Disease Neuroimaging Initiative dataset and the thyroid dataset.

The results of the comparative analysis of the methods were presented in the form of graphic images.

Keywords sample; data; dimensionality; analysis; research; classification; visualization.

Tab. 0. Fig. 02. Ref.: 12 items.

УДК 004.89:61

Дослідження ансамблювання моделей machine learning в медичній діагностиці / І.Ю.

Панфьорова, А.С. Буцька // АСУ і прилади автоматики. 2023. № 179. С. 50-57.

Розглядається процес використання моделей машинного навчання (machine learning – ML) для постановки медичного діагнозу, що має велике значення в сучасній медицині. Стаття зосереджується на використанні ансамблювання моделей ML, в якому беруть участь згорткові нейронні мережі (CNN), моделі машини векторів підтримки (SVM) та рекурентні мережі з довготривалою короткостроковою пам'яттю (LSTM). Ефективність цих методів порівнюється як індивідуально так і у ансамблевому підході у контексті покращення точності медичної діагностики.

Незважаючи на те, що численні дослідження, які були розглянуті у статті, поєднували в собі кілька

моделей одночасно для покращення точності прогнозування, тема дослідження ансамблювання моделей machine learning в медичній діагностиці лишається актуальною. У попередніх дослідженнях було запропоновано широкий спектр моделей ML для аналізу електронних медичних записів. Більшість робіт використовували k-середні, KNN і SVM.

Результати дослідження свідчать, що використання ансамблевої моделі, яка комбінує вище зазначені методи ML, призводить до підвищення точності діагнозу. Навіть якщо окремі методи ML можуть бути ефективними в обраних областях, їх комбінування в інтегровану структуру дозволяє уникнути обмежень, які можуть виникнути при використанні лише одного методу.

Обговорюються переваги та недоліки обраної ансамблевої моделі. Основна перевага полягає в підвищенні точності медичного діагнозу, що може призвести до раннього виявлення захворювань і, отже, покращити результати лікування. З іншого боку, недоліками можуть бути складність розробки та налаштування ансамблевого підходу, а також вимоги до обчислювальних ресурсів.

Необхідно зазначити, що отримані результати свідчать про те, що поєднання різних моделей може істотно покращити якість діагностування захворювань. Це важливий внесок у розвиток медичної сфери, оскільки може сприяти ранньому виявленню та ефективному лікуванню різних патологій.

Ключові слова: автоматизація; аналіз; ансамбль; діагностика; дослідження; прогнозування, точність.

Табл. 1. Іл. 3. Бібліогр.: 14 назв.

UDC 004.89:61

Study of the ensemble of machine learning models in medical diagnostics / I. Panforova, A. Butska // Management Information System and Devices. 2023. № 179. P. 50-57.

The research discusses the use of machine learning models for medical diagnosis, which is of great significance in modern medicine. The focus of the article is on the use of ensemble models in which convolutional neural networks (CNN), models for classification and regression tasks (SVM), and long short-term memory recurrent networks (LSTM) are involved. The effectiveness of these methods is compared both individually and in an ensemble approach in the context of improving the accuracy of medical diagnosis.

Despite numerous studies discussed in the article combining multiple models simultaneously to enhance prediction accuracy, the topic of ensemble modeling in machine learning for medical diagnosis remains relevant. Previous research has proposed a wide range of ML models for analyzing electronic medical records, with most studies using k-means, KNN, and SVM.

Research results indicate that the use of an ensemble model that combines the mentioned ML methods leads to an increase in diagnostic accuracy. Even though individual ML methods may be effective in specific areas, their combination into an integrated structure allows overcoming limitations that may arise when using only one method.

The advantages and disadvantages of the chosen ensemble model are discussed. The main advantage lies in the increased accuracy of medical diagnosis, which can lead to early detection of diseases and, consequently, improve treatment outcomes. On the other hand, drawbacks may include the complexity of developing and tuning the ensemble approach, as well as computational resource requirements.

It is important to note that the obtained results suggest that combining different models can significantly improve the quality of disease diagnosis. This is a significant contribution to the development of the medical field, as it can contribute to the early detection and effective treatment of various pathologies.

Key words: accuracy; analysis; automation; diagnosis; ensemble; prediction; research.

Tab. 1. Fig. 3. Ref.: 14 items

УДК 004.9:005.94

Дослідження моделей формального опису знань про функціональні вимоги до інформаційної системи з використанням EER-діаграми / А.С. Кожанов. АСУ і прилади автоматики. 2023. № 179. С. 58-72.

Об'єктом дослідження у роботі є формальний опис знань про функціональні вимоги до інформаційної системи (ІС).

У процесах, що працюють із вимогами, часто основою для моделювання вимог до ІС є представлення цих вимог на рівні інформації, виражене у вигляді їх текстових описів або візуальних моделей (ВМ). Такі процеси можуть розглядатися як послідовні перетворення представлень вимог у вигляді даних, інформації й знань під час виявлення та формулювання вимог до ІС відповідно до обраної методології розробки ІС. Однак подібний підхід залишає відкритим питання про вид представлень окремих вимог до ІС, методи і способи формування цих представлень і перетворення цих представлень самих у себе й одне в інше. Тому проведення досліджень з вирішення цього питання є актуальним з теоретичної і прикладної точок зору.

Зазвичай, для опису архітектур ІЗ та ПЗ ІС використовують ER-діаграму та діаграму класів (ДК) відповідно. Але під час розробки, у деяких випадках, виникає проблема сумісності ІЗ та ПЗ через те, що діаграми виконані лише з таких точок зору на відповідні види забезпечення, для яких візуальні описи були створені. Цю проблему пропонується вирішити уніфікацією цих описів та розробкою формального апарату, що використовує фреймову модель для відображення опису функціональних вимог на рівні знань з використанням діаграми класів та ER-діаграми.

Досліджені візуальні нотації функціональних вимог, що описують архітектуру ІЗ (ER-діаграма та EER-діаграма) та ПЗ (ДК). За результатами дослідження спільних елементів діаграм та запропонованої мережі фреймів виявлено, що мережа фреймів може відображати опис функціональних вимог на рівні знань у EER-діаграму. Отриманий результат дозволяє визначити ВМ (EER-діаграму), яка буде використовуватися для формального апарату візуального відображення описів функціональних вимог до ІС на рівні знань, і запропонувати механізм трансформації цієї ВМ у ER-діаграму та ДК, які, відповідно, описують архітектуру ІЗ та ПЗ ІС.

Ключові слова: візуальне моделювання, діаграма класів, мережа фреймів, функціональні вимоги, ER-модель, EER-модель.

Табл. 1. Іл. 0. Бібліогр.: 19 назв.

UDC 004.9:005.94

Study of models of formal description of knowledge about information systems' functional requirements using EER-diagram / Adrian Kozhanov // Management Information System and Devices. 2023. № 179. P. 58-72.

The object of research is a formal description of knowledge about the functional requirements for the information system.

In processes that work with requirements, the basis for modeling requirements for an information system is often the representation of these requirements at the level of information, expressed in the form of their textual descriptions or visual models. Such processes can be considered as successive transformations of representations of requirements in the form of data, information and knowledge during the identification and formulation of requirements for an information system in accordance with the chosen methodology of information system development. However, such an approach leaves open the question of the type of representations of individual requirements for the information system, the methods and methods of forming these representations and transforming these representations into themselves and into each other. Therefore, conducting research on solving this issue is relevant from a theoretical and applied point of view.

Usually, the ER diagram and the class diagram are used to describe information support and software of information system, respectively. But during development, in some cases, the problem of compatibility of information support and software arises because the diagrams are made only from such points of view of the

corresponding types of support for which the visual descriptions were created. This problem is proposed to be solved by the unification of these descriptions and the development of a formal apparatus that uses a frame model to display the description of functional requirements at the level of knowledge using a class diagram and an ER diagram.

Visual notations of functional requirements describing the architecture of information support (ER-diagram and EER-diagram) and software support (class diagram) were studied. According to the results of the study of the common elements of the diagrams and the proposed network of frames, it was found that the network of frames can reflect the description of functional requirements at the level of knowledge in the EER diagram. The obtained result makes it possible to define a visual model (EER-diagram), which will be used for the formal apparatus of visual display of descriptions of the functional requirements for the information system at the level of knowledge, and to propose a mechanism for transforming this visual model into an ER-diagram and a class diagram, which, respectively, describe architectures of information support and software of information system.

Key words: class diagram, EER-model, ER-model, functional requirements, network of frames, visual modeling.

Tab. 1. Fig. 0. Ref.: 19 items

**ПРАВИЛА ОФОРМЛЕННЯ СТАТЕЙ
У ВСЕУКРАЇНСЬКОМУ МІЖВІДОМЧОМУ НАУКОВО-ТЕХНІЧНОМУ ЗБІРНИКУ
«АВТОМАТИЗОВАНІ СИСТЕМИ УПРАВЛІННЯ І ПРИЛАДИ АВТОМАТИКИ»**

1. Загальні вимоги

До розгляду приймаються раніше не опубліковані статті українською та англійською мовами. Статті англійською мовою подаються разом з українськомовним варіантом. Статті, перекладені англійською за допомогою комп'ютерного перекладача та не відредаговані належним чином, не розглядаються.

Наукова стаття, яка подається до розгляду, має бути структурована та містити всі основні частини, характерні для наукової статті:

- постановка проблеми у загальному вигляді та її зв'язок з важливими науковими та практичними задачами;
- аналіз останніх досліджень та публікацій, у яких розпочато вирішення даної проблеми та на які спирається автор, виділення невирішених раніше частин загальної проблеми;
- формулювання цілей статті (постановка задачі);
- подання основного матеріалу досліджень з повним обґрунтуванням отриманих результатів;
- висновки даного дослідження та перспективи подальших досліджень у даному напрямку;
- перелік посилань (References).

2. Вимоги до структури рукопису

Структурно матеріали статті поділяються на такі елементи:

- УДК;
- прізвища та ініціали авторів статті;
- заголовки статті;
- анотація до статті;
- основний текст статті;
- перелік посилань;
- дата надходження статті до редколегії збірника;
- відомості про авторів статті;
- реферати українською та англійською мовами.

Бажаний порядок та зміст розділів основного тексту статті:

а) розділ 1 «Вступ», в якому визначається проблема у загальному вигляді та її зв'язок з важливими науковими та практичними задачами;

б) розділ 2 «Аналіз літературних джерел та визначення проблеми дослідження», в якому наводяться результати аналізу останніх досліджень та публікацій, де розпочато вирішення даної проблеми та на які спирається автор, виділяються невирішені раніше частини загальної проблеми дослідження та конкретизується головна проблема дослідження у даній статті;

в) розділ 3 «Мета і задачі дослідження», в якому наводяться описи мети дослідження та задач дослідження, вирішення яких дозволяє досягти визначеної раніше мети дослідження;

г) розділ 4 «Матеріали і методи дослідження», в якому наводяться описи формального апарату та раніше проведених експериментальних досліджень, які будуть використані у подальшому тексті статті;

д) розділ 5 «Результати дослідження», в якому структуровано наводяться результати вирішення сформульованих у розділі 3 окремих задач дослідження (теоретичних та експериментальних);

е) розділ 6 «Обговорення результатів дослідження», в якому наводяться: опис особливостей отриманих результатів дослідження та їх відмінності від результатів попередніх досліджень у відповідній галузі; опис переваг отриманих результатів перед існуючими; опис недоліків і обмежень, які утруднюють використання отриманих результатів дослідження; опис подальших перспектив проведення досліджень за цим напрямом;

ж) розділ 7 «Висновки», в якому наводяться стислі описи отриманих результатів вирішення окремих задач дослідження та загальний висновок про досягнення поставленої у розділі 3 мети дослідження.

Заголовки окремих розділів основного тексту статті можуть змінюватися відповідно до змісту конкретної статті.

Розділи основного тексту статті, перелік посилань, дата надходження статті до редколегії збірника та відомості про авторів статті відокремлюються один від одного одним порожнім рядком.

3. Вимоги до оформлення рукопису

До розгляду приймаються матеріали статей обсягом не менше 5 повних сторінок (з урахуванням рисунків і таблиць).

Матеріали статті повинні бути набраними у редакторі MS Word. Припустимі формати файлу з матеріалами статті – .doc або .docx.

Формат сторінки – А4 (210x297 мм). Поля знизу, зверху, справа, зліва – 3 см.

Основний текст статті набирається шрифтом Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Для УДК – шрифт Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервал перед – 0 мм, інтервал після – 6 мм, вирівнювання по ширині.

Для прізвищ та ініціалів авторів статті – шрифт Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 6 мм, вирівнювання по ширині.

Для заголовка статті – шрифт Times New Roman, кегль 11, напівжирний, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 6 мм, вирівнювання по ширині.

Для анотації – шрифт Times New Roman, кегль 10, інтервал – 1,1, відступ зліва – 0,8 см, абзацний відступ – 8 мм, інтервал перед – 6 мм, інтервал після – 0 мм, вирівнювання по ширині.

Для заголовків таблиць – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацного відступу немає, інтервали перед і після – 0 мм, слово «Таблиця» та її номер – з вирівнюванням вправо, назва таблиці (якщо вона є) – з вирівнюванням по центру.

Для підписуваних підписів – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацного відступу немає, інтервали перед і після – 0 мм, вирівнювання по центру.

Для переліку посилань та відомостей про авторів – шрифт Times New Roman, кегль 9, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Для рефератів – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Формули набираються у редакторі формул Microsoft Equation або MathType, розташовуються у центрі робочого поля, нумерація – з правої сторони поля. Для цього необхідно весь рядок розташувати справа, а потім вирівняти формулу табуляціями так, щоб вона розташовувалася по центру. Відступ зверху і знизу – по 6 пунктів. Нумерація формул усередині кожної статті наскрізна.

Формули, а також їх складові, присутні у тексті, набираються з такими параметрами (див. рис. 1).

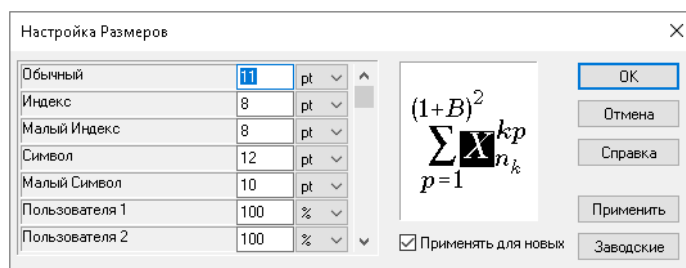


Рис. 1. Параметри настроювання розмірів редактора формул MathType

Кожна таблиця виконується та розташовується в тексті одразу після посилання на неї. Усі таблиці у статті обов'язково нумеруються, незважаючи на їх кількість. Таблиця відокремлюється від попереднього та наступного тексту (таблиці, рисунку тощо) одним порожнім рядком.

Дані всієї таблиці набираються шрифтом розміром 10 пунктів, розміщуються по центру; у випадках, коли необхідно показати розрядність, – вирівнювання за знаком. Товщина сіткі таблиці – 1 пункт. Приклад оформлення таблиці наведено на рис. 2.

Таблиця 1

Множина описів сутностей функціональної задачі

| ID | Найменування |
|----|------------------|
| 1 | Academic_load |
| 2 | Academic |
| 3 | Department |
| 4 | Individual_plan |
| 5 | Academic_section |

Рис. 2. Приклад оформлення таблиці у тексті статті.

Бажано таблицю зі сторінки на сторінку не переносити. Якщо таблиця не може розміститися на сторінці, її поділяють на частини. У кожній частині таблиці повторюють її головку та боковик або замінюють їх відповідно номерами колонок або рядків, нумеруючи їх арабськими цифрами на першій частині таблиці. Слово «Таблиця» подається лише над першою її частиною. Над наступними її частинами праворуч друкується: «Продовження таблиці», а на останній – «Кінець таблиці», в усіх випадках вказується номер таблиці.

Кожен рисунок виконується та розташовується в тексті одразу після посилання на нього. Усі рисунки в статті обов'язково нумеруються, незважаючи на їх кількість. Необхідно вставляти рисунки у текст як графічні об'єкти (файли з розширенням .bmp, .jpg, .tiff чи .png, якість не менше 300 dpi), об'єкти MS Word або MS Visio.

Рисунок відокремлюється від попереднього та наступного тексту (таблиці, рисунку тощо) одним порожнім рядком.

Кожен рисунок повинен мати підписковий підпис, в якому вказується номер та, у випадку необхідності, назва рисунку. Якщо рисунок займає менше 50 % ширини робочого поля, то можна зробити обтікання рисунку текстом, розташувавши його ліворуч або праворуч від робочого поля. Приклад рисунку з підписом наведений на рис. 3.

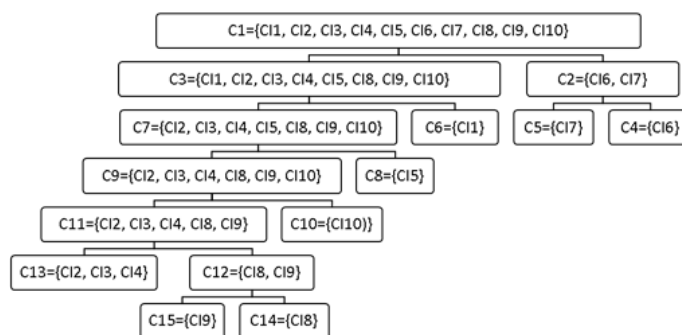


Рис. 3. Приклад виконання рисунку та підпискового підпису

Посилання на літературні та електронні джерела у тексті статті позначаються у квадратних дужках [1]. До переліку посилань включаються тільки ті роботи, на які посилається автор статті. Посилання на неопубліковані роботи не допускаються.

Для оформлення переліку посилань слід використовувати один з таких шаблонів:

а) шаблон IEEE (автоматичне оформлення за шаблоном IEEE <https://www.citethisforme.com/ieee/source-type>);

б) положення ДСТУ 8302:2015 «Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання» та ДСТУ 3582:2013 «Інформація та документація. Бібліографічний опис. Скорочення слів і словосполучень українською мовою. Загальні вимоги та правила».

Кожен з цих шаблонів слід використовувати для оформлення усіх елементів переліку посилань. Використання двох шаблонів для оформлення одного й того ж переліку посилань неприпустимо.

Кожне посилання у переліку посилань наводиться за порядком появи цих посилань у тексті статті.

У переліку посилань бажано використовувати посилання на сучасні публікації, вік яких не перевищує п'яти років у момент подачі статті до редакції. Крім того, під час формування переліку посилань статті необхідно дотримуватися такого розподілу: самоцитування – до 20 %, цитування зарубіжних публікацій – не менше 50%.

Відомості про авторів слід наводити українською та англійською мовами. У відомості про авторів слід включати: повні прізвище, ім'я та по-батькові; вчений ступінь (за наявності); вчене звання (за наявності); посаду; країну, місто; e-mail (вкрай бажано вказувати

корпоративний e-mail, можна вказувати кілька e-mail, на які ви бажаєте отримувати повідомлення від редакції та читачів, які можуть зацікавитися вашою статтею); ORCID.

Реферат набирається українською та англійською мовами. Реферат повинен бути змістовним, дотримуватися логіки опису результатів у статті та давати можливість встановити її основний зміст. Реферат не повинен містити формул та рисунків. Необхідні символи в рефераті необхідно додавати через функцію вставки символів.

Реферат містить: УДК, назву статті (напівжирним шрифтом), ініціали та прізвища авторів (курсивом), текст (не менше 1800 друкованих знаків з пробілами та ключовими словами), ключові слова, кількість таблиць, рисунків та посилань у статті.

Ключові слова повинні містити до 10 слів, а не словосполучень, без використання аббревіатур, в іменному відмінку, розділятися крапкою з комою.

Реферати надаються до редколегії разом із статтею у вигляді окремого файлу.

4. Правила надсилання статей та подальшої взаємодії з редакційною колегією збірника

До редколегії збірника «АСУ і прилади автоматики» слід надсилати такі матеріали:

- файл у форматі .doc або .docx з текстом статті українською мовою;
- файл у форматі .doc або .docx з текстом статті англійською мовою (якщо автори бажають опублікувати статтю у збірнику англійською мовою);
- файл (у форматі .doc або .docx з текстами рефератів статті українською та англійською мовами;
- відскановану копію експертного висновку з дозволом опублікувати матеріали статті у відкритому друку. В разі потреби експертні висновки для авторів – співробітників (студентів, аспірантів тощо) ХНУРЕ можуть оформлюватися редколегією централізовано.

Матеріали статей надсилати електронною поштою – за адресою misd@nure.ua.

Кожна надіслана в редакцію стаття після проходження рецензування і при позитивному рішенні редколегії буде надрукована в найближчому випуску збірника. Для цього авторам від імені редколегії надсилається ліцензійний договір, який закріплює право першої публікації статті у збірнику «АСУ і прилади автоматики». Автори статті повинні підписати цей ліцензійний договір та завірити свої підписи печаткою організації, в якій вони працюють. Підписаний ліцензійний договір автори статті надсилають на адресу редколегії збірника.

Відповідальний випусковий В.М. Левикін
Редактор О.Є. Неумивакіна
Комп'ютерна верстка М.В. Євланов, О.Є. Неумивакіна
Дизайн обкладинки номера за участю Є.Чех

Підп. до друку 27.11.2023. Формат 60x841/8. Умов. друк. арк.
Обл.-вид. арк. 9,8. Тираж 300 прим.
Зам. № 144. Ціна договірна.

Харківський національний університет радіоелектроніки (ХНУРЕ).
Україна, 61166, Харків, пр. Науки, 14

Оригінал-макет підготовлено у редакційно-видавничому відділі ХНУРЕ,
Україна, 61166, Харків, пр. Науки, 14

Збірник віддруковано в ТОВ «ДРУКАРНЯ МАДРИД»
61024, м. Харків, вул. Гуданова, 18
Тел.: +38(057)7565325
Свідоцтво суб'єкта видавничої справи
Серія ДК № 4399 від 27.08.2012 р.
www.madrid.in.ua e-mail:info@madrid.in.ua