ISSN 2524-2601 (Online)
ISSN 2304-6201 (Print)

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

ВІСНИК

Харківського національного університету імені В.Н. Каразіна

Серія
«Математичне моделювання.
Інформаційні технології.
Автоматизовані системи управління»

Випуск 62

Серія заснована 2003 р.

BULLETIN

of V.N. Karazin Kharkiv National University

Series
«Mathematical Modeling.
Information Technology.
Automated Control Systems»

Issue 62

First published in 2003

Харків
2024

Засновник журналу Харківський національний університет імені В. Н. Каразіна, Харків, Україна. Рік заснування 2003. Періодичність: 4 випуски на рік. https://periodicals.karazin.ua/mia

Статті містять дослідження у галузі математичного моделювання та обчислювальних методів, інформаційних технологій, захисту інформації. Висвітлюються нові математичні методи дослідження та керування фізичними, технічними та інформаційними процесами, дослідження з програмування та комп'ютерного моделювання в наукоємних технологіях.

Для викладачів, наукових працівників, аспірантів, працюючих у відповідних або суміжних напрямках.

Наказом Міністерства освіти і науки України від 17.03.2020 № 409 наукове фахове періодичне видання Вісник Харківського національного університету імені В.Н. Каразіна серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» включено до Категорії «Б» Переліку наукових фахових видань України за наступними спеціальностями: 113 – Прикладна математика;122 – Комп'ютерні науки та інформаційні технології; 123 – Комп'ютерна інженерія; 125 – Кібербезпека.

Затверджено до друку рішенням Вченої ради Харківського національного університету імені В. Н. Каразіна (протокол № 11 від 21.06.2024 р.)

Адреса редакційної колегії: 61022, м. Харків, майдан Свободи, 6, ХНУ імені В. Н. Каразіна, к. 534. Тел. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Мова публікації: українська, англійська.

Статті пройшли внутрішнє та зовнішнє рецензування.

Ідентифікатор медіа у Реєстрі суб'єктів у сфері медіа: R30-04456
(Рішення № 1538 від 09.05.2024 р Національної ради України з питань телебачення і радіомовлення. Протокол № 15)

The founder of the Journal is V. N. Karazin Kharkiv National University, Kharkiv, Ukraine.
Year of foundation 2003. The journal is published four times a year.
https://periodicals.karazin.ua/mia

The articles are present research in the field of mathematical modeling and computing methods, information technologies, information security. New mathematical methods of research and management of physical, technical and information processes, research on programming and computer modeling in science-intensive technologies are covered.

For teachers, researchers, graduate students working in relevant or related fields.

By the order of the Ministry of Education and Science of Ukraine from 17.03.2020 № 409 scientific professional periodical Bulletin of V.N. Karazin Kharkiv National University series "Mathematical modeling. Information Technologies. Automated control systems" is included in Category "B" of the List of scientific professional publications of Ukraine in the following specialties: 113 – Applied Mathematics, 122 – Computer Science and Information Technology; 123 – Computer engineering; 125 – Cybersecurity.

Approved for publication by the decision of the Academic Council of V.N. Karazin Kharkiv National University (Minutes № 11 of 21.06.2024).

Editorial Address: 61022, Kharkiv, Svobodi sq., 6, V.N. Karazin Kharkiv National University, r. 534.
Phone. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Language of publication: Ukrainian, English.

The articles pass internal and external review.

Media identifier in the Register of the field of Media Entities: R30-04456
(Decision № 1538 dated May 9, 2024 of the National Council of Television and Radio Broadcasting of Ukraine, Protocol № 15)

ЗМІСТ

CONTENTS

DOI: https://doi.org/10.26565/2304-6201-2024-62-01

УДК (UDC) 004.051

Даас Тимур
Імад Ахмад
аспірант кафедри моделювання систем і технологій
Харківський національний університет імені В. Н. Каразіна, майдан Свободи 4, м. Харків, Україна, 61022
e-mail: timurkadaas@gmail.com
https://orcid.org/0000-0003-2796-8145

Ткачук Микола Вячеславович
д.т.н., професор; професор кафедри моделювання систем і технологій
Харківський національний університет імені В. Н. Каразіна, майдан Свободи 4, м. Харків, Україна, 61022
e-mail: mykola.tkachuk@karazin.ua;
https://orcid.org/0000-0003-0852-1081

Аналіз сучасного стану і перспектив розвитку у галузі розробки та супроводу систем інтернет-банкінгу

Актуальність. Розробка систем інтернет-банкінгу потребує вирішення проблеми їх проектування з урахуванням постійного розвитку галузі та появи нових вимог до ПЗ, як функціональних, так і нефункціональних. Побудова таких систем з забезпеченням належного рівня показників якості вимагає обґрунтованого вибору еталонної системної архітектури, а також використання сучасних методів обробки знань щодо вимог користувачів. І тому, питання побудови цього класу систем є актуальною науково-технічною задачею.
Мета. Метою цього дослідження є аналіз сучасного стану та перспектив розвитку у галузі розробки та супроводу систем e-Banking, що дозволить мотивовано запропонувати їх вдосконалення шляхом застосування знання-орієнтованого підходу для обробки знань щодо вимог користувачів системи та використання мікросервісної архітектури, як еталонної системної архітектури, що в кінцевому рахунку має на меті підвищення показників якості.
Методи дослідження. Для досягнення мети дослідження виконано огляд існуючих в Україні систем e-Banking, проведено їх порівняльний аналіз та синтезована схема їх типової функціональності. Для подальшого розвитку цього класу систем мотивовано обрано метод аналізу часових рядів, як основний метод прогнозування дій користувача при роботі с функціоналом побудови виписок за рахунком. Виконано збір інформації щодо поточних показників продуктивності систем при побудові виписок.
Результати. На основі проведеного аналізу поточного стану, зроблено обґрунтований висновок про можливість та доцільність підвищення продуктивності систем e-Banking шляхом використання знання-орієнтованого підходу для обробки знань щодо вимог користувачів, а також мікросервісної архітектури при проектування та розробці систем e-Banking. Мотивовано запропоновано запровадити новий окремий модуль прогнозування дій користувача у системі, який дозволить системі виконувати важкі за часом та ресурсами операції клієнтів заздалегідь, для оптимального використання системних ресурсів.
Висновки. Розглянуті проблеми пов'язані з побудовою та розробкою систем e-Banking, зроблено аналітичний огляд існуючих на ринку України систем та побудована схема їх типової функціональності. Запропоновано розробити модуль прогнозування дій користувача у системі, який використовує знання-орієнтовані методи, та використати його для прогнозування операції побудови користувачем виписки за рахунками та попередньої підготовки цієї виписки для підвищення продуктивності роботи систем e-Banking.

Ключові слова: *інтернет-банкінг, знання-орієнтований підхід, мікросервісна архітектура, метрики якості, продуктивність, метод аналізу часових рядів, рекомендаційна система, виписка за рахунком.*

Як цитувати: Даас Т.І., Ткачук М.В. Аналіз сучасного стану і перспектив розвитку у галузі розробки та супроводу систем інтернет-банкінгу. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2024. вип. 62. С.6-18.
https://doi.org/10.26565/2304-6201-2024-62-01

How to quote: T.I. Daas, M.V. Tkachuk, "Analysis of the current state and future prospects in the domain of development and maintenance of Internet banking systems", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modeling. Information technology. Automated control systems*, vol. 62, pp. 6-18, 2024. [In Ukrainian] https://doi.org/10.26565/2304-6201-2024-62-01

1. Актуальність проблем розробки і супроводу систем e-Banking. Мета і задачі дослідження

У сучасному світі зростає необхідність впровадження все більшої кількості електронних послуг у банківській сфері, зокрема у системах інтернет-банкінгу (e-Banking). Оскільки системи e-Banking є основним класом банківських інформаційних систем, з якими безпосередньо взаємодіє клієнт банку, який є основним джерелом прибутку будь-якої банківської установи. Перелік таких послуг та їх складність зростає щоденно. Бізнес-вимоги користувачів вимагають від розробників забезпечення швидкого впровадження нового функціоналу у системи e-Banking, забезпечення мінімізації, а краще, відсутності впливу нових модулів на якість та логіку роботи вже впровадженого функціоналу, а також забезпечення відмовостійкості при пікових навантаженнях на систему.

У свою чергу, банки, як кінцеві власники продукту, також висувають свої вимоги до програмного забезпечення (ПЗ) систем e-Banking, серед них можна виділити: мінімізація ресурсів при горизонтальному масштабуванні, мінімізація витрат на супровід ПЗ, можливість моніторингу робочого стану системи, впровадження вимог регулятора (Національний банк України - НБУ) у відведені строки, а також швидке усунення знайдених вразливостей. Останнє є вкрай важливим в умовах значного росту кількості кібератак на критичну інфраструктуру держави [1].

В умовах коли надання банком послуг своєму клієнту повинно відбуватись у найкоротші строки і з мінімізацією кількості відвідувань клієнтом відділень банку, системи e-Banking, які забезпечують дистанційне обслуговування клієнтів, є одним з основних інструментів для продажу банківських продуктів та ведення бізнесу банком в цілому. Якість роботи цих систем є одним з основних факторів, які сприяють заохоченню нових клієнтів для будь-якого банку [2].

Більшість систем e-Banking, які використовуються українськими банками, досі побудовані з використанням монолітних архітектур (МА), які за своїми властивостями гірше підходять для забезпечення вищенаведених вимог, аніж мікросервісні архітектури (МСА), які у свою чергу забезпечують гнучкість IT-інфраструктури, незалежність компонентів один від одного, можливість адаптивного масштабування та здатність до швидкої модернізації ПЗ [3].

Метою цього дослідження є аналіз сучасного стану та перспектив розвитку у галузі розробки та супроводу систем e-Banking, шляхом огляду та порівняльного аналізу існуючих систем, синтезу їх типової функціональності. Також однією з задач дослідження є мотивований вибір напрямків та підходів для вдосконалення і розвитку цих систем. Для підвищення продуктивності систем e-Banking в даній роботі пропонується застосування знання-орієнтованого підходу при розробці цих систем для обробки знань щодо вимог користувачів, а також використання МСА, як еталонної системної архітектури. Кінцева мета даного підходу є підвищення якості обслуговування клієнтів банку.

Пропонована концепція має на меті забезпечити ефективну обробку накопичених знань та впровадження результатів їх обробки у системи e-Banking, що доповнює вже існуючі підходи до розробки програмного забезпечення таких систем. У роботі [4] описуються принципи та переваги застосування модельного підходу у системах e-Banking, а також наведені Computational Independent Model та Platform Independent Model для систем e-Banking, які є певною концептуальною моделлю для систем цього класу та описують основні функції системи і їх взаємозв'язок. У роботі [5] компанія Yalantis, яка спеціалізується на розробці ПЗ у сфері фін-тех, спираючись на світовий досвід, наводить принципи застосування МСА, як reference architecture, у типових системах e-Banking, описує виклики з якими стикається при впровадженні МСА та виділяє певні функціональні рівні, які характерні системам цього класу. Але існуючі дослідження не описують використання саме знання-орієнтованих моделей та методів у системах e-Banking.

2. Порівняльний огляд деяких існуючих систем e-Banking

На підставі аналізу інформаційних джерел було зроблено порівняльний огляд деяких існуючих таких систем, які представлені в Україні [6 - 13].

Усі існуючі системи e-Banking можна умовно розділити на два класи:

- розвинені комерційні рішення;
- системи власної розробки.

Варто відзначити, що для систем e-Banking відсутній такий клас, як "безкоштовні програмні продукти" (open source). Це обумовлено тим, що системи e-Banking мають відповідати

міжнародним стандартам, таким як наприклад PCI DSS, а також постійно впроваджувати нові вимоги національного регулятора.

До програмних комерційних рішень із групи (1) належать, наприклад, такі системи як iFOBS, яка розроблена компанією CS, і iBank2, яка розроблена компанією ДБО Софт. Перший продукт є одним з лідерів на ринку і є більш популярним серед юридичних осіб та ФОП, аніж для фізичних осіб. Абсолютна більшість банків, при наданні своїх послуг для юридичних осіб, використовують саме систему iFOBS. Система iBank2 також більше використовується банками при наданні послуг юридичним особам, але є менш популярною через меншу кількість функціоналу та інтеграційних можливостей.

Програмні продукти групи (2) також використовуються досить широко, і зазвичай розробляються власними командами програмістів найбільш успішних банків, які краще представлені у сегменті фізичних осіб. Це обумовлено необхідністю швидкого впровадження оновлень для реакції на зміни у суспільних процесах, а також бажанням мати власний унікальний дизайн застосунку. Характерними представниками цього класу є системи e-Banking таких банків як Приватбанк, Монобанк та А-Банк.

Нижче стисло розглянемо характеристики деяких найбільш поширених систем e-Banking, з метою побудови та аналізу, в подальшому, схеми їх типової функціональності.

2.1. Система iFOBS

Ця система є однією з найбільш популярних систем на українському ринку, оскільки є комерційним продуктом, який має велику різноманітність функціоналу та можливості для інтеграції з core-системами банку, такими як, наприклад, автоматизовані банківські системи(АБС) чи CRM-системи. Система представлена у двох системних архітектурах: монолітній та мікросервісній.

Система надає для клієнтів Web-інтерфейс, Windows-додаток і мобільні додатки під платформи IOS та Android, а також інтерфейс для адміністрування. Додатково система підтримує для клієнтів відкриті інтерфейси для зовнішньої інтеграції (Open API) [6, 7].

Серед найбільш цікавих особливостей цього продукту можна виділити:

- наявність окремих інтерфейсів для клієнтів фізичних осіб, юридичних осіб, а також ФОП. Кожен інтерфейс розроблений з урахуванням специфіки роботи і необхідного функціоналу кожної з груп користувачів
- наявність гнучкої рольової системи як для користувачів, так і для адміністраторів
- широкий спектр підтримуваних засобів авторизації та їх комбінації
- висока ступінь масштабованості системи у версії з мікросервісною архітектурою

2.2. Система iBank2

Ця система також є комерційним продуктом, але вона менше представлена на ринку через гіршу різноманітність функціоналу, а також гіршу продуктивність при високому навантаженні. Система наразі представлена лише у монолітній архітектурі, але все одно експлуатується декількома банками національного рівня, такими як, наприклад, Райффайзен Банк чи Таскомбанк.

Система надає для клієнтів лише Web-інтерфейс і мобільні додатки під платформи IOS та Android, а також інтерфейс для адміністрування [8, 9].

Особливостями даної системи є:

- наявність окремих інтерфейсів для клієнтів фізичних осіб, юридичних осіб, а також ФОП
- наявність сертифікованих криптографічних бібліотек власного виробництва, а також центру сертифікації ключів(ЦСК) [10, 11]
- можливість двофакторної авторизації за допомогою додатку Google Authenticator

Серед недоліків системи можна виділити погані можливості для масштабування, які обумовлені використаною архітектурою, а також клієнтський інтерфейс, який не відповідає вимогам сучасного UI/UX-дизайну.

2.3. Система Privat24

Система Privat24 є власною розробкою Приватбанку - найбільшого українського банку. Ця система є найбільш використовуваною за кількістю користувачів. Як і попередня, ця система

надає для клієнтів лише Web-інтерфейс і мобільні додатки під платформи IOS та Android, а також інтерфейс для адміністрування [12].

Серед особливостей системи можна виділити:

- наявність окремих інтерфейсів для клієнтів фізичних осіб та юридичних осіб
- найбільшу різноманітність функціоналу
- наявність засобів детального моніторингу дій користувача
- гарні можливості масштабування системи при зміні в середовищі функціонування

Система повністю наразі побудована з використанням мікросервісної архітектури, і є яскравим прикладом плавного переходу від монолітної архітектури до мікросервісної.

2.4. Система Monobank

Система Monobank є власною розробкою однойменного банку. Ця система була введена в експлуатацію у 2017 році і є однією з найновіших e-Banking систем, які діють на українському банківському ринку. Через новизну система є єдиним представником, серед наведених в цьому документі, яка з самого початку була побудована та розроблена з використанням МСА. Але наразі, на відміну від попередніх перерахованих систем, ця система представлена лише у вигляді мобільних додатків під платформи IOS та Android, і має інтерфейси лише для клієнтів фізичних осіб [13]. Це можна вважати основними недоліками порівняно з аналогами.

Серед інших особливостей виділяються:

- наявність засобів детального моніторингу дій користувача
- гарні можливості масштабування системи при зміні в середовищі функціонування
- найшвидше впровадження нового функціоналу в продуктивне середовище

2.5. Порівняльний аналіз наведених систем

Для проведення порівняльного аналізу властивостей вищезгаданих систем e-Banking, синтезу загальної схеми їх типової функціональності, а також для розробки мотивованої пропозиції щодо її можливого подальшого вдосконалення, у цій роботі використана методика, яка була запропонована у [14] для опрацювання аналогічної проблематики у галузі розробки складних систем управління IT-інфраструктурою підприємств.

Результат порівняння розглянутих в даному розділі систем наведено у таблиці 1. Оцінки виставлялися групою експертів за 5-ти бальною шкалою.

Таблиця 1. Порівняння систем e-Banking

Критерії	iFOBS	iBank2	Privat24	Monobank
Використовувана архітектура	Частково мікросервісна	Монолітна	Мікросервісна	Мікросервісна
Масштабованість	4	3	5	5
Гнучкість конфігурування	5	4	5	5
Супроводжуваність	4	4	4	5
Інтерфейс	4	3	4	5
Різноманітність функціоналу	5	4	5	3
Безпека	4	4	4	4

Продуктивність	5	4	5	4
Можливості для інтеграції зовнішніх систем	4	4	5	4

Проведений порівняльний аналіз цих продуктів дозволяє визначити основні критерії, за якими можна виконати оцінку типової системи e-Banking. В наступних розділах буде побудована типова функціональність існуючих в Україні систем e-Banking, а також буде запропонований шлях її вдосконалення за рахунок впровадження знання-орієнтованого підходу у процес проектування та розробки цього класу банківських інформаційних систем. Пропонований підхід включає також використання мікросервісної архітектури, яка забезпечує кращі показники якості для цих систем.

2.6. Типова функціональність систем e-Banking

Типова функціональність систем e-Banking має забезпечувати весь спектр послуг, які банк, як фінансова установ, повинен надавати клієнтам у дистанційному вигляді. Як було зазначено у попередніх розділах, системи класу e-Banking є такими що динамічно розвиваються, і, відповідно, формалізована схема типової функціональності систем e-Banking, яка наведена на рис.2.1 у вигляді UML-діаграми пакетів, може розглядатися як певна концептуальна модель “ідеальної системи”, але лише станом на зараз. В процесі розвитку можуть додаватися нові функції, але зазвичай це будуть функції 3-го рівня, згідно наведеної діаграми.

Операції клієнтів. Функції цієї групи мають стратегічне значення для кожної системи e-Banking, оскільки саме вони забезпечують безпосередню взаємодію банку та його клієнтів, кількість та рівень задоволення яких відіграє головну роль в успішності функціонування банку як бізнесу. Саме у цій групі функцій основну роль відіграє дизайн, швидкодія та відмовостійкість, оскільки клієнта банку не цікавить внутрішня складова системи, її архітектура чи стек технологій. Для клієнта важливо отримати якісну послугу, без збоїв в роботі системи та з виконанням якомога меншої кількості дій у системі. Для функцій цієї групи обов'язково має бути розроблений та підтримуватися певний рівень обслуговування (Service Level Agreement - SLA).

Система адміністрування. Ця група процесів відповідає за роботу співробітників банку, які безпосередньо взаємодіють з клієнтами: працюють у відділеннях або у відділі підтримки. Система адміністрування повинна забезпечувати повний спектр функції, які потрібні для створення та редагування облікових записів користувачів системи, в тому числі і видачі нових прав чи ролей користувачам, для виконання операції авторизації різноманітних дій користувача та/або іншого адміністратора в системі, в тому числі і в режимі “maker - checker”(або 4-eyes), для фіксації та перегляду усіх успішних та неуспішних дій будь-якого користувача, особливо критичних фінансових операції та авторизації.

Забезпечення безпеки. Функції цієї групи є одними з головних оскільки в умовах постійного розвитку методів для здійснення кібератак, банківська сфера є однією з найпріоритетніших для здійснення цих атак, оскільки обробляє цінний актив - гроші. До функцій цієї групи відносять модулі, що відповідають за вияв шахрайських та аномальних операцій, блокування їх чи накладання додаткового фактору для підтвердження, а також за сповіщення відповідального підрозділу банку про наявність підозрілих платежів у конкретного клієнта чи групи клієнтів. Для захисту від відомих та найбільш загрозливих атак на веб-додатки мають бути впроваджені функції аналізу вхідних запитів до системи для запобігання обробки модулями бізнес-логіки системи запитів, які мають ознаки атаки на веб-додаток. Перелік актуальних вразливостей підтримується учасниками відкритого проекту захисту веб-додатків(OWASP) [15], в рамках якого також розробляється керівний документ, який де-факто є стандартом, захисту веб-додатків при проектуванні та розробці OWASP ASVS [16]. Необхідність керуватися документами розробленими OWASP є наразі вимогою Національного банку України [17].

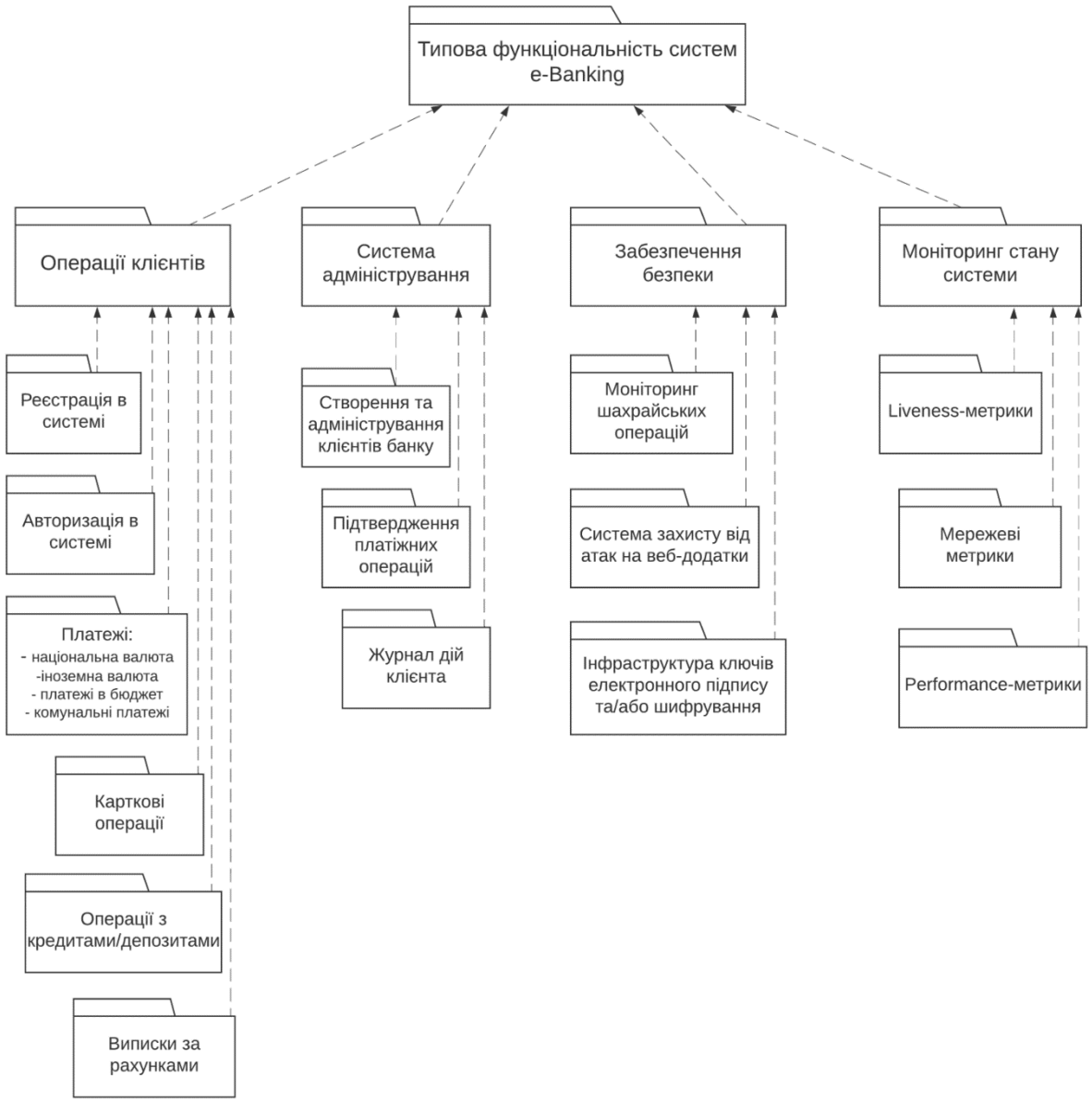


Рис. 2.1 Типова функціональність систем e-Banking

Моніторинг стану системи. Функції моніторингу забезпечують IT-відділам банку, які супроводжують систему, можливість оперативного реагування на позаштатні інциденти в системі, а також розслідування цих подій. До основних метрик, які повинні бути забезпечені системами e-Banking відносять внутрішні показники стану компонентів системи, які також називають liveness-метриками, мережеві показники як при взаємодії різних компонентів системи між собою, так і при взаємодії з сторонніми системами, а також метрики продуктивності ключових бізнес-функцій системи. Для зберігання та накопичення цих метрик може бути як розроблений інтегрований в систему e-Banking модуль, так і використовуватися корпоративні системи обробки, накопичення та візуалізації метрик системи, які вже впроваджені в банку як корпоративний стандарт. До таких систем можна віднести, наприклад, Grafana, Prometheus, Zipkin, Jaeger [18-21].

3. Побудова перспективної функціональної схеми системи e-Banking

Однією з основних функцій системи e-Banking є побудова виписок за рахунками. Ця операція є однією з найчастіше використовуваних функцій типової системи e-Banking [22]. Визначення, структура, призначення виписок та особливості їх побудови наведені у [23, 24]. За своєю суттю побудова виписки за рахунком є операцією час виконання якої, а також необхідні ресурси, залежать від вхідних умов, таких як: кількість обраних рахунків, кількість усіх операцій за

рахунком, обраний період виписки, формат виписки (наприклад, PDF, XLS, XML, тощо), необхідність сортування, тощо. Всі ці чинники призводять до двох проблем:

- неможливо однозначно оцінити час виконання операції і створити певні SLA-критерії
- потрібен великий об'єм ресурсів, а особливо спільних, таких як робота бази даних, в процесі виконання операції побудови виписки з великим обсягом платежів

Друга проблема негативно впливає на функціонування системи в цілому, оскільки призводить до того, що модулі, які виконують інші операції в системі, очікують звільнення ресурсу бази даних. Особливо це відчутно у періоди, які є найбільш типовими для побудови виписок багатьма клієнтами системи e-Banking. Яскравим прикладом є побудова річної виписки в останніх числах грудня, коли такі виписки будують і найбільші за обсягом платежів клієнти банку. При цьому, системи, які побудовані на основі монолітних архітектур, через використання спільних ресурсів та середовища усіма модулями системи відчувають більший негативний ефект, ніж системи на основі мікросервісної архітектури.

Саме тому доцільно розглянути деякі можливості знання-орієнтованого підходу до проектування та розробки систем e-Banking, який дозволив би виконувати аналіз поведінки та попередніх дій користувача у системі при побудові будь-яких виписок та використати його для прогнозування майбутніх дій цього користувача пов'язаних з побудовою виписки. Отримані прогнози пропонується використати для можливості системи "діяти на упередження" і готувати дані для "прогнозованої виписки" заздалегідь і, що найбільш важливо, у час найменшої завантаженості системи. Наприклад, якщо користувач кожен останній день місяця, на протязі вже півроку, будує місячну виписку за своїм гривневим рахунком, то є велика ймовірність того, що й наступного місяця він також буде будувати місячну виписку за цим рахунком. Одним з використовуваних підходів до такого прогнозування є метод аналізу часових рядів, який дозволяє для описаної моделі, на основі раніше зібраних даних, виконати прогнозування [25]. Метод відноситься до категорії формалізованих математичних методів, що базується на статистиці [26]. Складові частини прогнозу наведені у [27] та схематично зображені у вигляді блок-схеми на рисунку 3.1.

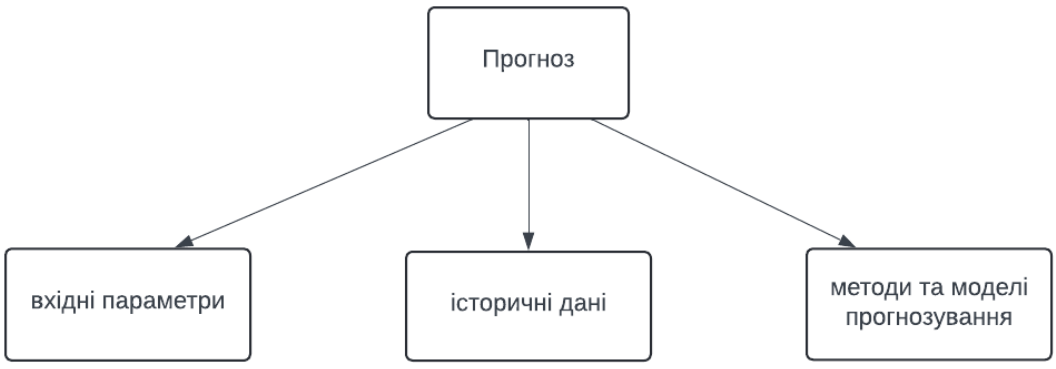


Рис. 3.1 Блок-схема складових частин прогнозу

Вхідними параметрами можуть бути поточний час та стан рахунків користувача. Історичними даними є усі минулі операції з побудови виписок цього конкретного користувача, а також інших користувачів системи, в тому числі і метаінформація операції: обраний період виписки, обрані рахунки, час формування виписки, формат виписки, тощо. До моделей та методів прогнозування можна віднести семантичні правила обробки та аналізу збереженої історичної інформації, методи зберігання цієї інформації, побудовану доменну модель операції, а також метод отримання "прогнозу" в конкретний момент часу.

Слід окремо зазначити, що методи аналізу часових рядів також застосовуються при розробці сучасних рекомендаційних систем (recommender system) різного призначення (див., наприклад у [28, 29, 30]), що також може бути одним з чинників мотивації для обрання запропонованого підходу до розширення функціональності перспективної системи e-Banking.

З урахуванням цих міркувань пропонується розширити типову функціональність систем e-Banking новим модулем, як зазначено на рисунку 3.2. На цій схемі групу функцій “операції клієнтів” доповнено “універсальним модулем прогнозування дій користувача”. В контексті піддомену виписок цей модуль відповідає за обробку історичної інформації, прогнозування дій користувача в системі при побудові виписок за рахунками, а також повинен створювати у розрахований ним час системні задачі по побудові “прогнозованої виписки”, які вже далі обробляються модулем побудови виписок з урахування прогнозованих параметрів цієї виписки. В майбутньому функціонал універсального модуля прогнозування може бути розширений можливістю прогнозування дій користувача у системі в цілому. До таких дій можна, наприклад, віднести: заповнення параметрів створюваних платежів, оплата комунальних послуг, підбір умов кредитів та депозитів.

Рис. 3.2 Розширена функціональність системи e-Banking

Використання мікросервісної архітектури дозволить отримати максимальний результат від запропонованого підходу. Через властивість мікросервісної архітектури, яка полягає в розділенні системи на окремі незалежні сервіси, то є можливість відокремлення нового модулю в окремий мікросервіс, який не матиме впливу на стабільність роботи інших клієнтських операцій та системи в цілому[31]. Також впровадження мікросервісної архітектури забезпечить набагато більш гнучке масштабування системи з мінімальними ресурсами при пікових навантаженнях, що забезпечує підвищення продуктивності системи e-Banking в цілому, оскільки максимально знижує вплив операції формування виписок на інші функції системи, особливо в порівнянні з монолітною архітектурою, в якій існують спільні ресурси [32, 33]. В контексті цієї роботи це в першу чергу стосується мікросервісу, який відповідатиме в системі за доменну область “формування виписок за рахунками”, оскільки саме на нього покладатиметься корисне навантаження по побудові прогнозованих виписок.

4. Мотивація доцільності застосування знання-орієнтованого підходу при розробці системи e-Banking для підвищення їх ефективності

Для виявлення існуючих проблем в побудові та роботі систем e-Banking, у період лютий-квітень 2024 року було проведено опитування співробітників відділів супроводу та розробки української компанії CS, яка є розробником системи iFOBS. Було виявлено, що операція побудови виписки за рахунком є найбільш довгою операцією в системі, а також, через необхідність оперування великим об’ємом даних при виконанні, створює найбільш негативний вплив на швидкодію інших операцій користувачів в системі. Одним з прикладів такого впливу було зниження у 4 рази продуктивності операцій з проведення гривневих платежів через заповнення кешованої області пам’яті у СУБД Oracle даними, які отримувалися сервером для формування річної виписки одного з найбільших клієнтів банку, який є корпорацією національного масштабу та має досить велику кількість рахунків і проведених за цими рахунками платежів. Зниження

продуктивності спостерігалося в проміжок часу, впродовж якого формувалася вищезгадана виписка.

Наразі в системі iFOBS побудова виписки за рахунком виконується синхронно та в момент запиту користувача. Побудова річної виписки займає від 1-2 секунд для користувачів із категорії малого/середнього бізнесу, до декількох десятків секунд для корпорацій національного масштабу. У таблиці 2 наведена статистика формування виписок за рахунками у форматі pdf, який наразі є найбільш популярним серед користувачів, в залежності від кількості сторінок сформованої виписки. Процедура формування виписки складається з двох основних частин:

- підготовка даних про платежі на основі вхідних параметрів виписки
- формування PDF-файлу за допомогою бібліотеки JasperReports [34]

Таблиця 2. Статистика формування виписок за рахунками

Кількість сторінок	Підготовка даних (мс)	Формування PDF-файлу (мс)	Усього (мс)
1	40	120	160
5	60	220	280
10	70	350	420
20	80	600	680
50	200	1500	1700
100	350	3100	3450
150	800	4200	5000
360	2500	12000	14500
500	4500	15000	19500
850	5500	40000	45500

Отримана статистика виконання операцій формування виписки користувачами системи показує, що є певні часові закономірності таких операцій та їх параметрів. Все це дозволяє зробити висновок, що впровадження "модулю прогнозування дій користувача" для операцій побудови виписок за рахунками є цілком доцільним і має допомогти в досягненні основної мети роботи, а саме підвищення продуктивності роботи системи e-Banking. Пропоноване рішення в першу чергу направлене на банки всеукраїнського масштабу, які серед своїх клієнтів мають як корпорації національного масштабу, для яких потрібно забезпечити можливість обробки великих об'ємів платіжних даних, так і велику кількість користувачів з категорії фізичних осіб та/або малого-середнього бізнесу, які через свою велику кількість постійно створюють навантаження на систему і потребують певний рівень якості наданих банком дистанційних послуг впродовж усього часу роботи системи.

5 Висновки та напрямок подальших досліджень

В статті розглянуті деякі проблеми пов'язані побудовою та розробкою систем e-Banking, зроблено аналітичний огляд існуючих на ринку України систем та побудована схема їх типової

функціональності. Для підвищення продуктивності роботи цих систем запропоновано розробити модуль прогнозування дій користувача у системі, який використовує знання-орієнтовані методи, а саме метод аналізу часових рядів в поєднанні з семантичними правилами, та використати його для прогнозування операції побудови користувачем виписки за рахунками та попередньої підготовки цієї виписки для оптимізації витрат ресурсів системи на цю операцію та зниження впливу операції формування виписки на функції системи в цілому.

В наступних роботах планується описати особливості застосування мікросервісної архітектури в поєднанні зі знання-орієнтованими методами при побудові систем e-banking. В подальшому планується розробити прототип відповідного модуля системи та дослідити ефективність його застосування у реальній системі e-Banking.

СПИСОК ЛІТЕРАТУРИ

1. Tan T.H., Tan T.H. E-Banking SAF: A TOGAF-NIST Aligned Security Architecture Framework for E-Banking Systems. *The 7th International Conference on Information and Computer Technologies*. 2024. Honolulu, Hawaii, USA. P. 1–6
2. Коваль В. Правове визначення інформаційної безпеки електронного банкінгу. *Молодий вчений.* 2023. №1 (113), С. 121–125. https://doi.org/10.32839/2304-5809/2023-1-113-25
3. Newman S. Building Microservices, 2nd Edition. New York. USA: O`Reilly Media, 2021. 616 p.
4. Avornicului M.C., Bresfelean Vasile P. Model driven development of online banking systems. *Annals of the University of Oradea, Economic Science Series*, 2011, Vol 20, Issue 1, p. 795– 800
5. Roid I., Panasenko Y., Huba S. Digital banking architecture: things to consider when building banking software. URL: https://yalantis.com/blog/technical-side-of-digital-banking-software (дата звернення: 01.07.2024)
6. IFOBS.Corporate сайт. [Електронний реурс] URL: https://csltd.com.ua/products/corporate_online_banking/ (дата звернення: 03.07.2024)
7. IFOBS.Private сайт. [Електронний ресурс] URL: https://csltd.com.ua/products/private_online_banking/ (дата звернення: 02.07.2024)
8. iBank2.Corporate сайт. [Електронний ресурс] URL: https://dbosoft.com.ua/#/products/business/web_banking_business/ (дата звернення: 01.07.2024)
9. iBank2.Private сайт. [Електронний ресурс] URL: https://dbosoft.com.ua/#/products/private/web_banking_private/ (дата звернення: 01.07.2024)
10. IBank2, криптобіблиіотека “Гепард 2.0” [Електронний ресурс] URL: https://dbosoft.com.ua/assets/about/licenses/eo_2099.pdf (дата звернення: 01.07.2024)
11. IBank2, центр сертифікації ключів “Integra CA” [Електронний ресурс] URL: https://dbosoft.com.ua/assets/about/licenses/eo_1541.pdf (дата звернення: 01.07.2024)
12. Privat24 сайт. [Електронний ресурс] URL: https://privatbank.ua/udalenniy-banking/privat24 (дата звернення: 01.07.2024)
13. Monobank сайт. [Електронний ресурс] URL: https://www.monobank.ua/ (дата звернення: 01.07.2024)
14. Ткачук М.В., Сокол В.Є. Деякі проблеми управління ІТ-інфраструктурою підприємства: сучасний стан та перспективи розвитку. *Східно-Європейський журнал передових технологій.* 2010. № 6/2 (48). С. 68–72.
15. OWASP Top 10 сайт. [Електронний ресурс] URL: https://owasp.org/www-project-top-ten (дата звернення: 01.07.2024)
16. OWASP Application Security Verification Standard v4.0.3
17. Про затвердження Положення про організацію заходів із забезпечення інформаційної безпеки в банківській системі України: Постанова НБУ №95 від 28.09.2017р.
18. Офіційний сайт Grafana. [Електронний ресурс] URL: https://grafana.com (дата звернення: 02.07.2024)
19. Офіційний сайт Prometheus. [Електронний ресурс] URL: https://prometheus.io (дата звернення: 02.07.2024)
20. Офіційний сайт Zipkin. [Електронний ресурс] URL: https://zipkin.io (дата звернення: 02.07.2024)
21. Офіційний сайт Jaeger. [Електронний ресурс] URL: https://www.jaegertracing.io (дата звернення: 03.07.2024)

22. Тищенко О. І. Огляд сучасних тенденцій на ринку онлайн-банкінгу в Україні. *Електронний журнал "Економіка і суспільство"*. 2017. №13. С. 1237–1243. URL: https://economyandsociety.in.ua/journals/13_ukr/206.pdf (дата звернення: 01.07.2024)
23. Tripathi P. What is a Bank Statement: Components, Purpose & How to Process It. URL: https://www.docsumo.com/blogs/bank-statement-extraction/what-is-bank-statement (дата звернення: 29.06.2024)
24. Brosens J., Kruger R. M., Smuts H. Guidelines for Designing e-Statements for e-Banking. *The Second African Conference for Human Computer Interaction: Thriving Communities*. December 2018. p. 1–6
25. Shah D., Thaker M. A Review of Time Series Forecasting Methods. *International journal of research and analytical reviews.* April 2024. Vol. 11, No. 2. p. 749–755
26. Buchatskaya V., Buchatsky P., Teploukhov S. Forecasting Methods Classification and its Applicability. *Indian Journal of Science and Technology*. November 2015. Vol. 8(30)
27. Liu Z., Zhu Z., Gao J., Xu C. Forecast methods for time series data: A survey. *IEEE Access*. 2021, Vol. 9, p. 91896–91912
28. Gupta V. Modeling Time-Series and Spatial Data for Recommendations and Other Applications: PhD dissertation / Indian Institute of Technology Delhi. 2022. 178 p.
29. Joorabloo N., Jalili M., Ren Y. A new temporal recommendation system based on users` similarity prediction. *11th International Conference on Knowledge Discovery and Information Retrieval.* 2019. p. 555 – 560
30. Gomez-Losada A., Duch N. Time Series Forecasting by Recommendation: An Empirical Analysis on Amazon Marketplace. *International Conference on Business Information Systems*. 2019. Vol. 1, p. 45 – 54
31. Newman S. Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith. New York. USA: O'Reilly, 2019. 256 p.
32. Hamzehloui M., Sahibuddin S. and Ashabi, A. A Study on the Most Prominent Areas of Research in Microservices. *International Journal of Machine Learning and Computing*. 2019. Vol. 9, No. 2. p. 242–247.
33. Mendonca N., Jamshidi P., Garlan D., Developing Self-Adaptive Microservice Systems: Challenges and Directions. *IEEE Software*. 2019. Vol. 38 (Issue 2). p. 70–79.
34. Офіційний сайт Jaspersoft. [Електронний ресурс] URL: https://community.jaspersoft.com/ (дата звернення: 04.07.2024)

REFERENCES

1. T. H. Tan and T. K. Tan, "E-Banking SAF: A TOGAF-NIST Aligned Security Architecture Framework for E-Banking Systems," The 7th International Conference on Information and Computer Technologies., 2024, pp. 1-6.
2. V. Koval, "Legal Definition of Information Security in Electronic Banking," Young Scientist, no. 1(113), pp. 121-125, 2023. [Online]. Available: https://doi.org/10.32839/2304-5809/2023-1-113-25 [in Ukrainian]
3. S. Newman, Building Microservices, 2nd ed., New York, USA: O'Reilly Media, 2021
4. M. C. Avornicului and V. P. Bresfelean, "Model driven development of online banking systems," Annals of the University of Oradea, Economic Science Series, vol. 20, no. 1, pp. 795-800, 2011.
5. I. Roid, Y. Panasenko, and S. Huba, "Digital banking architecture: things to consider when building banking software," [Online]. Available: https://yalantis.com/blog/technical-side-of-digital-banking-software. [Accessed: Jul. 1, 2024].
6. IFOBS.Corporate site, [Online]. Available: https://csltd.com.ua/products/corporate_online_banking/ [Accessed: Jul. 1, 2024].
7. IFOBS.Private site, [Online]. Available: https://csltd.com.ua/products/private_online_banking/. [Accessed: Jul. 1, 2024].
8. IBank2.Corporate site, [Online]. Available: https://dbosoft.com.ua/#/products/business/web_banking_business/. [Accessed: Jul. 1, 2024].
9. iBank2.Private site, [Online]. Available: https://dbosoft.com.ua/#/products/private/web_banking_private/. [Accessed: Jul. 1, 2024].

10. iBank2, Crypto library 'Gepard 2.0', [Online]. Available: https://dbosoft.com.ua/assets/about/licenses/eo_2099.pdf. [Accessed: Jul. 1, 2024].
11. iBank2, Key Certification Center 'Integra CA', [Online]. Available: https://dbosoft.com.ua/assets/about/licenses/eo_1541.pdf. [Accessed: Jul. 1, 2024].
12. Privat24 Site, [Online]. Available: https://privatbank.ua/udalenniy-banking/privat24. [Accessed: Jul. 1, 2024].
13. Monobank Site, [Online]. Available: https://www.monobank.ua/. [Accessed: Jul. 1, 2024].
14. M. V. Tkachuk and V. Ye. Sokol, "Some issues of enterprise IT infrastructure management: current state and development prospects," Eastern-European Journal of Enterprise Technologies, no. 6/2(48), pp. 68-72, 2010. [in Ukrainian]
15. OWASP Top 10 site, [Online]. Available: https://owasp.org/www-project-top-ten. [Accessed: Jul. 1, 2024].
16. OWASP Application Security Verification Standard v4.0.3, [Online].
17. National Bank of Ukraine, "On the approval of the Regulation on the organization of measures to ensure information security in the banking system of Ukraine," Resolution NBU No. 95, Sep. 28, 2017. [in Ukrainian]
18. Official site "Grafana", [Online]. Available: https://grafana.com. [Accessed: Jul. 1, 2024].
19. Official site "Prometheus". [Online]. Available: https://prometheus.io. [Accessed: Jul. 1, 2024].
20. Official site "Zipkin". [Online]. Available: https://zipkin.io. [Accessed: Jul. 1, 2024].
21. Official site "Jaeger". [Online]. Available: https://www.jaegertracing.io. [Accessed: Jul. 1, 2024].
22. O. I. Tyshchenko, "Review of modern trends in the online banking market in Ukraine," Electronic Journal "Economy and Society", no. 13, pp. 1237-1243, 2017. [Online]. Available: https://economyandsociety.in.ua/journals/13_ukr/206.pdf. [Accessed: Jul. 1, 2024]. [in Ukrainian]
23. P. Tripathi, "What is a Bank Statement: Components, Purpose & How to Process It," [Online]. Available: https://www.docsumo.com/blogs/bank-statement-extraction/what-is-bank-statement. [Accessed: Jun. 29, 2024].
24. J. Brosens, R. M. Kruger, and H. Smuts, "Guidelines for Designing e-Statements for e-Banking," in Proc. 2nd African Conf. Human Computer Interaction: Thriving Communities, Dec. 2018, pp. 1-6.
25. D. Shah and M. Thaker, "A Review of Time Series Forecasting Methods," International journal of research and analytical reviews, vol. 11, no. 2, pp. 749-755, Apr. 2024.
26. V. Buchatskaya, P. Buchatsky, and S. Teploukhov, "Forecasting Methods Classification and its Applicability," Indian Journal of Science and Technology, vol. 8, no. 30, Nov. 2015.
27. Z. Liu, Z. Zhu, J. Gao, and C. Xu, "Forecast methods for time series data: A survey," IEEE Access, vol. 9, pp. 91896-91912, 2021.
28. V. Gupta, "Modeling Time-Series and Spatial Data for Recommendations and Other Applications": PhD thesis, Dept. of Computer Science and Engineering, Indian Institute of Technology Delhi, Delhi, India. 2022. 178 p.
29. N. Joorabloo, M. Jalili, Y. Ren, "A new temporal recommendation system based on users` similarity prediction," 11th International Conference on Knowledge Discovery and Information Retrieval, pp. 555 – 560, 2019.
30. A. Gomez-Losada, N. Duch, "Time Series Forecasting by Recommendation: An Empirical Analysis on Amazon Marketplace," International Conference on Business Information Systems, vol. 1, pp. 45 – 54, 2019
31. S. Newman, Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith, New York, USA: O'Reilly, 2019.
32. M. Hamzehloui, S. Sahibuddin, and A. Ashabi, "A Study on the Most Prominent Areas of Research in Microservices," International Journal of Machine Learning and Computing, vol. 9, no. 2, pp. 242-247, 2019.
33. N. Mendonca, P. Jamshidi, and D. Garlan, "Developing Self-Adaptive Microservice Systems: Challenges and Directions," IEEE Software, vol. 38, no. 2, pp. 70-79, 2019.
34. Official site "Jaspersoft". [Online]. Available: https://community.jaspersoft.com/. [Accessed: Jul. 1, 2024].

Daas Tymur *PhD student of the Department of Systems and Technology Modeling, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine;*

Tkachuk Mykola *Doctor of Technical Sciences, Professor; Professor of the Department of Systems and Technology Modeling, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine.*

Analysis of the current state and future prospects in the domain of development and maintenance of Internet banking systems

Relevance. The development of Internet banking systems requires solving the problem of their design, where the constant development of the industry and the emergence of new requirements, both functional and non-functional, needs to be taken into account. The development of such systems requires a reasonable choice of reference system architecture, as well as the use of modern knowledge-based methods for processing user requirements, to ensure the appropriate level of system quality metrics. Therefore, the issue of building this class of systems is an urgent scientific and technical task.
Goal. The purpose of this study is to analyze the current state and future prospects in the field of development and maintenance of e-Banking systems, which will allow to substantiate their improvement by applying a knowledge-based approach to processing knowledge about the user requirements of system and using microservice architecture as a reference system architecture, which ultimately aims to improve quality metrics.
Research methods. In order to achieve the goal of the study, a review of the existing e-Banking systems in Ukraine was carried out, their comparative analysis was carried out and the diagram of their typical functionality was synthesized. For the further development of this class of systems, the method of time series analysis was chosen as the main method of predicting user actions when working with the functionality of building account statements. The collection of information on the current performance indicators of systems during the construction of statements was carried out too.
The results. Based on the analysis of the current state of affairs, a well-founded conclusion is made about the possibility and feasibility of increasing the performance of e-Banking systems by using a knowledge-oriented approach to processing knowledge about user requirements, as well as microservice architecture in the design and development process of e-Banking systems. Reasonably substantiated proposal to introduce a new separate module for predicting user actions in the system, which will allow the system to perform time- and resource-intensive customer operations in advance for optimal use of system resources.
Conclusions. The considered problems are related to the designing and development of e-Banking systems, an analytical review of existing systems on the Ukrainian market was made, and a diagram of their typical functionality was built. It is proposed to develop a module for predicting user actions in the system, which uses knowledge-oriented methods, and to use it to predict the user`s operation of building a statement of accounts and pre-preparing this statement to improve the performance of e-Banking systems.

Keywords*: e-banking, knowledge-oriented approach, microservice architecture, quality metrics, performance, method of time series analysis, recommender system, account statement..*

Надійшла у першій редакції 18.04.2024, в останній - 21.05.2024.
The first version has been received on 18.04.2024, the final version - on 21.05.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-02

УДК (UDC) 519.216

Danilevskyi Mykhailo *PhD student; V.N. Karazin Kharkiv National University, Svobody Square, 4, Kharkiv-22, Ukraine, 61022*
e-mail: m.danilevskiy@gmail.com
https://orcid.org/0009-0000-0030-2218

Yanovsky Volodymyr *Doctor of Physical and Mathematical Sciences, professor; V. N. Karazin Kharkiv National University, sq. Svobody 4, Kharkiv, Ukraine, 61000; Institute of Single Crystals, National Academy of Sciences of Ukraine, Nauki Ave. 60, Kharkiv, Ukraine, 61001*
e-mail: yanovsky@isc.kharkov.ua
https://orcid.org/0000-0003-0461-749X

Matsiy Olga *PhD of Technical Sciences, docent; V.N. Karazin Kharkiv National University, Svobody Square, 4, Kharkiv-22, Ukraine, 61022*
e-mail: olga.matsiy@gmail.com
https://orcid.org/0000-0002-1350-9418

Modeling and Analysis of a Dynamic Network of Telephone Subscribers Considering the Degree of Connectivity by Means of Contact Lists

Abstract. Modeling complex dynamic networks, whose components interact and evolve, is essential for understanding and predicting their behaviour. It helps to optimize performance, improve resilience, and effectively manage resources in technological, information, social, and biological networks. **Purpose.** The purpose of the work is to model a dynamic network of telephone subscribers, identify and evaluate its main properties. The focus is on experiments with the resulting model and determining the dependencies of network properties based on simulation data. **Research methods.** The methods of constructing computer models, methods of analyzing network parameters, the method of least squares and the Monte Carlo method of stochastic dynamics of discrete states by using time steps of equal length have been used in the work. The computer model has been developed in Python by using the Pandas, Numpy and NetworkX libraries. **Results.** A model of a dynamic network of telephone subscribers is proposed with imitation of contact lists, which usually include family members, colleagues, and friends. Experiments have been conducted with the model and the dependences of network properties on the number of subscribers and the fraction of contacts within contact lists have been investigated. The values of the model parameters at which the network exhibits the properties of a small-world network has been determined. **Conclusions.** The proposed model of a dynamic network of telephone subscribers with imitation of contact lists has allowed to identify the dependences of the network properties on the number of subscribers and the fraction of contacts within the contact lists. It was revealed that the node degree distribution corresponds to the lognormal law. The number of links in the call graph depends on the number of subscribers linearly, and the higher the fraction of contacts, the fewer links are created when a new subscriber appears. An increase in the number of subscribers affects the network density reducing it according to a hyperbolic law. As the fraction of contacts increases, the network density decreases, since an increasing number of connections are created among a limited number of subscribers. The clustering coefficient changes according to a hyperbolic law as well. The average value of the shortest path length for certain network parameters is well approximated by a logarithmic function when the fraction of contacts is more than 0.80 within contact lists. Finally, the qualities of a small-world network can be recognised in the dynamic network of telephone subscribers when the fraction of contacts in the contact list falls between 0.80 and 0.90, as determined by the coefficient ω (4).

Keywords: *dynamic complex network, mobile call graph, telephone network, lognormal distribution, degree distribution, network density, clustering coefficient, average shortest path length, small-world network.*

How to quote: Danilevskyi M., Yanovsky V., Matsiy O., “Modeling and Analysis of a Dynamic Network of Telephone Subscribers Considering the Degree of Connectivity by Means of Contact Lists”, *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp. 19-29, 2024. https://doi.org/10.26565/2304-6201-2024-62-02

Як цитувати: Danilevskyi M., Yanovsky V., Matsiy O Modeling and Analysis of a Dynamic Network of Telephone Subscribers Considering the Degree of Connectivity by Means of Contact Lists. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне*

моделювання. Інформаційні технології. Автоматизовані системи управління. 2024. вип. 62. С.19-29.
https://doi.org/10.26565/2304-6201-2024-62-02

1. Introduction

Various world phenomena – social, physical, biological to name a few – can be represented as complex dynamic networks. In [1], the authors identified four categories of complex networks: technological, informational, social and biological. Representing complex networks as structures that change over time allows identifying structural and temporal patterns. Thus, a dynamic network can be identified as one in which nodes and edges appear and/or disappear over time [2].

Numerous studies have been conducted on the modeling and analysis of complex dynamic networks. In [2], the authors present a classification of dynamic networks and the basics of their modeling using graph neural networks. In [3], the authors propose criteria for choosing a dynamic and static model of a social network to analyze their properties. The social network models predominantly presented in the physics-oriented literature on the complex networks have been classified and compared in [4].

The data obtained from actual networks of phone subscribers have been analyzed in several papers. For instance, in [5], the authors have presented studies on the interaction patterns of millions of mobile phone users. In [6], a network of telephone subscribers of more than 1 million users is studied based on call data presented by the mobile network. The authors evaluate such indicators as the distribution of the number of telephone calls, conversation time, and the number of unique connections per subscriber. In [7], the authors have considered the data on telephone calls from four different geographical regions, have generated call graphs and analyzed their static properties. The dynamic network of telephone subscribers and the influence of the degree of persistence of connections on the structure of the network have been examined in [8]. Other types of dynamic networks have been analyzed as well. The overview of financial networks – definitions of various types of networks and discussion on the dynamic aspect of the functioning and distribution of resources in such networks has been presented in [9]. In [10], the authors describe biological networks, as well as the principles and algorithms of graph neural networks, which can be used to predict the emergence of new connections and diseases.

The papers [5–8] focus on the analysis of existing networks of telephone subscribers. However, only a limited number of works are devoted to the simulation modeling of dynamic networks of telephone subscribers. For example, in [11], the authors proposed a simulation model of a call-center, which allows doing a “what-if” analysis to determine the number of operators required to satisfy a certain requirement for the system. In [12], the authors have developed a telephone network model for an electricity distribution company with the aim of modeling and further optimizing the technical infrastructure of the telephone network. Modeling a dynamic telephone network as a social communications network with different types of subscribers has not been covered in the literature.

The paper proposes a model of a dynamic network of telephone subscribers, in which the number of subscribers is static, and the connections between them change at discrete time intervals According to the terminology presented in [2], such a network can be classified as a discrete node-static temporal network. Furthermore, following the classification of dynamic communication networks proposed in [13], the telephone subscriber network is synchronous with one-to-one connections, which means that connections exist only between two subscribers simultaneously.

2. Modeling dynamic network of telephone subscribers

A network of telephone subscribers is formed by a certain number of nodes (subscribers) and the connections between them. In such a network, a connection is established when two subscribers are engaged in a call. For instance, the average person makes five to seven calls per day, while sale managers may make two or three hundred calls daily. Consequently, at any given moment, a certain number of pairwise connections are established. During the experiment with the model, calls and connections between subscribers have been saved in a log. From the recorded set of paired connections, a dynamic network of telephone subscribers is formed over a specific period. The study aims to analyze the properties of the obtained dynamic network.

The telephone network model proposed in this work differs from the model presented in [14]. The distinction is that each subscriber has a contact list in our model. The fraction of calls between subscribers from the contact list is the specified by the model parameter, and in other cases, the contacts are selected randomly. The lognormal distribution law is used to assign the quantity of calls per day to subscribers

[15]. New calls cannot be placed or received by subscribers while they are engaged in a call. The probability of a call ending at a certain time is used to model the call duration.

The input parameters of the model include the number of subscribers, parameters of the lognormal distribution law for modeling the number of calls per day, the duration of the experiment, the average duration of conversations, the fraction of calls to subscribers from the contact list, and the number of subscribers in the contact list.

With the help of a developed model, such properties of a dynamic network of telephone subscribers as the degree distribution law, the number of created links, network density, clustering coefficient, and the average shortest path length have been investigated.

3. Description of the experiment

During the experiments conducted over a specified period, connections between subscribers are simulated, resulting in a dynamic network. The number of subscribers is predetermined and remains constant. In this model, one unit of time is equivalent to one minute. The duration of the experiment, the parameters of the lognormal distribution law for the daily number of calls, the size of the contact list, and the fraction of calls to subscribers from the contact list are specified. The uniform distribution law is used to select a subscriber for communication, both within the contact list and among all other subscribers. The duration of a call is determined by the probability of its termination at any unit of time, which is set at 0.99 for all experiments. Once a connection is established, subscribers are isolated, which means that they cannot make or receive additional calls. The progress of the experiments is recorded in Table 1:

Table 1. Experiment data sample

idx	caller_id	callee_id	start_time	finish_time
0	12	62	0	1
1	23	4	4	7
2	72	64	8	9

where idx is the record number, caller_id is the caller identifier, callee_id is the callee identifier, start_time is the call starting time, finish_time is the call finishing time, duration is the call duration.

Input data for the experiment: number of subscribers is 1000, duration is 10080 of time units (minutes), the parameters of the lognormal distribution correspond to normal subscribers ($\mu = 1.1, \sigma = 1.0$) [15].

4. Study of the obtained data

During the experiments, data on 34834 subscriber contacts have been obtained. Using this data, the distribution of the number of subscriber calls during the day has been constructed. This distribution can be compared to the distribution obtained in [15], where it has been found that the number of calls follows a lognormal distribution. Therefore, the consistency of the model with the experimental data could be evaluated.

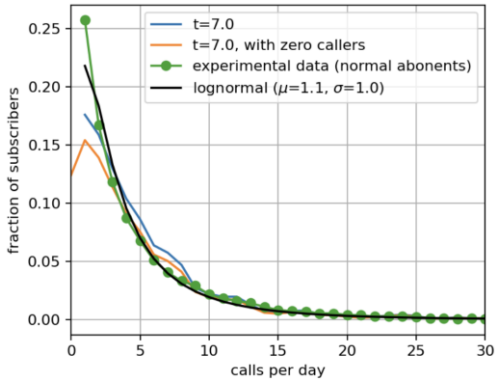


Fig.1 Distribution of the number of subscribers by the number of calls per day. In green – experimental data of a real network, simulation data: yellow – all subscribers are included, blue – only those subscribers who made at least one call, black curve – the lognormal law with μ = 1.1 and σ = 1.0.

The green line corresponds to experimental data from the real telephone network of subscribers [15]. The results obtained from the telephone network simulation are depicted by the orange and blue lines.

The blue line represents the distribution of calls determined without considering subscribers who did not make a single call throughout the entire experiment. The orange line represents the distribution including all subscribers. The black curve corresponds to the data obtained by approximating experimental observations by using a lognormal distribution law with parameters $\mu = 1.1$ and $\sigma = 1.0$.

In the experimental data from a real telephone subscriber network, all subscribers made at least one call. In contrast, the simulated network included subscribers who did not make a single call, as indicated by the distribution depicted by the orange line. That resulted in different data distributions.

The distribution of subscribers who made at least one call during the model simulation is represented by the blue curve in Fig. 1, which shows good agreement with the experimental data, represented by the green curve. Over time, subscribers create a network of connections. We have used the information obtained from experiments with the constructed computer model to conduct a more thorough analysis of the characteristics of the telephone subscriber network.

5. Analysis of dynamic network properties

Numerical experiments have been conducted with varying numbers of subscribers at consistent time intervals to investigate the properties of a dynamic network of telephone subscribers. We have assessed the relationship between these properties and the fraction of calls made within the contact list, as well as the total number of subscribers in the network. Among the properties analyzed, we have examined the number of edges in the network, clustering coefficients, network density, average shortest path length, and degree distribution. Additionally, we have determined the type of network resulting from those model experiments.

A total of 580 experiments have been conducted. The number of subscribers ranged from 10 in the first experiment to 1000 in the last one, with the fraction of calls within the contact list varying from 0.0 to 1.0. The interval was 0.1 from 0.0 to 0.8, and 0.01 from 0.81 to 1.0. In each experiment, the number of subscribers increased by 50. Fig. 2 presents graphs of various networks formed with different fractions of calls but with a consistent number of 100 subscribers. The graphs clearly show the formation of subscriber groups within contact lists and the connections between these groups when calls are made to other subscribers. Those groups or cliques are especially visible in networks with a high fraction of calls within their contact lists.

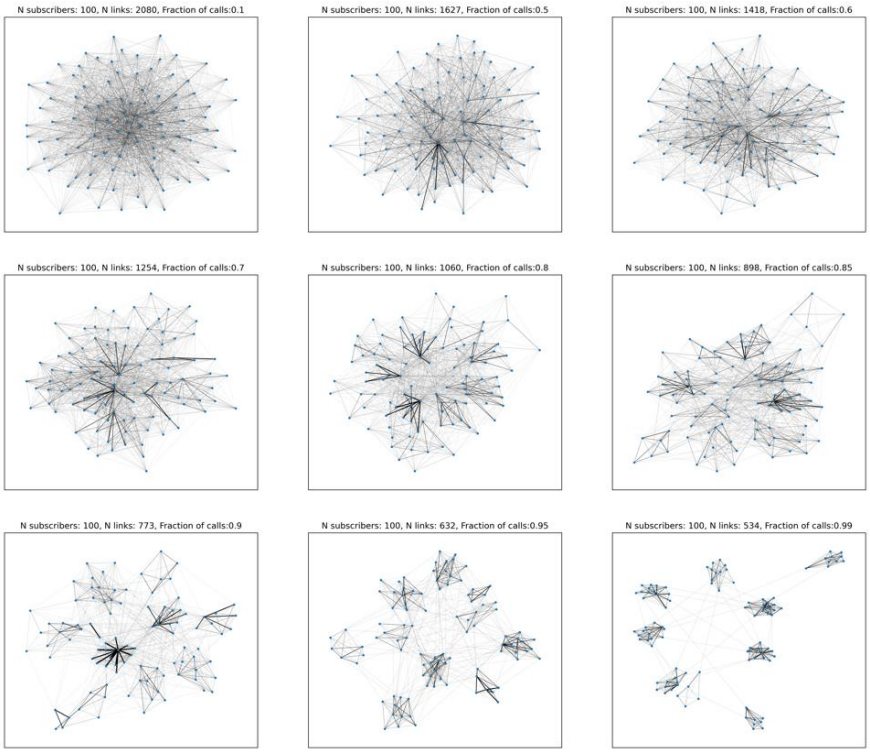
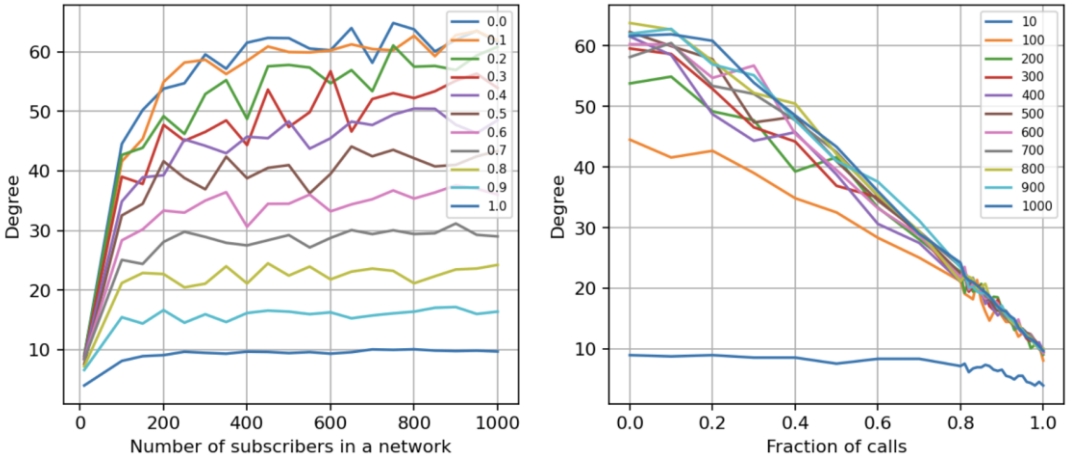


Fig.2 Simulated telephone subscriber networks. N subscribers – number of subscribers, N links – number of links, Fraction of calls – fraction of calls made within subscriber contact lists

The properties of the generated networks have been examined. The form of each network can be explained by using the distribution of node degrees, which represents the number of connections a subscriber established with other subscribers during the experiment. The obtained dependencies are shown in Fig. 3.

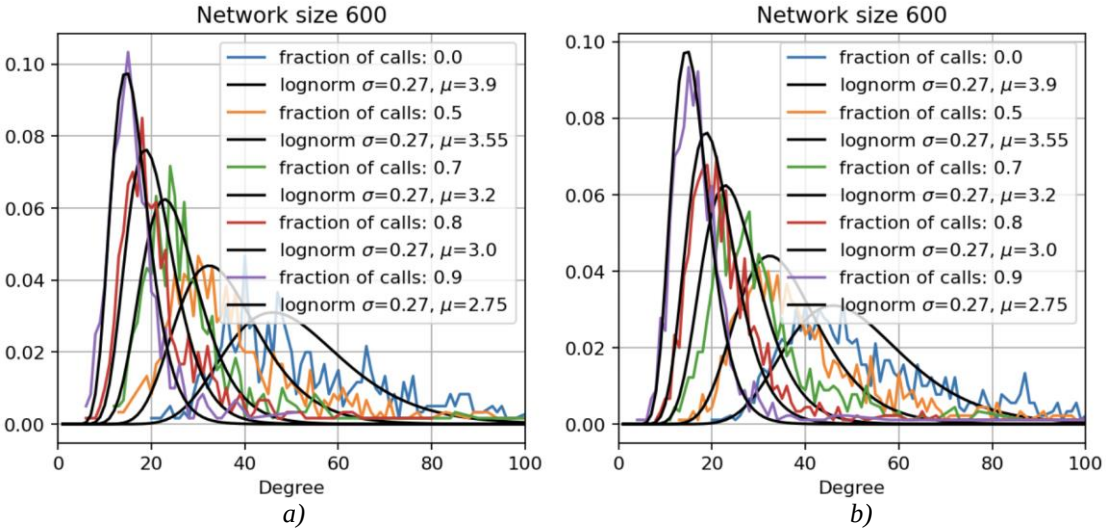


a) *b)*

Fig.3 Average node degrees dependence on a) number of subscribers; and b) fraction of calls within contact lists.

From the graphs presented in Fig. 3, it could be seen that the degrees of the nodes/subscribers weakly depend on the number of subscribers in the network and have a linear dependence on the fraction of calls. For networks with a small number of subscribers where, for example, 10 subscribers have time to form all possible connections, the degree of nodes is equal to the number of subscribers in the network minus 1. Consequently, the higher the fraction of calls within contact lists, the lower the node degrees.

As stated previously, the number of calls per day is distributed according to a lognormal law. The degree of nodes and the number of calls are closely related because connections and degrees in a dynamic network of telephone subscribers are formed through calls between subscribers.



a) *b)*

Fig.4 Distribution of node degrees and approximating lognormal distribution curves: a) network with the 600 subscribers; b) network with the 900 subscribers.

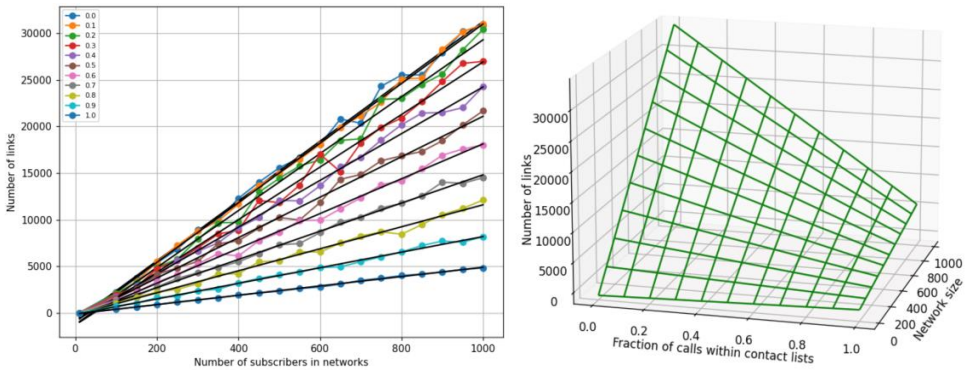
The lognormal distribution curves that correspond to the node degree distributions are shown in Fig. 4. For networks with 600 and 900 subscribers, we have examined whether the node distribution agreed with the lognormal law. The data indicates a decent degree of agreement, with μ depending on the fraction of calls in contact lists (the higher the fraction, the smaller μ); meanwhile, σ remains unchanged.

Next, we have examined the relationship between the number of links and the number of subscribers in the network as well as the fraction of calls. An edge in the call graph between two subscribers is

considered to be a link. The dependencies discovered through model experiments are displayed in Fig. 5, along with straight lines representing them. The results show that the number of links made during the experiment is directly correlated with the total number of subscribers in the network. Additionally, the fraction of calls made within contact lists influences the angle of inclination of the straight line. The higher the fraction is, the smaller the angle is and, consequently, the fewer links made over a given amount of time. The relationship between those data, when approximated linearly by using the least squares method, is as follows:

$$L = -30.15NF + 1212.57F + 35.57N - 1184.21 \quad (1)$$

where L is the number of links, N is the number of subscribers in the network, F is the fraction of calls within the contact list.



a) *b)*

Fig.5 Dependence of the number of links on a) the number of subscribers; b) the number of subscribers and the fraction of calls within contact lists.

From that dependence it follows that each new subscriber on average creates 5 new links during the experiment (10800 discrete intervals) when the fraction of calls is equal to 1. For instance, with a fraction of 0.7, the subscriber creates on average about 15 new links etc. Fig. 5b shows the dependence of the number of links on the number of subscribers and the fraction of calls in contact lists.

Next, we present simulation data which shows how the network density changes with an increase in the number of subscribers over the same time interval for different values of the fraction of contacts. Network density is the ratio of an actual number of links to the maximum number of links that can be created with the same number of nodes [1]. The obtained relationships, as well as their approximations, are shown in Fig. 6.

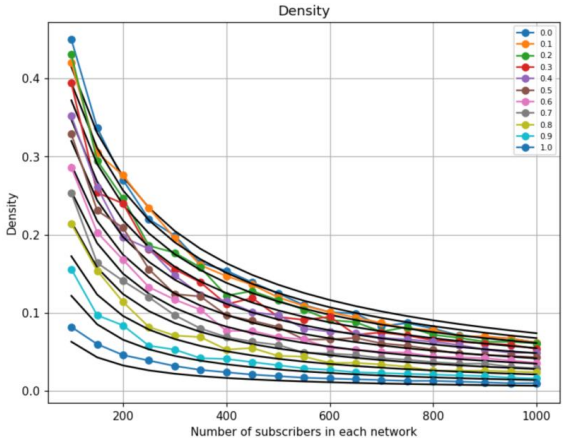


Fig.6 Dependence of network density on the number of subscribers in the network and the fraction of calls

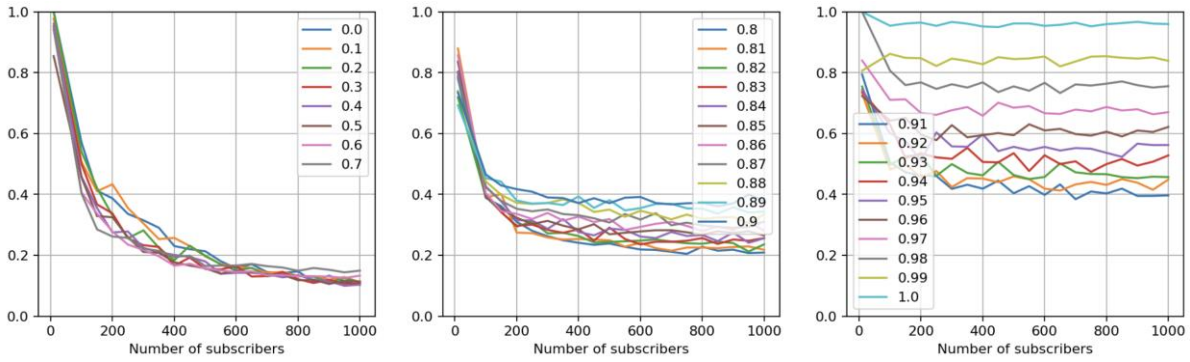
The simulation data is shown in Fig. 6 by the lines with dots, and the approximation of the dependencies is shown in black. An examination of the data has revealed a hyperbolic dependence of density on the number of subscribers in the network and the fraction of calls:

$$\rho = \frac{0.85}{1+\frac{N}{95.30-87.29F}} \tag{2}$$

where ρ is the network density.

Good agreement with the simulation data is noticeable in Fig. 6. As the number of subscribers increases, the network density decreases. Additionally, as the fraction of calls increases, the network density decreases further. That is due to the fact that a higher fraction of calls leads to fewer new links being created among the possible ones, and vice versa. The network density value determined by relation (2) also depends on the duration of the experiment, necessitating further research.

The next network metric analyzed is the average clustering coefficient, which measures the degree of interconnection for subscriber's contacts. A higher clustering coefficient indicates greater connectivity within the graph. Fig. 7 shows how the average clustering coefficient varies with the number of subscribers and the fraction of calls.



a) *b)* *c)*

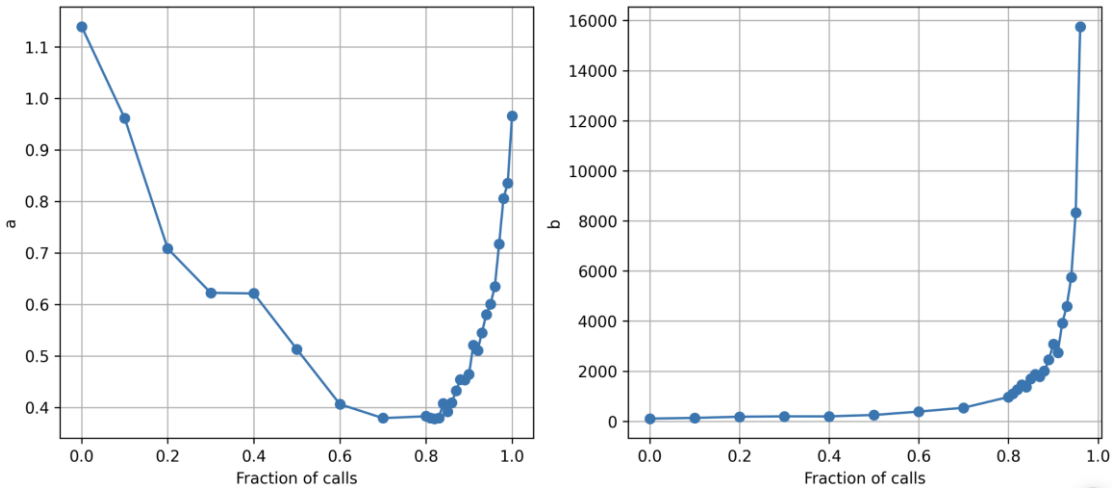
Fig.7 Dependence of the clustering coefficient on the number of subscribers in networks with different fractions of calls

In Fig. 7, the hyperbolic dependence of the clustering coefficient on the number of subscribers in the network with the fraction of calls in the interval [0.0, 0.7] can be observed.

$$C = \frac{a}{1+\frac{N}{b}} \tag{3}$$

where C is the clustering coefficient, a and b are the parameters of the hyperbolic dependence.

The parameters a and b depend on the fraction of calls. The corresponding dependencies are presented in Fig. 8.

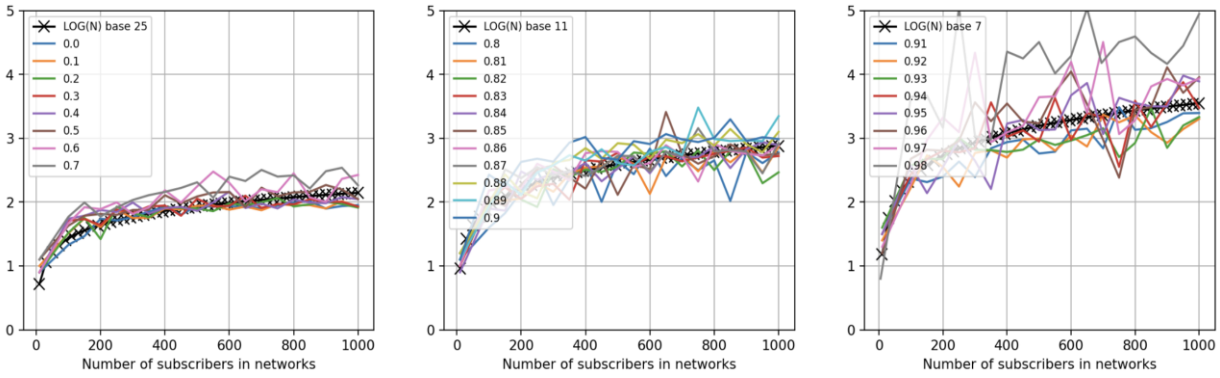


a) *b)*

Fig.8 Dependence of parameters "a" and "b" on the fraction of calls

It should be noted that the experiments have been carried out with network sizes from 10 to 1000 subscribers and for the same time interval. This explains the transformation of the hyperbolic curve, which depends on the fraction of calls, and the values of the obtained parameters are valid over the time interval of the experiment.

Next, we have considered the changes in the average shortest path length obtained as a result of the simulation. The lengths of all links between adjacent nodes are considered to be equal to one. The average internode distance is the average path over all pairs of nodes between which there is at least one path connecting them. In the categories of the telephone network and connections between people, this indicator characterizes the average number of contacts required to connect any two subscribers.



a) *b)* *c)*

Fig.9 Dependence of the average shortest path length between two subscribers on the number of subscribers in the network and the proportion of contacts within contact lists

In small-world networks, the average shortest path length between two nodes grows in proportion to the logarithm of the number of nodes [16] or subscribers based on the average number of contacts in the case of the telephone network. In the model, the dependence of the average length of the shortest path on the number of subscribers in the network is well approximated by the logarithm to base 11 and 7 for networks with the fraction of contacts in the intervals [0.80, 0.90] and [0.91, 0.98], which is shown in Fig. 9b and 9c, respectively. For networks with fractions of up to 0.7, the logarithmic curve in Fig. 9a corresponds to the logarithm of 25, which is too many regular contacts for normal subscribers.

Let us check if any of the networks obtained as a result of the experiments are small-world networks. Small-world networks are characterized by a high clustering coefficient and a short path length [16, 17]. The distribution of node degrees in small-world networks can be similar to the distributions in random networks [18], so they can be Poisson or normal distributions. The degree to which a network is a small-world network can be measured using the coefficient ω [17], which considers the clustering coefficient and the average shortest path length and compares them with the equivalent regular lattice and random network, respectively:

$$\omega = \frac{L_{rand}}{L} - \frac{C}{C_{latt}} \quad (4)$$

where ω is the small-world measurement

The values of ω are in the range from -1 to 1. Moreover, if the value of ω tends to 1, then it means that the network has the characteristics of a random graph, and if it approaches -1, then the characteristics of a lattice. Values ω around 0 indicate that the network is a small-world network. An alternative small-world measurement metric is the σ proposed in [19]. That metric has several disadvantages, described in [17], so the ω metric has been used to analyze the simulated network.

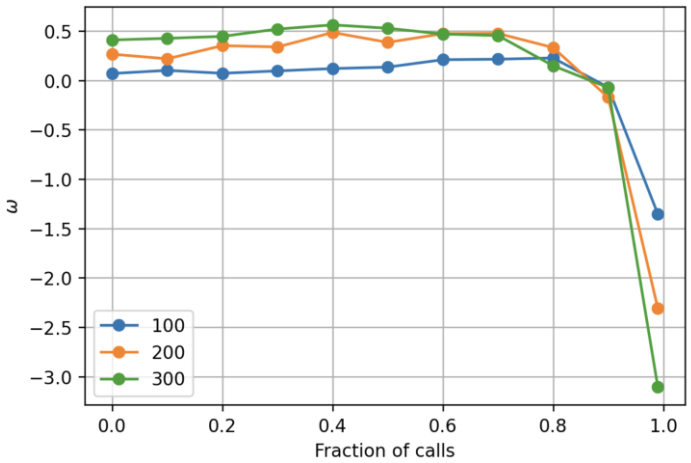


Fig.10 Dependence of ω on the number of subscribers in the network and the fraction of contacts within contact lists

Fig. 10 shows the dependence of ω on the share of contacts and the number of subscribers in the network. Let us reiterate that the proportion of contacts determines the degree of call and, accordingly, at low values ω the network resembles a random graph. In our case, for networks with a fraction of contacts up to 0.80, the values of the coefficient ω are in the interval [0.05, 0.55], which characterizes such networks as random graphs. In a dynamic network of telephone subscribers with fraction values in the range [0.80, 0.90], the coefficient ω crosses the value 0; as a result, those networks show small-world network characteristics. At higher values of the fraction of calls within the contact list, the network has the properties of a regular lattice.

6. Conclusions

In this work, a dynamic network of telephone subscribers has been modeled and its features have been analyzed in relation to the number of subscribers and the fraction of calls made within subscribers' contact lists. The presence of subscribers' contact list, which usually includes family members, colleagues, friends, and the closeness of their connections determines the topology of the network and the dynamics of its properties. Subscribers connected by contact lists and the connections between them are clearly visualized with a high fraction of contacts between them. When the fraction of calls within contact lists is low, the network is transformed into a random graph. As a result of the analysis of the network properties, it has been revealed that the distribution of node degrees corresponds to the lognormal law with parameters μ and σ. The distribution parameters do not depend on the number of subscribers in the network but depend on the fraction of contacts – the higher the fraction, the smaller the parameter μ, and, accordingly, the smaller the value of degrees. The parameter σ is equal to 0.27 and is independent of both the subscriber number and the call fraction. The number of links depends linearly on the number of subscribers – the higher the fraction, the fewer links are created with the appearance of additional subscribers (1). Thus, with a fraction of 0.9, a new subscriber creates on average 8.3 links with other subscribers, and with a share of 0.7, 15 links are created. An increase in the number of subscribers affects the network density, reducing it according to the hyperbolic law (2). As the fraction of contacts increases, the density decreases, since an increasing proportion of links are created among a limited number of subscribers. The clustering coefficient, just like the density, changes according to a hyperbolic law. The parameters of the hyperbolic law have a complex dependence on the fraction of contacts presented graphically in Fig. 8. The average value of the shortest path length is well approximated by a logarithmic function to base 11 and 7 for networks with a fraction of calls within contact list more than 0.80, which indicates the properties of small-world network. Additionally, the presence of small-world properties has been assessed by using the coefficient ω (4) and it has been revealed that a dynamic network of telephone subscribers demonstrates such properties when the fraction of contacts within the contact list is in the interval [0.80, 0.90].

It should be noted that experiments for all networks from 10 to 1000 subscribers have been carried out over the same period. This limitation affects the number of links created between subscribers and, accordingly, in networks with a small number of subscribers, all possible links can be established over a given period, and with a large number, only a small fraction of links can be formed. This model could be

used for the further research on the effects of abnormal users (spammers) on telephone subscriber network properties, information propagation speed, and its relationship to contact list sizes and call share.

REFERENCES

1. M. Newman, Networks, vol. 1. Oxford University Press, 2018. DOI: https://doi.org/10.1093/oso/9780198805090.001.0001 (Last accessed: 20.09.2023).
2. Skarding J, Gabrys B, Musial K (2021) Foundations and modelling of dynamic networks using Dynamic Graph Neural Networks: A survey. IEEE Access 9:79143–79168. DOI: https://doi.org/10.1109/ACCESS.2021.3082932 (Last accessed: 20.09.2023).
3. Farine DR (2018) When to choose dynamic vs. static social network analysis. Journal of Animal Ecology 87:128–138. DOI: https://doi.org/10.1111/1365-2656.12764 (Last accessed: 20.09.2023).
4. Toivonen R, Kovanen L, Kivelä M, et al (2008) A comparative study of social network models: network evolution models and nodal attribute models. DOI: https://doi.org/10.1016/j.socnet.2009.06.004 (Last accessed: 20.09.2023).
5. Onnela J-P, Saramäki J, Hyvönen J, et al (2007) Structure and tie strengths in mobile communication networks. Proc Natl Acad Sci USA 104:7332–7336. DOI: https://doi.org/10.1073/pnas.0610245104
6. Seshadri M, Machiraju S, Sridharan A, et al (2008) Mobile call graphs: beyond power-law and lognormal distributions. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Las Vegas Nevada USA, pp 596–604. DOI: https://doi.org/10.1145/1401890.1401963 (Last accessed: 20.09.2023).
7. Nanavati AA, Gurumurthy S, Das G, et al (2006) On the Structural Properties of Massive Telecom Call Graphs: Findings and Implications. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management. Association for Computing Machinery, New York, NY, USA, pp 435–444. DOI: https://doi.org/10.1145/1183614.1183678 (Last accessed: 20.09.2023).
8. Hidalgo CA, Rodriguez-Sickert C (2008) The dynamics of a mobile phone network. Physica A: Statistical Mechanics and its Applications 387:3017–3024. DOI: https://doi.org/10.1016/j.physa.2008.01.073 (Last accessed: 20.09.2023).
9. Bardoscia M, Barucca P, Battiston S, et al (2021) The Physics of Financial Networks. Nat Rev Phys 3:490–507. DOI: https://doi.org/10.1038/s42254-021-00322-5 (Last accessed: 20.09.2023).
10. Muzio G, O'Bray L, Borgwardt K (2021) Biological network analysis with deep learning. Briefings in Bioinformatics 22:1515–1530. DOI: https://doi.org/10.1093/bib/bbaa257 (Last accessed: 20.09.2023).
11. Pichitlamken J, Deslauriers A, L'Ecuyer P, Avramidis AN (2003) Modelling and simulation of a telephone call center. In: Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693). IEEE, New Orleans, LA, USA, pp 1805–1812. DOI: 10.1109/WSC.2003.1261636 (Last accessed: 20.09.2023).
12. Niaki S, B. Rad Z (2004) Designing a communication network using simulation. Scientia Iranica 11:165–180. URL: https://www.researchgate.net/publication/236107639_Designing_a_communication_network_using_simulation (Last accessed: 20.09.2023).
13. Lehmann S (2019) Fundamental Structures in Dynamic Communication Networks. DOI: https://doi.org/10.1007/978-3-030-23495-9_2 (Last accessed: 20.09.2023).
14. Danilevskyi M, Yanovsky V (2023) Modeling and Analyzing the Simplest Network of Telephone Subscribers. Bulletin of VN Karazin Kharkiv National University, series «Mathematical modeling Information technology Automated control systems» 59:6–15. DOI: https://doi.org/10.26565/2304-6201-2023-59-01 (Last accessed: 20.09.2023).
15. Danilevskiy V, Yanovsky V (2020) Statistical properties of telephone communication network. arXiv preprint arXiv:200403172. URL: https://arxiv.org/pdf/2004.03172.pdf (Last accessed: 20.09.2023).
16. Watts D, Strogatz S (1998) Collective dynamics of 'small-world' networks. Nature 393:440–442. DOI: https://doi.org/10.1038/30918 (Last accessed: 20.09.2023).

17. Telesford QK, Joyce KE, Hayasaka S, et al (2011) The Ubiquity of Small-World Networks. Brain Connectivity 1:367–375. DOI: https://doi.org/10.1089/brain.2011.0038 (Last accessed: 20.09.2023).
18. Simone A, Ridolfi L, Berardi L, et al Complex Network Theory for Water Distribution Networks Analysis. pp 1971–1962 URL: https://www.researchgate.net/publication/329323388_Complex_Network_Theory_for_Water_Distribution_Networks_analysis (Last accessed: 20.09.2023).
19. Humphries MD, Gurney K, Prescott TJ (2006) The brainstem reticular formation is a small-world, not scale-free, network. Proc R Soc B 273:503–511. DOI: https://doi.org/10.1098/rspb.2005.3354 (Last accessed: 20.09.2023).

Данілевський Михайло Вікторович *аспірант; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022*

Яновський Володимир Володимирович *доктор фізико-математичних наук, професор, професор кафедри штучного інтелекту та програмного забезпечення Харківський національний університет імені В. Н. Каразіна, майдан Свободи 4, Харків-22, Україна, 61022 Завідувач теоретичним відділом, інститут монокристалів НАН України, пр.Науки 60, Харків, Україна, 61001*

Маций Ольга Борисівна *доцент; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022*

Моделювання та аналіз динамічної мережі телефонних абонентів з урахуванням ступеня зв'язаності засобами контакт-листів

Актуальність. Моделювання складних динамічних мереж, компоненти яких взаємодіють і розвиваються з часом, має важливе значення, оскільки дозволяє розуміти та прогнозувати їхню поведінку. Це допомагає оптимізувати продуктивність, підвищувати стійкість та ефективно управляти ресурсами у технологічних, інформаційних, соціальних та біологічних мережах.
Мета. Метою роботи є моделювання динамічної мережі телефонних абонентів та виявлення основних властивостей мережі. Головна увага зосереджена на експериментах з розробленою моделлю та визначення залежностей властивостей мережі за даними моделювання.
Методи дослідження. В роботі використовувалися методи побудови комп'ютерних моделей, методи аналізу параметрів мереж, метод найменших квадратів і метод Монте-Карло стохастичної динаміки дискретних станів з використанням тимчасових кроків однакової довжини. Комп'ютерна модель розроблена мовою Python із використанням бібліотек Pandas, Numpy та NetworkX.
Результати. Запропоновано модель динамічної мережі телефонних абонентів з імітацією контакт-листів, до яких зазвичай входять члени сім'ї, колеги, друзі. Проведено експерименти з моделлю та досліджено залежності властивостей мережі від кількості абонентів та частки контактів у рамках контакт-листів. Визначено при яких значеннях параметрів моделі мережа, що отримується в результаті моделювання, має властивості мережі тісного світу.
Висновки. Запропонована модель динамічної мережі телефонних абонентів з імітацією контакт-листів дозволила виявити залежності властивостей мережі від кількості абонентів та частки контактів у рамках контакт-листів. Виявлено, що розподіл ступенів вершин відповідає логнормальному закону. Кількість зв'язків лінійно залежить від кількості абонентів, причому чим вища частка контактів, тим менше зв'язків створюється з появою нового абонента. Збільшення кількості абонентів впливає на щільність мережі та знижує її за гіперболічним законом. У разі підвищення частки контактів щільність знижується, оскільки дедалі більша частина зв'язків створюється серед обмеженої кількості абонентів. Коефіцієнт кластеризації так само, як і щільність змінюється за гіперболічним законом. Середнє значення найкоротшого шляху за певних параметрів мережі добре апроксимується логарифмічною функцією за частки контактів більше 0.80. Коефіцієнт ω показує, що при частці контактів в межах контакт-листів в інтервалі [0.80, 0.90] змодельована мережа телефонних абонентів має властивості мережі тісного світу.

Ключові слова: *складна динамічна мережа, граф мобільних викликів, телефонна мережа, логнормальний розподіл, розподіл ступенів, щільність мережі, коефіцієнт кластеризації, середня довжина найкоротшого шляху, мережа тісного світу.*

Надійшла у першій редакції 07.04.2024, в останній - 11.05.2024.
The first version has been received on 07.04.2024, the final version - on 11.05.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-03

УДК (UDC) 004,94

Єгоров Єгор Сергійович
магістр
Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022
e-mail: ees010301@gmail.com
https://orcid.org/0000-0001-8731-7198

Шматков Сергій Ігорович
д.т.н., професор; завідуючий кафедри теоретичної і прикладної системотехніки факультету комп'ютерних наук;
Харківський національний університет імені В. Н. Каразіна, майдан Свободи,4, Харків, Україна, 61022
e-mail: s.shmatkov@karazin.ua
https://orcid.org/0000-0002-0298-7174

Розробка моделі нейронної мережі для усунення неоднозначності омографів у текстових даних

У цій науковій статті досліджується створення моделі нейронної мережі, спрямованої на вирішення неоднозначності омографів у текстових даних. Різні типи нейронних мереж ретельно вивчаються на предмет їхнього потенціалу у вирішенні цієї проблеми. Стаття заглиблюється в методологічні аспекти архітектури нейронної мережі, що включає аналіз, проектування, реалізацію, тестування, оцінку та оптимізацію. Важливість кожного етапу цього процесу наголошується на важливості глибокого розуміння типів нейронних мереж та їх застосування, а також розумного вибору технології. Корисність розробленої моделі поширюється на домени, що використовують автоматичне розпізнавання мови, підтримку прийняття рішень на основі тексту, вдосконалення систем пошуку та обробку природної мови. Дослідники та практики в області обробки природної мови та класифікації текстових даних знайдуть цю статтю цінною.

Актуальність: у сучасному світі розробка моделі нейронної мережі для вирішення неоднозначності омографів у текстових даних визначається викликами, які стоїть перед галуззю обробки природної мови. Ця модель здатна виправляти помилки, пов'язані з невірним розумінням значень слів у тексті, що забезпечить більшу точність та якість аналізу текстового матеріалу. Використання її можливе в різних сферах, включаючи автоматичне визначення мови, підтримку прийняття рішень на основі текстової інформації, удосконалення систем пошуку та класифікації даних.
Мета: підвищення якості обробки тексту, зокрема, поліпшення точності розпізнавання слів, які мають більше одного значення, за допомогою розробки та впровадження нейронної мережі, яка буде мати можливість усувати неоднозначності омографів у текстових даних в реальному часі.

Методи дослідження: для дослідження обраної області було використано методи системний аналіз, методи глибокого навчання, теорія нейронних мереж, методи обробки та підготовки даних, імітаційне моделювання. Програмне забезпечення розроблено за допомогою мови Python і використані пакети sklearn, keras і т.д..

Результати: головним результатом роботи була розробка моделі нейронні мережі, яка усуває неоднозначності омографів у текстових даних в реальному часі, що дає можливість розвитку її на інших мовах.
Висновки: розглянуто проблему неоднозначності омографів у тектстових даних. Оскільки це завдання обробки природної мови була розроблена модель нейронної мережі з довгою короткостроковою пам'яттю з використанням ембедінгових моделей, LSTM-шару та повнозв'язаних шарів. Дослідження засвідчило важливість інноваційних підходів у вирішенні проблем неоднозначності омографів у текстових даних, а використання нейронних мереж та технологій штучного інтелекту стає обіцяючим напрямком для подальших досліджень та впроваджень у цій області.
Ключові слова: *нейронні мережі, омограф, модель, ембедінг, PYTHON, LSTM.*

Як цитувати: Єгоров Є. С., Шматков С. І. Розробка моделі нейронної мережі для усунення неоднозначності омографів у текстових даних. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2024. вип. 62. С.30-36. https://doi.org/10.26565/2304-6201-2024-62-03

How to quote: Yehorov Y., Shmatkov S "Development of a neural network model to resolve homograph ambiguity in text data", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp. 30-36, 2024. [In Ukrainian] https://doi.org/10.26565/2304-6201-2024-62-03

1. Вступ

У сучасному інформаційному суспільстві, в якому величезні об'єми текстової інформації обробляються щоденно, розробка ефективних інструментів для автоматичної обробки та аналізу текстових даних стає надзвичайно важливою задачею. Однак існують різні лексичні неоднозначності, які ускладнюють цей процес, і однією з основних проблем є неоднозначність омографів — слів, що мають однакову лексичну форму, але різне значення.

2. Обґрунтування та вибір типу нейронної мережі з довгою короткостроковою пам'яттю (LSTM)

a) **Обробка послідовностей:** LSTM мережі ефективно працюють з послідовностями даних, такими як текст, часові ряди тощо. Їхній механізм довгої пам'яті дозволяє зберігати та використовувати інформацію з попередніх кроків послідовності, що робить їх ідеальними для прогнозування на основі історичних даних.

b) **Вирішення проблеми зникаючого градієнту:** LSTM мережі мають механізми воротного контролю, що дозволяють ефективно управляти градієнтами у процесі зворотного розповсюдження помилки. Це допомагає уникнути проблеми зникаючого градієнту та дозволяє тренувати глибокі мережі більш ефективно.

c) **Управління довгостроковою залежністю:** LSTM здатні ефективно вирішувати завдання, де потрібно управляти довгостроковими залежностями між даними. Це особливо корисно у задачах, де важливо враховувати контекст на більш довгій відстані в часі або послідовності.

d) **Гнучкість в архітектурі:** LSTM мережі можуть бути модифіковані та адаптовані для різних завдань, включаючи класифікацію, прогнозування та генерацію послідовностей. Вони можуть бути використані як частина більш складних архітектур або вбудовані в інші моделі, що робить їх універсальними для різноманітних застосувань (рис. 1). [1-2]

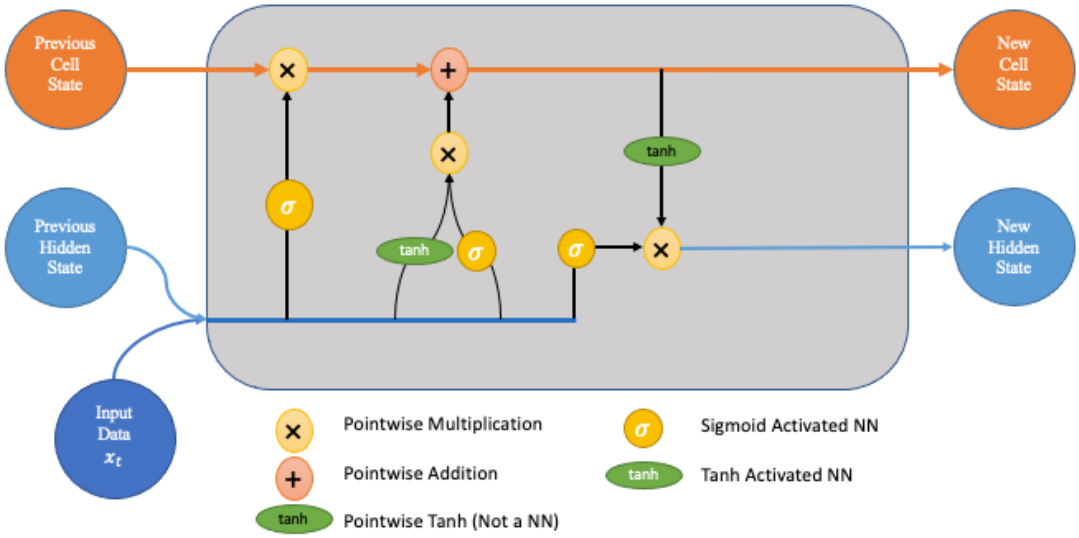


Рисунок 1 – Діаграма LSTM

3. Архітектура нейронної мережі (рис. 2)

- Input Layer (Вхідний шар): цей шар відображає вхідні дані в модель. Даним вхідним шаром є Embedding з параметрами, які визначають розмір словника (vocab_size), розмірність вектора ембедінгу (embedding_dim) і довжину вхідної послідовності (input_length).
- Embedding Layer (Шар ембедінгу): цей шар використовується для перетворення індексів слів у вектори ембедінгу. Вектор ембедінгу - це числовий вектор, який представляє слово у просторі високої розмірності. Величина input_length вказує на максимальну довжину вхідних послідовностей.
- LSTM Layer (Шар LSTM): шар довготривалої короткострокової пам'яті (LSTM) використовується для моделювання довготривалих залежностей в послідовностях.

Він приймає вхід від шару ембедінгу і генерує внутрішній стан, який передається наступним шарам.

- Dense Layer (Повнозв'язаний шар): цей шар використовується для класифікації або регресії. Для вирішення поставленої задачі використовується функція активації softmax для визначення ймовірностей кожного класу.[3]
- Output Layer (Вихідний шар): шар визначає вихідні ймовірності для різних класів у задачі класифікації.
- Зв'язки між шарами: стрілки між шарами представляють потік даних від одного шару до іншого. Вони також вказують напрямок передачі інформації. Враховуючи вищезазначені фактори, обрана архітектура LSTM забезпечує ефективневиявлення та усунення неоднозначності омографів у текстових даних, здатність адаптуватися до різних контекстів та уникнення проблем зниклих градієнтів під час тренування.[4-5]

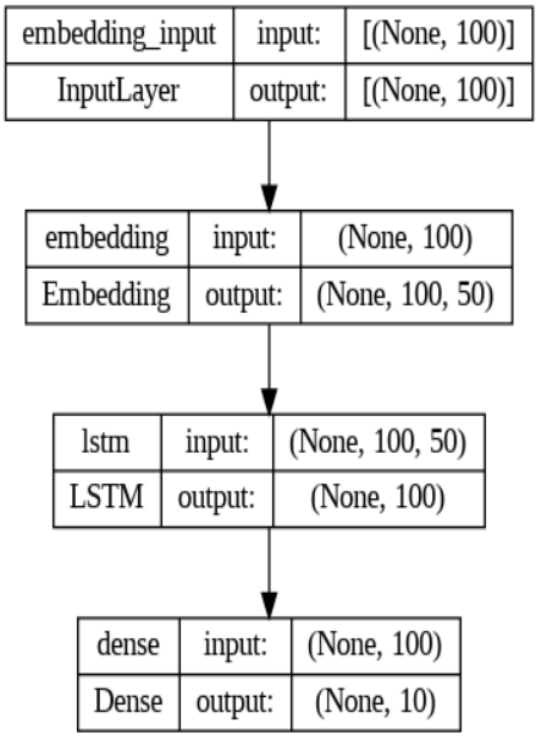


Рисунок 2 – Модель нейронної мережі для усунення неоднозначності омографів у текстових даних

4. Навчання нейронної мережі, та тестування

Кроки:

- Попередня обробка:

  Очищення даних: Видалення непотрібної або неінформативної інформації, такої як зайві символи, пробіли, знаки пунктуації. Це допомагає у покращенні якості даних та ефективності моделі.

  Токенізація: Розбиття тексту на окремі "токени" або слова. Це необхідно для подальшого представлення тексту у вигляді числових векторів, зрозумілих для моделі.

  Лематизація та стемінг: Зменшення слова до його базової форми (леми) або кореневої форми (стему). Це допомагає моделі у виборі ключових слов для розпізнавання семантики.[6]
- Векторизація тексту:

  Цей процес перетворює слова і токени в числові представлення, що використовуються для подальшого навчання моделі.
- Ініціалізація:

  Визначає початкові параметри моделі та її компонентів.
- Етап оптимізації:

  За допомогою оптимізатора Adam проходить процес адаптації ваг моделі під

конкретні дані.

- Етап форвардного та зворотнього поширення:

  Форвардне поширення: вхідні дані подаються в модель, де вони проходять через різні шари та отримують прогнозовані значення. Після цього відбувається обрахунок втрат та початок зворотного поширення.

  Зворотнє поширення: обчислення градієнтів втрат відносно параметрів моделі. Ці градієнти використовуються для актуалізації ваг шарів у відповідності із заданими правилами оптимізації. Цей процес дозволяє моделі "вчитися" з входов та виправляти помилки, підлаштовуючи параметри.

- Оцінка:

  Валідація та тестування: Оцінка моделі на валідаційному наборі для налаштування параметрів та на тестовому наборі для оцінки загальної ефективності.
  Метрики ефективності: Використання відповідних метрик для вимірювання точності,чутливості, специфічності та інших параметрів продуктивності[7].

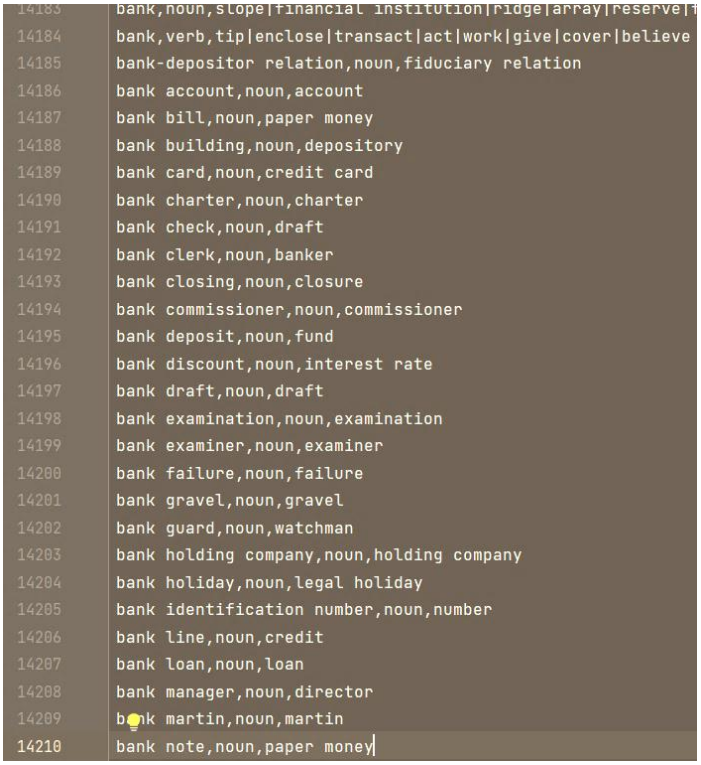


Рисунок 3 – Вигляд у якому зберігаються дані

Проведений експеримент з різними конфігураціями (табл. 1) моделі показав, що варіант (друга колонка) з параметрами LSTM = 64, Dense = 128, batch_size = 64, Epochs = 10, Adam = 0.001 демонструє найвищі значення обох метрик - високу точність (Accuracy = 0.8802) та значення AUC-ROC (0.8916). Це свідчить про те, що ця конфігурація моделі ефективно вирішує задачу класифікації омографів, надаючи найкращі результати у порівнянні з іншими варіантами.

Також важливо зазначити, що збільшення кількості LSTM-шарів не завжди призводить до поліпшення результатів, а зменшення кількості може вплинути на ефективність моделі.

Збільшення кількості епох не завжди призводить до покращення, а швидкість навчання може виявитися ключовим фактором: збільшення швидкості може призвести до зростання точності таAUC-ROC.

Таблиця 1 - Результати експериментів з різними параметрами моделі

	LSTM = 64 Demes = 128 Batch_size = 64 Epochs = 10 Adam = 0.001	LSTM = Demes = 128 Batch_size = 64 Epoch Adam = 0.001	LSTM = 32 Demes = 64 Batch_size = 64 Epochs = 10 Adam = 0.001	LSTM = 128 Demes = 256 Batch_size = 64 Epochs = 5 Adam = 0.01	LSTM = 64 Demes = 256 Batch_size = 64 Epochs = 10 Adam = 0.01	LSTM = 64 Demes = 128 Batch_size = 64 Epochs = 10 Adam = 0.0001
Accuracy						
AUC-ROC						

Навчання моделі пройшло досить успішно. Починаючи з високого значення точності на першій епосі, помітно певний зріст наступних епох, що свідчить про те, що модель ефективно вчиться на навчальних даних. Однак, можливо, варто звернути увагу на те, що на останніхепохах можна спостерігати певний спад у точності та інших метриках на валідаційному наборі. Це може бути ознакою перенавчання моделі або необхідності підгонки гіперпараметрів.

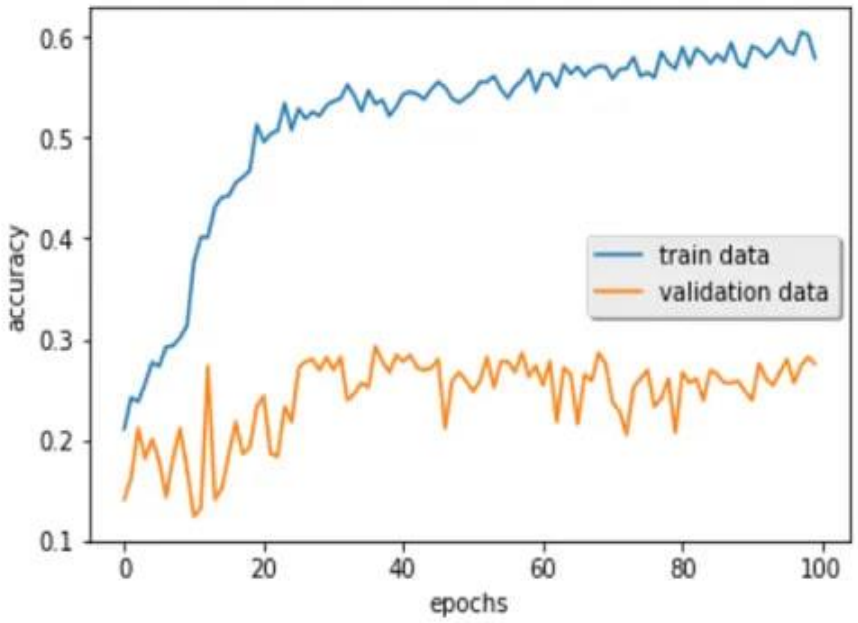


Рисунок 4 – Графік класової точночті моделі

Результати та висновки

Модель демонструє високий рівень ефективності та продуктивності завдяки розробленій архітектурі, різноманітності та якості даних для навчання, а також обраним гіперпараметрам. Архітектура включає LSTM-шари, ембедінги та дропаут-шари, які ефективно працюють з текстовими даними, що підтверджується високою точністю 89% та значенням AUC-ROC 0.8916. Навчальні дані складаються з 10 тисяч текстових даних з датасету, попередньо оброблених шляхом видалення HTML-сутностей, URL-адрес та інших непотрібних символів. Проведені експерименти та аналіз результатів дозволили визначити оптимальні гіперпараметри моделі.

Модульна структура та архітектура програмної реалізації також забезпечують легку інтеграцію та адаптацію моделі для використання на різних платформах. Наприклад, модель була успішно

впроваджена у середовищі Telegram, що дозволяє в реальному часі проводити аналіз і усувати неоднозначності омографів у текстових даних, які надходять через обрану платформу.

СПИСОК ЛІТЕРАТУРИ

1. Saiful Islam, Md.; Hossain, Emam (2020-10-26). "Foreign Exchange Currency Rate Prediction using a GRU-LSTM Hybrid Network":
https://www.sciencedirect.com/science/article/pii/S2666222120300083?via%3Dihub (Last accessed: 21.05.2024).
2. Fully Connected Layers in Convolutional Neural Networks:
https://indiantechwarrior.com/fully-connected-layers-in-convolutional-neural-networks/ (Last accessed: 21.05.2024).
3. Стрілець В. Є., Шматков С. І., Угрюмов М. Л. та ін. – Харків, Методи машинного навчання монографія, Харківський національний університет імені В. Н. Каразіна, 2020.
4. Zell, Andreas (1994). Simulation Neuronaler Netze [Simulation of Neural Networks] (in German) (1st ed.). Addison-Wesley. p. 73
5. McCrae, John P.; Labropoulou, Penny; Gracia, Jorge; Villegas, Marta; Rodríguez-Doncel, Víctor; Cimiano, Philipp (2015). "One Ontology to Bind Them All: The META-SHARE OWL Ontology for the Interoperability of Linguistic Datasets on the Web". In Gandon, Fabien; Guéret, Christophe; Villata, Serena; Breslin, John; Faron-Zucker, Catherine; Zimmermann, Antoine (eds.). The Semantic Web: ESWC 2015 Satellite Events. Lecture Notes in Computer Science. Vol. 9341. Cham: Springer International Publishing. pp. 271–282.
6. Kilgarriff,A.:Getting to know your corpus. In: Text, Speech and Dialogue, Springer (2012)3–15
7. "Deep Learning" Ian Goodfellow, Yoshua Bengio, Aaron Courville, 2016, MIT Press

REFERENCES

1. Saiful Islam, Md.; Hossain, Emam (2020-10-26). "Foreign Exchange Currency Rate Prediction using a GRU-LSTM Hybrid Network":
https://www.sciencedirect.com/science/article/pii/S2666222120300083?via%3Dihub (Last accessed: 21.05.2024).
2. Fully Connected Layers in Convolutional Neural Networks:
https://indiantechwarrior.com/fully-connected-layers-in-convolutional-neural-networks/ (Last accessed: 21.05.2024).
3. Strelets V. E., Shmatkov S. I., Ugryumov M. L. and others. – Kharkiv, Methods of machine learning monograph, V. N. Karazin Kharkiv National University, 2020. [in Ukrainian]
4. Zell, Andreas (1994). Simulation Neuronaler Netze [Simulation of Neural Networks] (in German) (1st ed.). Addison-Wesley. p. 73
5. McCrae, John P.; Labropoulou, Penny; Gracia, Jorge; Villegas, Marta; Rodríguez-Doncel, Víctor; Cimiano, Philipp (2015). "One Ontology to Bind Them All: The META-SHARE OWL Ontology for the Interoperability of Linguistic Datasets on the Web". In Gandon, Fabien; Guéret, Christophe; Villata, Serena; Breslin, John; Faron-Zucker, Catherine; Zimmermann, Antoine (eds.). The Semantic Web: ESWC 2015 Satellite Events. Lecture Notes in Computer Science. Vol. 9341. Cham: Springer International Publishing. pp. 271–282.
6. Kilgarriff,A.:Getting to know your corpus. In: Text, Speech and Dialogue, Springer (2012)3–15
7. "Deep Learning" Ian Goodfellow, Yoshua Bengio, Aaron Courville, 2016, MIT Press

Yehorov Yehor *Master student; V.N. Karazin Kharkiv National University, Svobody Square, 4, Kharkiv-22, Ukraine, 61022.*
e-mail: ees010301@gmail.com
https://orcid.org/0000-0001-8731-7198

Shmatkov Segii *Professor; V.N. Karazin Kharkiv National University, Svobody Square, 4, Kharkiv-22, Ukraine, 61022.*
e-mail: s.shmatkov@karazin.ua
https://orcid.org/0000-0002-0298-7174

Development of a neural network model to resolve homograph ambiguity in text data

This paper explores the creation of a neural network model aimed at resolving the ambiguity of homographs in textual data. Various neural network types are scrutinized for their potential in addressing this challenge. The paper delves into the methodological aspects of neural network architecture, encompassing analysis, design, implementation, testing, evaluation, and optimization. Each stage of this process is underlined for its significance, stressing the importance of a thorough understanding of neural network types and their applications, as well as judicious technology selection. The utility of the developed model extends to domains utilizing automatic language recognition, text-based decision support, enhancement of search systems, and natural language processing. Researchers and practitioners in the field of natural language processing and text data classification will find this paper valuable.

Relevance: in today's world, the development of a neural network model to resolve the ambiguity of homographs in textual data is determined by the challenges facing the field of natural language processing. This model is able to correct errors associated with incorrect understanding of the meanings of words in the text, which will ensure greater accuracy and quality of the analysis of the text material. Its use is possible in various areas, including automatic language detection, decision support based on text information, improvement of search systems and data classification.

The goal: to improve the quality of text processing, in particular, to improve the accuracy of recognition of words that have more than one meaning, through the development and implementation of a neural network that will be able to resolve homograph ambiguities in text data in real time.

Research methods: system analysis, deep learning methods, neural network theory, data processing and preparation methods, simulation modeling were used to study the selected area. The software is developed using the Python language and uses the sklearn, keras and other packages.

Results: the main result of the work is the development of a neural network model that eliminates homograph ambiguities in text data in real time, which makes it possible to expand it for the other languages.

Conclusions: the problem of ambiguity of homographs in textual data has been considered. For this natural language processing task, a neural network model with long short-term memory was developed using embedding models, LSTM layer, and fully connected layers. The study proves the importance of innovative approaches in solving homograph ambiguity problems in textual data, and that the use of neural networks and artificial intelligence technologies becomes a promising direction for further research and implementation in this area.

***Keywords**: neural networks, homograph, model, embedding, PYTHON, LSTM*

Надійшла у першій редакції 03.04.2024, в останній -17.05.2024.
The first version has been received 3.04.2024, the final version – on 17.05.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-04

УДК (UDC) 004.8

Конюхов Владислав Дмитрович *аспірант*
Інститут проблем машинобудування ім. А.М.Підгорного НАН України
адреса: вул. Пожарського 2/10, Харків, 61046, Україна
e-mail: riggellll@gmail.com https://orcid.org/0009-0007-0256-1388

Моргун Олег Миколайович *к.ф.-м.н., директор*
ТОВ «Лабораторія рентгенівської медичної техніки»
адреса: вул. Достоєвського, 1, м. Харків, 61102, Україна
e-mail: lrmt@ukr.net https://orcid.org/0009-0005-6157-9110

Нємченко Костянтин Едуардович *д.ф.-м.н., завідувач кафедри*
Харківський національний університет імені В. Н. Каразіна
майдан Свободи, 4 м. Харків, 61022, Україна
e-mail: nemchenko@karazin.ua
https://orcid.org/0000-0002-0734-942X

Багаторазове навчання нейронних мереж для автоматичної сегментації області хребта

Актуальність. Профілактичні та діагностичні дослідження наявності кісткових захворювань, потребують морфометричних досліджень рентгенівських знімків відділу грудної клітини. В теперішній час для розв'язування таких задач все частіше використовують методи штучного інтелекту. Зокрема в цій роботі досліджена можливість застосування штучного інтелекту при сегментації медичних зображень з ціллю автоматичного діагностування захворювань кісткової системи людини. Основні труднощі цієї задачі пов'язані з тим, що реальні рентгенівські зображення мають граничні характеристики якості, наприклад, за параметром відношення сигналу до шуму або контрасту. З цієї причини застосування стандартних методів розпізнавання зображень або автоматичної діагностики стає неможливим. Ці труднощі привели до того, що в теперішній час існує достатньо велика кількість робіт в цій галузі, але результати більшості з них демонструють недостатні для практичного використання результати. Наведена в цій роботі оригінальна методика використання ансамблю нейронних мереж дозволила узагальнити здобуті дотепер результати та поєднати переваги інших підходів.
Мета. Дослідити можливість використання штучного інтелекту в сегментації медичних зображень з метою автоматичної діагностики захворювань кісткової системи людини.
Методи дослідження. У даному досліджені використовувався ансамблевий метод сегментації рентгенівських зображень. Основу навчальних даних було створено на базі рентгенівських знімків узятих з відкритих джерел. Всього кількість зображень становила 183 знімків. Початкові дані було модифіковано згідно з вимогами необхідними для навчання моделей. Всі зображення були переведені до градації сірого та змінені до розмірів 256x256 пікселів.
Результати. Завдяки використанню цього методу в двох тестових випадках було отримано покращення точності з 0,543 до 0,820 для першого знімку та з 0,725 до 0,923 для другого знімку.
Висновки. Було запропоновано і досліджено застосування методики використання ансамблю нейронних мереж, що багаторазово навчаються, для автоматичної сегментації певної області хребта, а саме ділянки хребта Th8-Th11.Застосування цього методу дозволило отримати більш стабільні та точні передбачення для шуканих ділянок хребта, навіть для знімків з високим рівнем шуму.

Ключові слова: *штучний інтелект, машинне навчання, розпізнавання зображень, нейронна мережа, ансамбль нейронних мереж, морфометрія.*

Як цитувати: Конюхов В. Д., Моргун О. М., Нємченко К.Е. Багаторазове навчання нейронних мереж для автоматичної сегментації області хребта. *Вісник Харківського національного університету імені В.Н.Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2024. вип. 62. С.37-44.
https://doi.org/10.26565/2304-6201-2024-62-04

How to quote: V.D. Koniukhov, O.M. Morgun and K.E. Nemchenko, "Multiple training of neural networks for automatic spine segmentation". *Bulletin of V.N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp. 37 - 44, 2024. [In Ukrainian] https://doi.org/10.26565/2304-6201-2024-62-04

1 Вступ

Діагностика та оцінка наявності кісткових захворювань, зокрема компресійних переломів , потребує морфометричних досліджень бічних рентгенівських знімків відділу грудної клітини (ВГК) [1], які вимагають від лікаря-рентгенолога дуже тривалої і трудомісткої роботи. Для полегшення цієї роботи, тобто автоматичної сегментації за допомогою комп'ютерної програми,

раніше розроблялася значна кількість різноманітних алгоритмів [2 – 4]. Проте труднощі реалізації цих алгоритмів пов'язані з низкою причин. При рентгенологічних дослідженнях повинен виконуватися принцип ALARA (*As Low As Reasonably Achievable*), тобто доза опромінення пацієнта має бути настільки низькою, наскільки це можливо. Це призводить до низького відношення сигнал/шум, де під сигналом розуміється радіаційний контраст, а під шумом квантові та структурні шуми. Структурний шум – це шум, який пов'язаний з накладенням на область хребта інших анатомічних органів при проектуванні тривимірного об'єкта (людського тіла) на двовимірний приймач рентгенівського зображення. Це проектування призводить також до розмиття границь зображення хребта, що відповідно погіршує його виявлення та сегментацію. У разі стандартного підходу до застосування, навіть спеціалізованих для медицини нейронних мереж (наприклад U-Net або аналогів) [5], виникають проблеми при сегментації реальних зашумлених і розмитих рентгенівських зображень бічної проекції ВГК. Для вирішення задачі автоматичної сегментації ділянки хребта навіть у таких зашумлених і розмитих рентгенівських знімках у цій статті запропоновано метод багаторазового навчання нейронних мереж.

Останнім часом завдяки прогресу в розвитку комп'ютерної техніки та алгоритмів штучного інтелекту з'явилася велика кількість досліджень з використанням згорткових нейронних мереж для автоматичної сегментації хребта [6 – 8]. Практично всі вони пов'язані з розробкою алгоритмів удосконалення стандартного застосування нейронних мереж до зображень хребта. До таких алгоритмів можна віднести гібридні методи, які поєднують раніше розроблені алгоритми машинного навчання та використання згорткових нейронних мереж з глибоким навчанням [7], або використання багатозадачних мультимодальних нейронних мереж [6].

У цій статті пропонується і досліджується застосування методики використання ансамблю нейронних мереж, що багаторазово навчаються, для автоматичної сегментації певної області хребта, а саме ділянки хребта Th8-Th11.

2 Методи

Задача автоматичної сегментації ділянки хребта з хребцями Th8-Th11 вирішувалося у два етапи, що дозволило досягти достатньо високого рівня визначення шуканої області.

2.1 Перший етап

На першому етапі , первинного передбачення шуканої області , досліджувалися повноформатні рентгенівські знімки бічної проекції ВГК людини. Процедура застосування нейронних мереж для визначення даної частини хребта була стандартною. Спочатку вибирався набір зображень за якими проводилося навчання нейронних мереж. В якості таких наборів , використовувалися загальнодоступні набори , які знаходяться у мережі Інтернет [10]. Потім вручну створювалися маски областей, тобто областей які містять шукану ділянку з хребцями Th8-Th11. Приклади вихідних зображень та відповідні маски показані на Рис. 2.1. У нашому випадку навчання всіх моделей нейронних мереж проводилося на наборі зі 183 знімків, які масштабувалися до розміру 256*256 пікселів. Усі зображення були виконані в градаціях сірого в діапазоні інтенсивності сигналу 0-255.

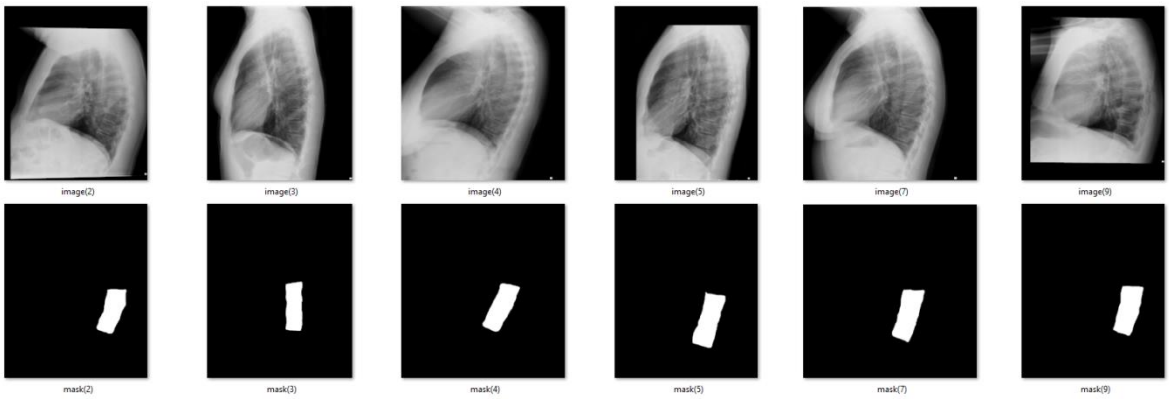


Рис.2.1 Приклади вхідних зображень та відповідні маски

Навчання здійснювалося на наступних нейронних мережах: FCN8, FCN32, MobileNet_Segnet, U-Net, MobileNet_Unet, VGG_Segnet. При цьому параметри навчання вибиралися однакові. Типовий результат передбачення - вихідний рентгенівський знімок, відповідна йому маска і результат визначення ділянки хребта, що шукається, показано на Рис.2.2.

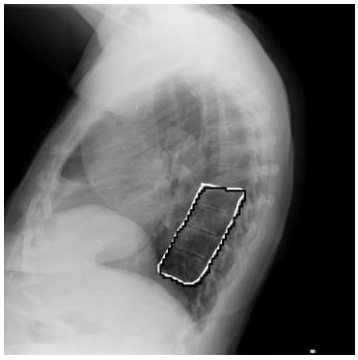


Рис.2.2 Приклад вихідного рентгенівського знімку, відповідна йому маска шуканої області (білий контур) та результат передбачення (чорний контур)

Відповідний цій моделі нейронної мережі графік втрат, а також графік зростання якості визначення шуканої області представлені на Рис.2.3.

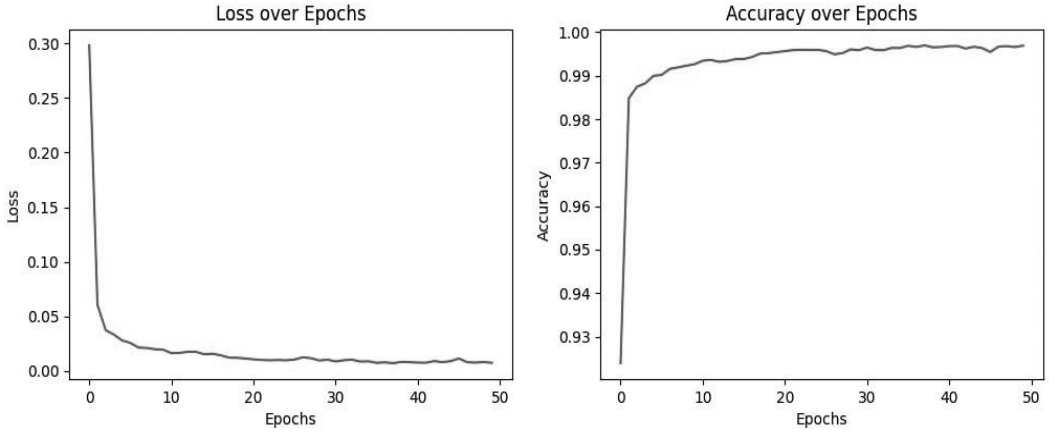


Рис.2.3 Залежність від кількості епох втрат (а) та якості передбачення зображення (б)

Кількісне порівняння передбачень, отриманих за допомогою навчених моделей нейронних мереж та людини, здійснювалося за допомогою коефіцієнта Dice [9].

Для підвищення точності передбачення масок (а отже підвищення коефіцієнта Dice) було запропоновано навчати нейронну мережу 10 разів, а потім визначені області усереднити за всіма визначеннями.

2.2 Другий етап

Для покращення якості сегментації, на другому етапі було запропоновано використовувати ряд нейронних мереж ще раз, але з меншим розміром досліджуваної області рентгенівського знімку, де здійснюється сегментація.

Місце знаходження цієї області визначається за результатами першого етапу, в такий спосіб. Як цю зменшену область вибираємо прямокутник , що включає хребці один вище (Th7) і один нижче (Th12) шуканої ділянки хребта, що включає хребці Th8-Th11. Приклади рентгенівських знімків і відповідних їм, створених вручну масок, за якими здійснювалося навчання нейронних мереж, показано на Рис.2.4.

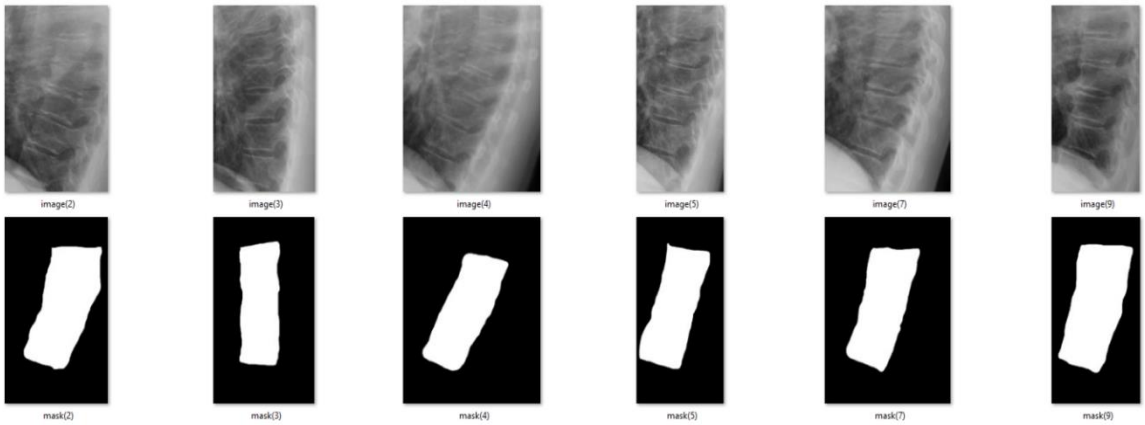


Рис.2.4 Приклади фрагментів вихідних зображень та відповідні маски шуканої області

Координати центру передбачуваної області зображення (X_c, Y_c) визначалися за допомогою передбаченої маски (отриманої на першому етапі)) за формулою:

$$X_c = \frac{1}{2}(Xmin_{max}) \tag{2.1}$$

та

$$Y_c = \frac{1}{2}(Ymin_{max}) \tag{2.2}$$

де $X_{\max}$, $X_{\min}$, $Y_{\max}$, та $Y_{\min}$ – координати крайніх точок у зображенні передбаченої області.

Далі навколо цього центру вибиралася вирізана з повного рентгенівського зображення прямокутна ділянка з розмірами $L_0 \times H_0$, які визначаються за формулою:

$$L_0 = 0.33 \cdot L, \tag{2.3}$$

та

$$H_0 = 0.5 \cdot H \tag{2.4}$$

де L і H - повна ширина та висота вхідного рентгенівського зображення. Коефіцієнти 0.33 і 0.5 вибиралися з аналізу 183 рентгенівських зображень так, щоб у вирізану область вмістилася ділянка хребта з хребцями Th7-Th12.

3 Експерименти та результати

Після проведення тренувань, проводилося тестування на іншому наборі зображень, що складаються з 58 рентгенівських знімків бічної проекції ВГК, взятих з інтернету [10], які отримані в різних умовах і на різних рентгенівських апаратах. При цьому слід зазначити, що попереднього відбору зображення за якістю не проводилося, а досліджувалися всі зображення, які були в цих базах.

Результати передбачення масок ділянки хребта, що включає хребці Th8-Th11, для двох різних навчань представлені на Рис.3.1. Для демонстрації ефективності багаторазового навчання вибрано знімки з високим рівнем шуму і розмитими межами хребта.

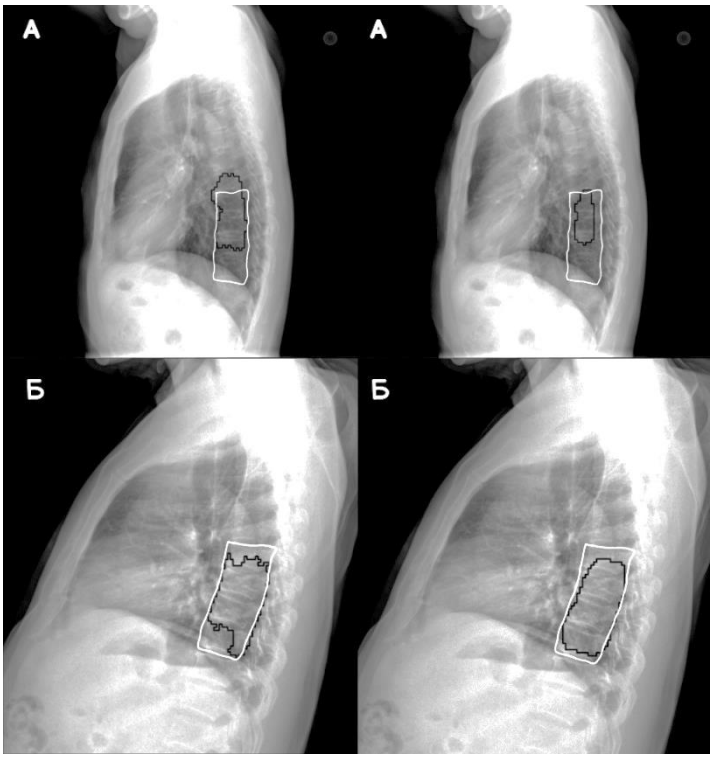


Рис.3.1 Приклади вихідних зображень, відповідних масок (білий контур) та передбачених областей (чорний контур) для двох різних навчань. А - зображення 14 з бази [10], Б - зображення 23 з бази [10]

Кількісні результати передбачень нейронних мереж, що окремо навчаються, а також усереднень по десяти навчаннях, наведені на Рис.3.2. Як видно з гістограм, наведених на Рис.3.2, окремі навчання можуть давати кращі результати за коефіцієнтом Dice для якогось конкретного рентгенівського зображення, але для іншого рентгенівського зображення це окреме навчання дає гірший результат порівняно з усередненням. На Рис.3.2, також наводяться результати усереднення передбачень двох нейронних мереж. Результати передбачення інших нейронних мереж (FCN8, FCN32, U-net, VGG_Segnet) Рис.3.2 не наводяться, оскільки вони значно поступалися нейронним мережам MobileNet_Segnet і MobileNet_Unet.

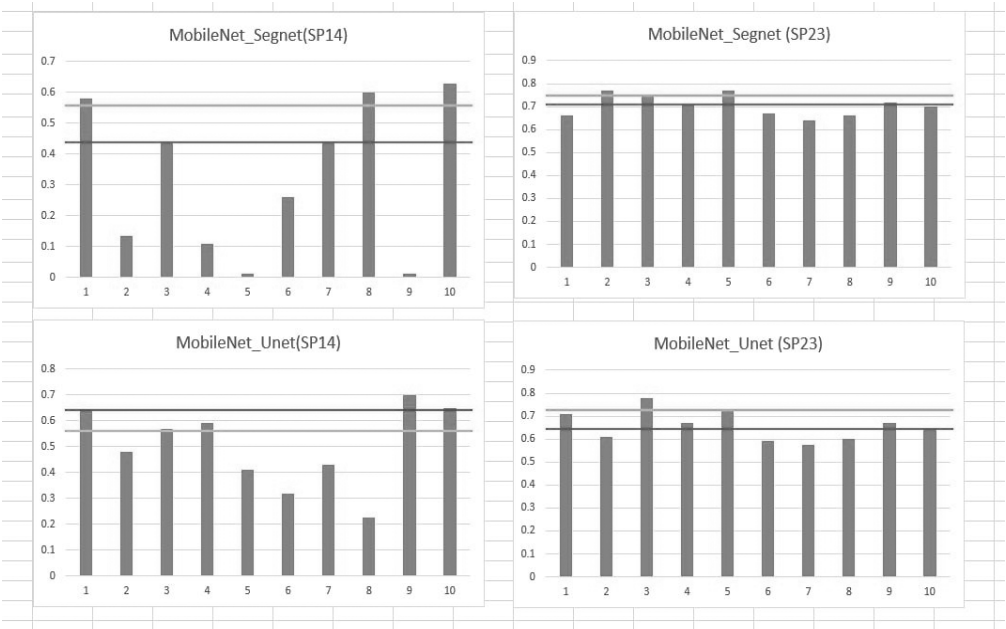


Рис.3.2 Кількісні результати десяти передбачень нейронних мереж, що окремо навчаються, а також усереднені по десяти навчаннях для двох зображень. Темна лінія – результат усереднення для певної мережі. Світла лінія – результат усереднення по обом мережам.

Однак, навіть результати цих найкращих нейронних мереж демонструють відносно низьку якість сегментації. Це зумовлено, як високим рівнем шуму, малим радіаційним контрастом, розмитими межами хребта, так й малим розміром шуканого зображення, проти повного зображення бічної проекції ВГК людини. Тому, як зазначалося вище, для поліпшення якості сегментації, на другому етапі, пропонувалося використовувати ряд нейронних мереж ще раз, але з меншим розміром досліджуваної області, де здійснюється сегментація.

Аналогічно першому етапу після проведення тренувань проводилося тестування на тому ж наборі зображень, що складаються з тих же 58 рентгенівських знімків бічної проекції ВГК, але зі зменшеною областю досліджень. Як і в першому етапі, для передбачення маски хребта, що включає хребці Th8-Th11, застосовувалися ті ж шість нейронних мереж з десятикратним навчанням і усередненням передбачених масок. Найкращі результати (за коефіцієнтом Dice) показали усереднення двох нейронних мереж MobileNet_Segnet та VGG_Segnet. Передбачені області для тих самих випадків, як і на Рис.5 показано на Рис.3.3. Коефіцієнт Dice цих двох випадків зріс з 0,543 до 0,820 для знімка № 14 і з 0,725 до 0,923 для знімка № 23.

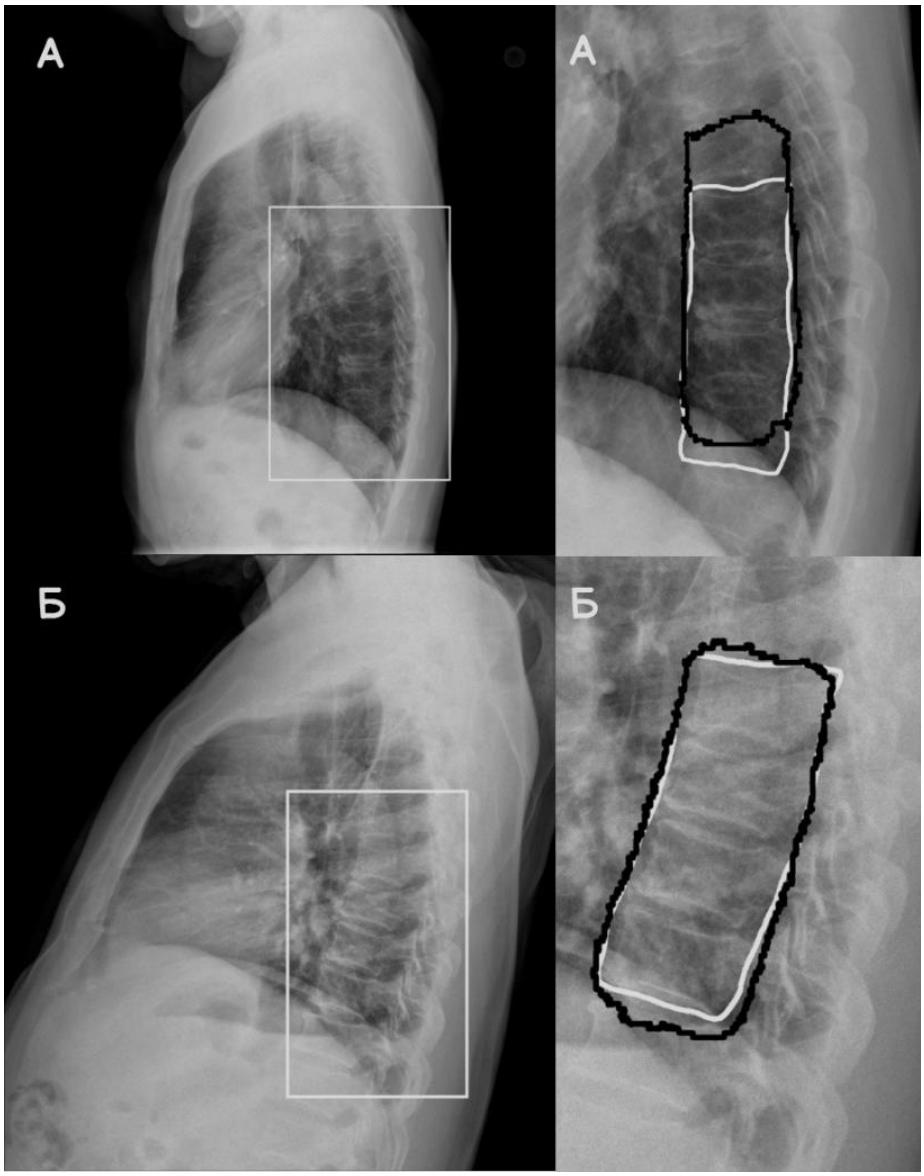


Рис.3.3 Приклади вихідних зображень, відповідних масок (білий контур) та передбачених областей (чорний контур). А - зображення 14 з бази [10], Б - зображення 23 з бази [10]

4 Висновки

Запропоновано метод підвищення точності передбачення нейронних мереж за рахунок двоетапного і багаторазового навчання низки нейронних мереж. Цей метод дозволив домогтися стабільності і точності передбачень шуканих ділянок хребта, навіть для знімків з високим рівнем шуму, низьким контрастом і розмитими межами шуканих об'єктів. Запропоновану методику можливо застосовувати як для дослідження інших органів людини, так і в не медичних застосуваннях, де зустрічаються неякісні зображення.

REFERENCES

1. G. Guglielmi, D. Diacinti, C. van Kuijk, F. Aparisi, C. Krestan, J. E. Adams, and T. M. Link, "Vertebral morphometric: current methods and recent advances," *European Radiology*, p. 14, 2008. https://link.springer.com/article/10.1007/s00330-008-0899-8?error=cookies_not_supported&code=d445f129-d830-4a13-9590-5989bcf5141c
2. S. Li and J. Yao, *Spinal Imaging and Image Analysis*, vol. 18, p. 507, 2015. https://link.springer.com/book/10.1007/978-3-319-12508-4
3. S. Ebrahimi, L. Gajny, W. Skalli, and E. Angelini, "Vertebral corners detection on sagittal X-rays based on shape modelling, random forest classifiers and dedicated visual features," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 7, no. 2, pp. 132-144, 2018. https://hal.science/hal-02181802/file/IBHGC_CMBBEIV_2018_Ebrahimi.pdf
4. L. R. Long and G. R. Thoma, "Segmentation and feature extraction of cervical spine x-ray images," in *Proc. SPIE 3661, Medical Imaging 1999: Image Processing*, May 1999. https://ui.adsabs.harvard.edu/abs/1999SPIE.3661.1037L/abstract
5. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," 2015. https://arxiv.org/abs/1505.04597
6. S. M. M. R. Al Arif, "Fully automatic image analysis framework for cervical vertebra in X-ray images," Doctoral thesis, City, University of London, 2018. https://pubmed.ncbi.nlm.nih.gov/29477438/
7. K. C. Kim, H. C. Cho, T. J. Jang, J. M. Choi, and J. K. Seo, "Automatic detection and segmentation of lumbar vertebrae from X-ray images for compression fracture evaluation," *arXiv:1904.07624v1 [physics.med-ph]*, Apr. 2019. https://arxiv.org/abs/1904.07624
8. Y. Chen, Y. Mo, A. Readie, G. Ligozio, T. Coroller, and B. W. Papie, "VertXNet: Automatic Segmentation and Identification of Lumbar and Cervical Vertebrae from Spinal X-ray Images," *arXiv:2207.05476v1 [eess.IV]*, Jul. 2022. https://arxiv.org/abs/2207.05476
9. L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, p. 297-302, 1945. https://esajournals.onlinelibrary.wiley.com/doi/10.2307/1932409
10. "Vindr.ai Datasets: SpineXR." [Online]. Available: https://vindr.ai/datasets/spinexr. [Accessed: 16-Jun-2024].

Koniukhov Vladyslav Dmytrovych *PhD student*
Anatolii Pidhornyi Institute of Mechanical Engineering Problems of NAS of Ukraine, 2/10, Pozharskyi str., Kharkiv, 61046, Ukraine

Morgun Oleg Mykolayovych *Ph.D., director*
"Laboratory of X-ray Medical Equipment" LTD
address: Dostoevsky str. 1, Kharkiv, 61102, Ukraine

Nemchenko Kostyantyn Eduardovych *D. of Sc., head of the department*
V. N. Karazin Kharkiv National University
Maydan Svobody, 4, Kharkiv, 61022, Ukraine

Multiple training of neural networks for automatic spine segmentation

Actuality
Preventive and diagnostic studies of the presence of bone diseases require morphometric studies of X-ray images of the chest area. Nowadays, artificial intelligence methods are increasingly being used to solve such problems. The main difficulties of this task are related to the fact that X-ray images have quality limitations, for example, in terms of signal-to-noise ratio or contrast. For this reason, the application of standard methods of image recognition or automatic diagnosis becomes impossible. These difficulties have led to the fact that there is currently a fairly large number of works in this field, but the results of most of them are insufficient for practical use.

Goal
Investigate the possibility of using artificial intelligence in the segmentation of medical images for the purpose of automatic diagnosis of diseases of the human bone system.

Research methods
The ensemble method of X-ray image segmentation has been used in the study. The baseline of training data was created on the basis of X-ray images taken from open sources. The total number of images is 183. The initial data was modified according to the requirements necessary for model training. All images were converted to grayscale and resized to 256x256 pixels.

Results
Using this method in the two test cases resulted in an improvement in accuracy from 0.543 to 0.820 for the first snapshot and from 0.725 to 0.923 for the second snapshot.

Conclusions
We have proposed and investigated the application of the methodology of using an ensemble of reusable neural networks for automatic segmentation of a certain area of the spine, namely the Th8-Th11 spine region. The application of this method allowed obtaining more stable and accurate predictions for the desired spine regions, even for images with high noise levels.

Keywords: *artificial intelligence, machine learning, image recognition, neural network, ensemble of neural networks, morphometry.*

Надійшла у першій редакції 23.04.2024, в останній - 26.05.2024.
The first version has been received on 23.04.2024, the final version - on 26.05.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-05

УДК (UDC) 004.94

Стрілець Вікторія Євгенівна *к.т.н., доцент кафедри теоретичної та прикладної системотехніки, Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, м. Харків, 61022*
e-mail: viktoria.strilets@karazin.ua;
https://orcid.org/0000-0002-2475-1496

Голіков Максим Сергійович *студент факультету комп'ютерних наук, Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, м. Харків, 61022*
e-mail: xa12284048@student.karazin.ua;
https://orcid.org/0000-0003-4842-7823

Модель комп'ютерної системи поселення студентів у гуртожиток

Мета роботи: автоматизація процесу поселення студентів у гуртожитки університетів через створення комп'ютерної системи поселення, яка враховує особисті вимоги студентів та покращує розподіл місць.
Методи дослідження: методи оптимізації, зокрема еволюційний підхід з застосуванням генетичних алгоритмів, моделювання архітектури програмного забезпечення, методи та технології розробки програмного забезпечення. Комп'ютерна система побудована на основі трирівневої архітектури, яка включає рівні презентації, бізнес-логіки та даних. Для розробки використані технології: Spring Boot для серверної частини, Angular для клієнтської частини та PostgreSQL для зберігання даних.
У **результаті** розроблена комп'ютерна система дозволяє автоматизувати процес поселення студентів у гуртожитки, враховуючи їхні індивідуальні потреби та переваги. Вона забезпечує кращий розподіл студентів по кімнатах гуртожитків, з урахуванням таких критеріїв, як стать, тип особистості (інтроверт/екстраверт), курс навчання. Система забезпечує можливість адміністрування та моніторингу процесу поселення, що дозволяє працівникам університету оперативно реагувати на зміни та вирішувати можливі проблеми. Вибір технологій, таких як Spring Boot, Angular та PostgreSQL, забезпечує кросплатформність, безпеку даних та можливість адаптації системи до змінних умов. Модель бази даних включає таблиці, у яких зберігається інформація про студентів, працівників, кімнати гуртожитків та інші необхідні дані, що дозволяє забезпечити ефективне управління інформацією.
Висновки: запропонована система дозволяє автоматизувати процес розподілення студентів у кімнати гуртожитку, враховуючи їхні індивідуальні потреби та побажання. Це робить її перспективним інструментом для управління житловим фондом закладів вищої освіти, що може прискорити процес поселення та спростити його бюрократичну сторону. Автоматизація процесу поселення дозволяє зменшити навантаження на адміністративний персонал та мінімізувати кількість помилок, що виникають під час ручного розподілу місць у гуртожитках. Таким чином, розроблена система не тільки покращує умови оформлення документів на проживання для студентів, але й сприяє підвищенню ефективності роботи адміністративного персоналу університетів.

***Ключові слова**: комп'ютерна система, еволюційний підхід, генетичний алгоритм, оптимізаційна задача, поселення студентів.*

Як цитувати: Стрілець В.Є., Голіков М.С. Модель комп'ютерної системи поселення студентів у гуртожиток. *Вісник Харківського національного університету імені В.Н.Каразіна, серія. «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2024. вип. 62. С.45-54. https://doi.org/10.26565/2304-6201-2024-62-05

How to quote: V.Y. Strilets, and M.S. Holikov, "Model of the computer system for settling students in a dormitory", *Bulletin of V.N. Karazin Kharkiv National University, series "Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp.45-54, 2024. https://doi.org/10.26565/2304-6201-2024-62-05

1 Вступ

Стратегія розвитку сучасного суспільства спрямована на впровадження інформаційно-комунікаційних технологій у всі сфери життя. Цифровізація є державною політикою України [1], яка має на меті побудову і розвиток інформаційного суспільства, орієнтованого на інтереси людей та відкрите до доступу, обміну і користування даними. Цифровізація сприяє полегшенню доступу до послуг у сфері охорони здоров'я й освіти, наданню фінансових послуг, прозорості та ефективності процесів управління.

Створення цифрового простору в закладах вищої освіти дозволяє не тільки забезпечити ефективний навчальний процес, але й поліпшити взаємодію між студентами і відділами та службами університетів, які відповідають за життєзабезпечення. Однією з таких служб є служба поселення студентів, яка відповідає за розміщення іногородніх студентів, аспірантів, докторантів, та сімей студентів, аспірантів, докторантів у гуртожитках університетів. Щорічно комісії з поселення розглядають заяви студентів (аспірантів, докторантів) на надання місця проживання у гуртожитках. Підходи до розподілення студентів по кімнатах зазвичай враховують лише стать студентів і напрямок підготовки (академічні групи). Тоді як особистісні риси студентів і побажання залишаються без уваги. Тому задля вирішення цієї проблеми є можливим створення інформаційних систем для поселення студентів, які будуть приймати заявки від студентів і дозволять автоматизувати процес розселення з урахуванням різноманітних критеріїв, починаючи від кількості місць житлового фонду і закінчуючи побажаннями студентів щодо психотипу їхніх майбутніх сусідів.

Метою роботи є автоматизація процесу поселення студентів до гуртожитків через створення, реалізацію і впровадження моделі комп'ютерної системи поселення студентів.

Створювана комп'ютерна система передбачає наявність модуля, який буде відповідати за вирішення задачі розселення студентів. Дана задача є комбінаторною оптимізаційною задачею про призначення. У контексті поселення студентів необхідно до уваги різні вимоги, наприклад, проживання студентів однієї статі в кімнатах, процес взаємного впливу чи спілкування студентів з урахуванням їх типу особистості, кількість вільних місць у кімнатах тощо. Саме ці вимоги (критерії) формують множину обмежень задачі оптимізації. Для її розв'язання пропонується використати еволюційний підхід, зокрема генетичні алгоритми.

Еволюційні алгоритми, натхненні природним відбором і генетичними процесами, є дієвими в розв'язанні комбінаторних оптимізаційних задач [2, 3]. Однією із ключових причин вибору еволюційних алгоритмів, особливо генетичного алгоритму, є здатність адаптуватися до змінних умов. Наприклад, у задачі розселення студентів можуть виникати нові обмеження або вимоги, на які традиційні методи реагують повільно або недостатньо гнучко. Еволюційні алгоритми, завдяки своїй природі, здатні оперативно корегувати свої рішення, забезпечуючи при цьому ефективність рішень.

Отже, впровадження еволюційного підходу в модель комп'ютерної системи поселення студентів обіцяє значні переваги: можливість врахування множини факторів різних типів, покращене використання ресурсів та гнучкість управління. Це робить створювану модель перспективним інструментом для вирішення організаційних і управлінських задач у сфері соціального забезпечення закладів вищої освіти.

2 Опис існуючих рішень і формулювання проблеми для розв'язання

Процес поселення студентів у гуртожиток регулюється нормативно-правовими документами навчальних закладів і є достатньо стандартним з незначними відмінностями, тобто автоматизовані системи поселення студентів і їх моделі будуть базуватися на схожих принципах. Таким чином, одними з основних вимог до подібних систем є кросплатформність і безпека даних [4].

Для побудови інтегрованого інформаційного середовища в закладах вищої освіти наразі створені і використовуються такі автоматизовані системи, як АС Деканат, АС Приймальна комісія, АС Студмістечко [5]. Дані системи розроблені Науково-дослідним інститутом прикладних технологій (м. Київ) і успішно впроваджуються у роботу ВНЗ. Вони відповідають вимогам до кросплатфомності і безпеки даних, але їх вартість є достатньо високою, а використання потребує попередніх тренінгів для адміністративного персоналу, щоб набути необхідних навичок для роботи з системами.

Також наразі існує модель автоматизації поселення студентів для Національного університету «Львівська політехніка», розроблена за допомогою HiAsm, інтегрованого середовища розробки програм для Windows [6]. Ця модель забезпечує повний автоматизований процес поселення студентів у гуртожиток. Однак, оскільки вона розроблена тільки для однієї платформи, не можна вважати її повністю кросплатформною.

Крім того, існуючі автоматизовані системи реалізують типовий підхід до розміщення студентів, без урахування додаткових особистих вимог. Тому пропонується розробити модель комп'ютерної системи поселення студентів, яка відповідає вимогам кросплатформності та

безпеки, а також здатна отримати оптимальне розміщення студентів, приймаючи до уваги низку додаткових факторів (наприклад, психотип людини або релігійні переконання).

3 Еволюційний підхід для розв'язання задачі про розподілення студентів в гуртожитку

У системах автоматизації поселення студентів у гуртожиток потрібно врахувати різні критерії і вимоги. Більшість критеріїв можуть бути суперечливими, тому виникає проблема вибору коректного критерію. Це означає, що доводиться йти на компроміс, обираючи для кожного критерію значення, яке не є оптимальним, але прийнятним з урахуванням інших критеріїв.

У створеній моделі поселення студентів пропонується взяти до уваги такі критерії: стать, тип особистості (інтроверт чи екстраверт) та курс студентів.

1. Гендерний критерій:
   - для кожної кімнати перевіряється, чи всі студенти однієї статі (чоловіки або жінки);
   - якщо в кімнаті є і чоловіки, і жінки, збільшується лічильник помилки невідповідності статі *genderMismatch*, який показує кількість таких кімнат.
2. Баланс інтровертів та екстравертів:
   - обчислюється кількість інтровертів та екстравертів у кожній кімнаті;
   - якщо кількість інтровертів і екстравертів однакова, показник невідповідності балансу *sociotypeMismatch* буде збільшуватися;
   - перевіряється, чи в половині кімнат цей баланс рівний, і якщо так, то *sociotypeMismatch* збільшується ще раз.
3. Розподіл за роками навчання:
   - підраховується кількість студентів кожного року навчання в кімнаті та обчислюється середня різниця між роками;
   - якщо різниця між роками для будь-якого студента перевищує середню різницю більше, ніж на один курс, збільшується помилка середньої різниці року навчання *courseMismatch*.

Загальна цільова функція обчислюється за формулою:

$$f = 0.8genderMismatch + 0.1sociotypeMismatch + 0.1courseMismatch.$$

Для даної цільової функції потрібно досягти мінімуму, оскільки в такому випадку буде забезпечена мінімальна помилка для кожного з критеріїв. Чим менше значення цільової функції, тим ефективнішим є розподілення студентів.

Вагові коефіцієнти для кожного критерію визначають їх важливість в загальній оцінці цільової функції. Найбільший вплив має гендерний критерій, оскільки це важливо для комфорту проживання студентів.

Для розв'язання сформульованої задачі про розподілення студентів в гуртожитку був обраний еволюційний підхід, а саме – генетичний алгоритм.

Еволюційний підхід використовує принципи еволюційних процесів у природі для вирішення проблем пошуку найкращих рішень серед можливих варіантів. Цей підхід базується на теорії природного відбору Дарвіна та концепції генетичного розвитку. Еволюційні алгоритми – це потужний інструмент для вирішення будь-яких складних задач. Цей тип алгоритмів включає в себе генетичні алгоритми, еволюційну стратегію, еволюційне програмування, алгоритми диференціальної еволюції та генетичне програмування. Їх широко використовують у різних галузях, включаючи оптимізацію, штучний інтелект та інженерію [7].

Генетичні алгоритми використовують концепції "хромосом" і "генів" для моделювання можливих розв'язків задачі. Хромосоми представляють розв'язки, а гени визначають їх параметри чи характеристики. Наприклад, для задачі поселення студентів у кімнати, кожна хромосома може представляти собою список кімнат, а гени визначатимуть кожну кімнату у цьому списку, і кожен студент у кімнаті – це значення гену [8].

Важливо зазначити, що генетичні алгоритми працюють з популяцією індивідів, що відрізняється від інших алгоритмів, які працюють з одним рішенням.

Генетичний алгоритм імітує природний відбір, створюючи та еволюціонуючи популяцію рішень, де кращі рішення мають більше шансів на розмноження та мутацію. В розглядуваній задачі рішення – це розподіл студентів по кімнатах, враховуючи їхні характеристики, такі як стать, курс і соціотип.

Основна ідея полягає в тому, що випадкові мутації та схрещування створюють нові покоління рішень, які все краще вирішують поставлену задачу. В алгоритмі реалізовано адаптацію

параметрів (ймовірності мутації та кросовера) на основі середньої пристосованості популяції, що дозволяє алгоритму швидше знаходити наближені до оптимальних рішення, адаптуючись до змінних умов.

Алгоритм складається з п'яти основних етапів у заданому порядку:

- ініціалізація;
- відбір;
- схрещування;
- мутація;
- оцінка пристосованості.

На етапі ініціалізації потрібно визначити вхідні дані, щоб створити початкову популяцію. Це здійснюється шляхом запитів до бази даних для отримання списку студентів і кімнат з їх кількістю місць. Дані про студентів включають стать, курс та соціотип, а також інформацію про кількість кімнат і кількість місць у кожній кімнаті.

Після отримання даних, студенти випадковим чином розподіляються по кімнатах, створюючи початкову популяцію. Геном, у цьому контексті, представляє структуру даних, яка визначає розподіл студентів по кімнатах. Встановлюється розмір популяції.

Обчислення пристосованості (цільової функції) кожного індивіда відбувається після створення популяції. Вони сортуються за рівнем пристосованості, і перша половина популяції відбирається як потенційні батьки. Метод відбору, який використовується, називається "турнірний відбір". Його суть полягає в тому, що краще пристосовані особини мають більше шансів передати свої "гени" наступному поколінню, покращуючи таким чином якість рішень.

Після відбору відбувається схрещування. Використовується одноточковий кросовер, де випадковим чином вибирається точка розрізу в генетичних послідовностях (списку кімнат). Частина від одного батька передається першому нащадку до точки розрізу, а частина від іншого батька після точки розрізу – другому нащадку.

Потім відбувається мутація, контрольована ймовірнісним методом. Використовується "мутація обміну", де двоє студентів змінюються місцями між різними кімнатами, щоб зберегти корисні комбінації.

Адаптивність кросовера та мутації – особлива риса цього алгоритму. Вона дозволяє алгоритму адаптуватися до умов задачі та підтримувати баланс між дослідженням нових рішень (exploration) і використанням вже знайдених (exploitation). Якщо кросовер добре працює, його ймовірність збільшується, і навпаки, якщо мутація ефективніша, то збільшується її ймовірність.

Такий підхід до розв’язання задачі дозволяє алгоритму добре пристосовуватися і ефективно знаходити наближені до оптимального рішення в змінному середовищі.

4 Моделювання процесу поселення студентів

При створенні моделі комп'ютерної системи необхідно визначити загальну структуру та напрям функціонування системи, які мають відображати основні етапи процесу поселення студентів у гуртожиток

Були виділені такі основні етапи цього процесу:

1. Реєстрація студента у системі.
2. Авторизація всіх користувачів, задіяних у процесі.
3. Студент надсилає запити працівникам університету для підтвердження виконання певних вимог, залежно від спеціалізації працівника.
4. Співробітники приймають рішення щодо підтвердження виконання конкретних вимог студентом.
5. Якщо студент виконав усі вимоги, він отримує дозвіл на поселення в гуртожиток.
6. Розподілу студентів по гуртожитках (кімнатах) з використанням генетичного алгоритму.
7. Додаткова перевірка на гендер, якщо це необхідно.

Детальніше розглянемо етапи 3 і 4. Під працівниками розуміються користувачі з наступними ролями:

- медсестра;
- економіст;
- заступник декана.

Кожен із цих співробітників відповідає за конкретні завдання: медсестра – підтверджує проходження студентом медичного огляду, економіст – перевіряє оплату місця в гуртожитку, а заступник декана – підтверджує статус студента. Цей процес можна описати на концептуальному рівні за допомогою діаграми прецедентів, як показано на рис. 1.

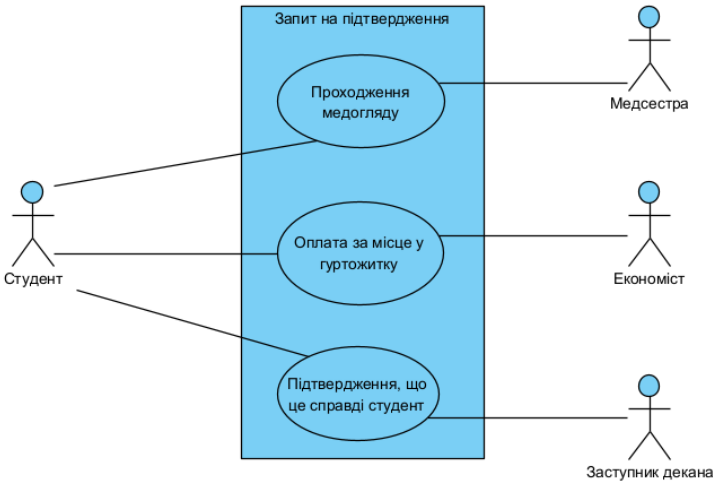


Рисунок 1. Діаграма прецедентів взаємодії студента й адміністрації

Таким чином студент через систему подає запит на поселення і необхідні для цього документи і отримує відповідь після розгляду документів адміністрацією і застосування моделі поселення.

5 Проектування архітектури комп'ютерної системи поселення студентів

Для забезпечення стабільного функціонування і безпеки даних для комп'ютерної системи поселення студентів була обрана розподілена архітектура.

Розподілена архітектура полягає в тому, що компоненти системи розташовані на різних фізичних або логічних машинах і з'єднані мережею для підвищення масштабованості та ефективності. Трирівнева архітектура є підтипом розподіленої архітектури, де програмна система розділена на три рівні або компоненти [9]. Кожен з них має своє призначення та спрямований на вирішення певних завдань. Ці рівні включають презентаційний, бізнес-логіки та рівень даних (рис. 2).

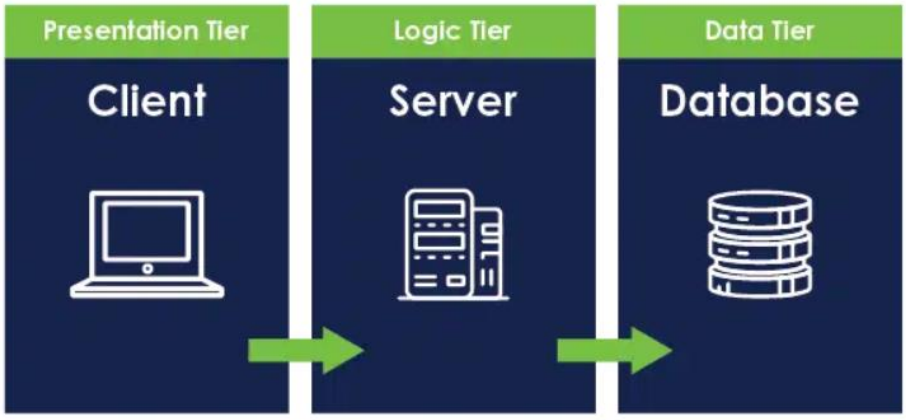


Рисунок 2. Модель трирівневої архітектури комп'ютерної системи

На рівні представлення знаходиться інтерфейс, який користувач бачить і з яким він взаємодіє. Цей рівень відповідає за виведення інформації для користувача та обробку його введень. Він передає команди для обробки наступним рівнем, а також отримує відповіді для відображення користувачеві.

Бізнес-логіка – це місце, де знаходиться "мозок" програмної системи. Тут обробляється бізнес-логіка та процеси, що визначають, як система має працювати. Цей рівень встановлює

контрольований та безпечний спосіб взаємодії між рівнем представлення та рівнем даних. Коли користувач взаємодіє з системою, наприклад, відправляє заповнену форму, бізнес-логіка аналізує цю інформацію та виконує потрібні операції для взаємодії з даними.

Рівень доступу до даних – це рівень, який передбачає обмін інформацією з базою даних чи іншими джерелами даних. Цей рівень відповідає за управління та доступ до даних для їх наступної обробки на шарі бізнес-логіки.

На рівні бізнес-логіки реалізований модуль для розв'язання задачі поселення студентів на основі запропонованого еволюційного підходу. Цей блок взаємодіє з базою даних та отримує вхідні дані про студентів: курс, соціотип та гендер. Після роботи алгоритму результати виводяться у консоль у спрощеному форматі, який показує кінцевий варіант поселення студентів по кімнатах. Цей підхід може бути розширений у майбутньому для створення повноцінної системи, яка буде враховувати більше параметрів та надавати користувачам більше інформації про їхнє майбутнє проживання.

6 Визначення структури бази даних комп'ютерної системи

Для формування структури бази даних перш за все, потрібно визначити головні сутності, з якими відбуватимуться основні операції. Такими сутностями є:

- student (таблиця з інформацією про студентів);
- users (таблиця для збереження даних користувачів);
- role (таблиця для визначення ролі користувача);
- employee (таблиця з інформацією про працівників університету);
- administrator (таблиця з даними про адміністраторів);
- dormitory (таблиця з інформацією про гуртожитки).

Додаткові таблиці використовуються для допоміжних цілей, щоб краще впорядкувати та структурувати дані. Наприклад, таблиці group, speciality і department доповнюють інформацію про приналежність студента до академічної групи і факультету, а таблиця room містить дані про кімнати гуртожитків. Загалом знадобилося 13 таблиць.

На основі концептуальної моделі бази даних побудована діаграма розширеної сутнісно-зв'язкової моделі (Enhanced Entity-Relationship, EER). Це розширення типової моделі сутнісно-зв'язкових діаграм додає можливості та додаткові концепції для повного і точнішого опису між об'єктами в базі даних саме взаємозв'язків [10]. Фрагмент цих зв'язків представлений у канонічній нотації EER-моделі на рис. 4.

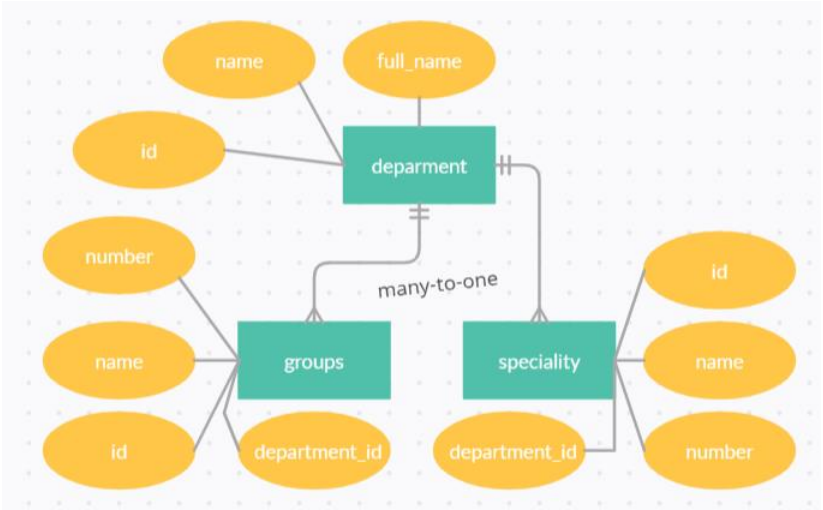


Рисунок 3. Фрагмент канонічної нотації EER-моделі для таблиць department, groups та speciality

7 Вибір компонентів для розробки

Щоб створене програмне забезпечення працювало на різних пристроях і платформах, необхідно використовувати технології, що забезпечують кросплатформність на рівні середовищ виконання. Одним із прикладів таких технологій є програмування на мові Java [11]. Ідеальним

рішенням була б модель, яка працює безпосередньо в браузері, без необхідності завантаження на пристрій [4].

Програмне рішення у форматі веб-додатку можна реалізувати за допомогою Spring Boot, Angular та PostgreSQL. Це дозволить використовувати комп'ютерну систему на різних пристроях, задовольняючи головний критерій — кросплатформність. У додатку також можна забезпечити безпечне зберігання даних студентів у хмарному сховищі, що є надійнішим, ніж тримання їх на окремому комп'ютері [4].

Під час розробки моделі комп'ютерної системи для вирішення задачі поселення студентів у гуртожиток важливо правильно обрати компоненти та інструменти, щоб забезпечити ефективність, надійність та безпеку додатку. Розглянемо переваги обраних компонентів:

1. Фреймворк Spring Boot ідеально підходить для розробки додатків завдяки своїй простоті та швидкому налаштуванню. Він має вбудований веб-сервер Tomcat, що полегшує розгортання додатків, а також пропонує багато корисних бібліотек і функцій, які заощаджують час і зусилля розробників [12, 13]. Його концепція "конвенції над конфігурацією" вказує, що багато параметрів вже вбудовані за замовчуванням, що дозволяє зосередитися на розробці коду, який вирішує конкретні завдання. Вбудований контейнер для обробки HTTP-запитів дозволяє запускати додатки як окремі виконувані файли, що полегшує процес розгортання
2. PostgreSQL – це найкращий вибір для бази даних з кількох причин:
   - надійність: PostgreSQL славиться своєю стабільністю і надійністю, що забезпечує безпеку даних студентів та гуртожитків;
   - розширюваність та гнучкість: можливість додавання власних типів даних, функцій та розширень дозволяє створювати унікальні рішення;
   - підтримує обробку географічної інформації та роботи з геоданими завдяки розширенням, таким як PostGIS;
   - відкритий код PostgreSQL сприяє створенню спільноти розробників і забезпечує повний контроль над базою даних;
   - сумісність з Java та Hibernate: PostgreSQL добре працює з Java та ORM-рішеннями, такими як Hibernate, що полегшує взаємодію з базою даних із веб-додатку [14, 15].
3. ORM (Java Persistence API з Hibernate) дозволяє представляти об'єкти Java у вигляді даних у базі даних. Hibernate спрощує взаємодію з базами даних, дозволяючи виконувати операції без написання складних SQL-запитів [13].
4. Spring MVC – цей компонент Spring Framework використовує архітектурний шаблон Model-View-Controller, що полегшує розробку та управління веб-запитами у додатках на Java [12, 13].
5. Spring Security забезпечує безпеку та автентифікацію користувачів, захищаючи особисту та конфіденційну інформацію [12, 13].
6. Maven – популярний інструмент для складання проектів та управління залежностями у середовищі Java, що спрощує додавання бібліотек, забезпечує уніфікований процес зборки, дозволяє швидко впроваджувати нові функції та оновлення [16].
7. Angular – потужний фреймворк для розробки користувацьких інтерфейсів, що дозволяє створювати динамічні та інтерактивні веб-сайти [17]. Однією з головних переваг Angular є двостороння прив'язка даних, що автоматично відображає зміни в моделі в інтерфейсі користувача і навпаки. Angular також має потужний механізм для обробки HTTP-запитів та інтегровану систему модулів, що спрощує організацію коду і додавання нових функцій.

Таким чином, вибір цих компонентів обумовлений їх продуктивністю, ефективністю, масштабованістю та простотою використання.

8 Застосування моделі комп'ютерної системи поселення студентів

Створена програмна реалізація моделі комп'ютерної системи поселення студентів була протестована для розселення 112 студентів по 28 кімнатам гуртожитку. Кожна кімната мала 4 вільних місця.

При ініціалізації ці початкові дані були надані генетичному алгоритму. Розмір популяції заданий – 100 індивідів.

Для адаптації алгоритму до умов задачі були задані параметри операторів кросовера та мутації (табл. 1).

Таблиця 1. Використані параметри

Значення	Мутація	Кросовер
Мінімальне	0.01	0.6
Максимальне	0.5	0.9
Початкове	0.1	0.8

Алгоритм виконувався протягом 10000 ітерацій.

У результаті роботи алгоритму був отриманий розподіл студентів по кімнатах (рис. 4, 5).

```
Кімната 1:
Гендер: чоловік, Курс: 6, Соціотип: інтроверт
Гендер: чоловік, Курс: 5, Соціотип: інтроверт
Гендер: чоловік, Курс: 6, Соціотип: інтроверт
Гендер: чоловік, Курс: 5, Соціотип: екстраверт

Кімната 2:
Гендер: чоловік, Курс: 3, Соціотип: інтроверт
Гендер: чоловік, Курс: 3, Соціотип: інтроверт
Гендер: чоловік, Курс: 5, Соціотип: інтроверт
Гендер: чоловік, Курс: 3, Соціотип: екстраверт

Кімната 3:
Гендер: чоловік, Курс: 5, Соціотип: інтроверт
Гендер: чоловік, Курс: 3, Соціотип: інтроверт
Гендер: чоловік, Курс: 6, Соціотип: інтроверт
Гендер: чоловік, Курс: 1, Соціотип: інтроверт

Кімната 4:
Гендер: жінка, Курс: 3, Соціотип: інтроверт
Гендер: жінка, Курс: 4, Соціотип: екстраверт
Гендер: жінка, Курс: 6, Соціотип: інтроверт
Гендер: жінка, Курс: 1, Соціотип: інтроверт

Кімната 5:
Гендер: чоловік, Курс: 6, Соціотип: інтроверт
Гендер: чоловік, Курс: 3, Соціотип: інтроверт
Гендер: чоловік, Курс: 6, Соціотип: інтроверт
Гендер: чоловік, Курс: 1, Соціотип: інтроверт
```

Рисунок 4. Фрагмент результатів поселення

```
Кімната 16:
Гендер: чоловік, Курс: 2, Соціотип: інтроверт
Гендер: жінка, Курс: 1, Соціотип: екстраверт
Гендер: чоловік, Курс: 5, Соціотип: екстраверт
Гендер: жінка, Курс: 5, Соціотип: екстраверт

Кімната 17:
Гендер: жінка, Курс: 2, Соціотип: інтроверт
Гендер: чоловік, Курс: 4, Соціотип: екстраверт
Гендер: жінка, Курс: 1, Соціотип: інтроверт
Гендер: жінка, Курс: 1, Соціотип: екстраверт

Кімната 18:
Гендер: жінка, Курс: 4, Соціотип: інтроверт
Гендер: жінка, Курс: 5, Соціотип: інтроверт
Гендер: чоловік, Курс: 4, Соціотип: екстраверт
Гендер: чоловік, Курс: 1, Соціотип: інтроверт
```

Рисунок 5. Студенти з різними гендерами у кімнаті

Аналіз результатів розселення показує, що алгоритм працює і задача вирішується, але іноді може виникати ситуація, коли в одній кімнаті опиняються студенти різної статі (рис. 5). Для вирішення цієї проблеми буде розроблений додатковий алгоритм, який шукає студентів, що підходять для переміщення, і змінює їх місцями, забезпечуючи правильний розподіл.

9 Висновки

Процес цифровізації вимагає створення різноманітних інформаційних систем, до яких можна віднести запропоновану модель комп'ютерної системи поселення студентів.

Впровадження в комп'ютерну систему генетичного алгоритму, дозволяє удосконалити процес поселення студентів у гуртожитки, забезпечуючи розв'язання задачі розподілення студентів як задачі комбінаторної оптимізації. Використання еволюційного підходу забезпечує ефективний розподіл студентів за рахунок адаптивності до змінних умов і здатності швидко коригувати рішення у відповідь на нові обмеження чи вимоги. Це забезпечує виконання обмежень і вимог задачі, ефективне використання ресурсів та більшу гнучкість управління процесом поселення.

Для успішної реалізації моделі комп'ютерної системи був проведений детальний аналіз доступних підходів і технологій розробки. Були обрані для програмної реалізаці технології, такі як Java, Spring Boot, Angular і PostgreSQL, що забезпечує кросплатформність, надійність і безпеку розроблюваного програмного забезпечення.

Розроблена архітектура комп'ютерної моделі з трирівневою структурою, гарантує ефективний розподіл обов'язків та кращу взаємодію між компонентами моделі. Впровадження еволюційних

алгоритмів на рівні бізнес-логіки дозволяє автоматизувати процес поселення з урахуванням гендерних, соціотипних та інших індивідуальних характеристик студентів.

Запропонований підхід має потенціал значно спростити та оптимізувати процес поселення студентів у гуртожитки, підвищити задоволеність користувачів і ефективність використання житлових ресурсів навчальних закладів.

СПИСОК ЛІТЕРАТУРИ

1. Концепція розвитку цифрової економіки та суспільства України на 2018 – 2020 роки / Розпорядження Кабінету Міністрів України від 17 січня 2018 р.
2. Wilson J. A genetic algorithm for the generalised assignment problem. J Oper Res Soc 48, 1997. pp. 804–809. DOI: https://doi.org/10.1057/palgrave.jors.2600431
3. Васильєв О. Б., Васильєва Н. С., Кічмаренко О. Д. Методи розв'язування задач багатокритеріальної оптимізації : метод. вказ. Одеса: Одеський національний університет імені І. І. Мечникова, 2017. 48 с.
4. Голіков М.С. Аналіз комп'ютерних моделей автоматизації поселення студентів у гуртожиток. *Комп'ютерне моделювання у наукоємних технологіях : зб. наукових праць VIII Міжнар. наук.-техн. конф.* Харків : ХНУ імені В.Н. Каразіна, 2022. С. 32- 34.
5. АСУ «ВНЗ». Головна сторінка : веб-сайт. URL: https://vuz.osvita.net/.
6. Струк Є., Дубленич Р., Фабрі Л., Автоматизація процесу поселення студентів у гуртожиток студмістечка. *Вісник національного університету «Львівська політехніка» Комп'ютерні науки та інформаційні технології*. 2016. Серія №843. С. 35-42.
7. Корнієнко В.І., Гусєв О.Ю., Герасіна О.В. Інтелектуальне моделювання нелінійних динамічних процесів у системах керування, кібербезпеки, телекомунцікайцій : підручник. Нац. техн. ун-т «Дніпровська політехніка». Дніпро : НТУ «ДП», 2020. 536 с.
8. Кононюк А.Ю. Нейроні мережі і генетичні алгоритми. Київ : Корнійчук, 2008. 446 с.
9. Three-Tier Architecture Approach for Custom Applications. IT Goals Achieved with Zirous, Technology Solutions : веб-сайт. URL: https://www.zirous.com/2022/11/15/three-tier-architecture-approach-for-custom-applications-2/ (дата звернення: 12.04.2024).
10. Visual-paradigm. Diagrams : веб-сайт. URL : https://www.visual-paradigm.com/ (дата звернення: 15.04.2024).
11. Bruce Eckel. Thinking in Java. 4th ed. p. cm. Includes bibliographical references and index : книга. United States of America : Courier in Stoughton, 2006. 1057 p.
12. Spring. Spring makes Java simple : веб-сайт. URL : https://spring.io/ (дата звернення: 01.04.2024).
13. Baeldung. Learn Java : веб-сайт. URL : https://www.baeldung.com/ (дата звернення: 01.04.2024).
14. Postgresql. База даних : веб-сайт. URL : https://www.postgresql.org/ (дата звернення: 01.04.2024).
15. Пасічник В.В., Резніченко В.А. Організація баз даних та знань : підручник МОНУ. Київ: BHV, 2006. 384 с.
16. Apache Maven. Software project management and comprehension tool : веб-сайт. URL : https://maven.apache.org/ (дата звернення: 01.04.2024).
17. Angular. Deliver web apps with confidence : веб-сайт. URL : https://angular.io/ (дата звернення: 01.04.2024).

REFERENCES

1. Concept of the development of the digital economy and society of Ukraine for 2018-2020 / Decree of the Cabinet of Ministers of Ukraine dated January 17, 2018. [in Ukrainian]
2. J. Wilson. *A genetic algorithm for the generalised assignment problem*. J Oper Res Soc 48, 1997. pp. 804–809. https://doi.org/10.1057/palgrave.jors.2600431
3. O.B. Vasiliev, N.S. Vasilieva, O.D. Kichmarenko. *Methods for solving multicriteria optimisation problems: methodological guidance*. Odesa: Odesa National University named after I. I. Mechnikov, 2017. 48 p. [in Ukrainian]
4. M.S. Holikov. Analysis of computer models for automation of student accommodation in a hostel. *Computer modelling in science-intensive technologies: proceedings of the VIII International Scientific and Technical Conference: V. N. Karazin Kharkiv National University*, 2022. P. 32-34. [in Ukrainian]

5. ASU «VNZ». Main page: website. URL: https://vuz.osvita.net/.
6. E. Struk, R. Dublenych, L. Fabri. Automation of the process of settling students in the campus dormitory. *Bulletin of the National University 'Lviv Polytechnic' Computer Science and Information Technology*, Series No. 843, pp. 35-42, 2016. [in Ukrainian]
7. V.I. Korniienko, O.Y. Gusev, O.V. Gerasina. *Intellectual modelling of nonlinear dynamic processes in control systems, cybersecurity, telecommunications: textbook*. National Technical University 'Dnipro Polytechnic'. Dnipro: NTU 'DP', 2020. 536 c. [in Ukrainian]
8. A.Y. Kononiuk. *Neural networks and genetic algorithms*. Kyiv: Korniychuk, 2008. 446 c. [in Ukrainian]
9. Three-Tier Architecture Approach for Custom Applications. IT Goals Achieved with Zirous, Technology Solutions: website. URL: https://www.zirous.com/2022/11/15/three-tier-architecture-approach-for-custom-applications-2/ (accessed 12.04.2024).
10. Visual-paradigm. Diagrams: website. URL: https://www.visual-paradigm.com/ (accessed 15.04.2024).
11. Bruce Eckel. Thinking in Java. 4th ed. p. cm. Includes bibliographical references and index : book. United States of America: Courier in Stoughton, 2006. 1057 c.
12. Spring. Spring makes Java simple: website. URL: https://spring.io/ (accessed 01.04.2024).
13. Baeldung. Learn Java : website. URL : https://www.baeldung.com/ (accessed 01.04.2024).
14. Postgresql. Database : website. URL: https://www.postgresql.org/ (accessed 01.04.2024).
15. V.V. Pasichnyk, V.A. Reznichenko. *Organisation of databases and knowledge: textbook of the Ministry of Education and Science*. Kyiv: BHV, 2006. 384 c. [in Ukrainian]
16. Apache Maven. Software project management and comprehension tool : website. URL: https://maven.apache.org/ (accessed 01.04.2024).
17. Angular. Deliver web apps with confidence : website. URL: https://angular.io/ (accessed 01.04.2024).

Strilets Viktoriia *Ph.D, associate professor of the theoretical and applied system engineering department, V.N. Karazin Kharkiv National University, 4 Svobody sq., Kharkiv, Ukraine, 61022*

Holikov Maksym *student of the computer science faculty V.N. Karazin Kharkiv National University, 4 Svobody sq., Kharkiv, Ukraine, 61022*

Model of the computer system for settling students in a dormitory

Purpose: to automate the process of settling students in university dormitories by creating a computer system that takes into account the personal requirements of students and improves the allocation of places.
Research methods: optimization methods, in particular evolutionary approach using genetic algorithms, software architecture modeling, software development methods and technologies. The computer system is built on a three-tier architecture that includes presentation, business logic, and data layers. Technologies used for development: Spring Boot for the server part, Angular for the client part and PostgreSQL for data storage.
As a **result**, the developed computer system allows automating the process of settling students in dormitories, considering their individual needs and preferences. It ensures better allocation of students to dormitory rooms, taking into account such criteria as gender, personality type (introvert/extrovert), study year and other important factors. The system also provides the ability to administer and monitor the settlement process, which allows university staff to quickly respond and solve possible problems. The choice of technologies, such as Spring Boot, Angular and PostgreSQL, ensures cross-platform compatibility, data security and the ability to adapt the system to changing conditions. The database model includes tables that store information about students, staff, dorm rooms and other necessary data, which allows for efficient information management.
Conclusions: the proposed system allows automating the process of assigning students to dormitory rooms, taking into account their individual needs and wishes. This makes it a promising tool for managing the residential fund of higher education institutions, which can speed up the settlement and simplify bureaucratic processes. Automation of the settlement process allows reducing the burden on administrative staff and minimize the number of errors that occur during the manual allocation of places in dormitories. Thus, the developed system not only improves the conditions for registration of residence documents for students, but also contributes to increasing the efficiency of the administrative staff of universities.

Keywords*: computer system, evolutionary approach, genetic algorithm, optimization problem, student accommodation.*

Надійшла у першій редакції 01.05.2024, в останній - 07.06.2024.
The first version has been received on 01.05.2024, the final version - on 07.06.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-06

УДК (UDC) 004.942

Сузікова Олена Геннадіївна
канд. псих.наук, ст.викл. кафедри прикладної математики
Харківський національний університет імені В. Н. Каразіна,
майдан Свободи, 6, м. Харків, 61022
e-mail: osuzikova@karazin.ua
https://orcid.org/0009-0002-1305-1256

Роль семантичного аналізу в подоланні обмежень реляційної моделі даних

Актуальність. Стаття присвячена визначенню ролі семантичного аналізу при подолання обмежень традиційної реляційної моделі даних. Потреба в семантичному аналізі даних виникає внаслідок запиту на більш виразні та ємні концептуальні моделі даних. Проблема, що досі повністю не вирішена, полягає в тому, що класичні реляційні модель не мають прямої підтримки семантики даних - зв'язків, абстракції даних, успадкування, поліморфізму, інкапсуляції, складних об'єктів, а також динамічних властивостей об'єктів. Під семантикою розуміється використання певних конструкцій та методів для того, щоб виражати особливості прикладного середовища, які залишаються поза традиційною реляційною моделлю. Семантичний аналіз дозволяє визначити більш складні відношення та взаємодії між сутностями, включаючи класифікації, агрегації (складні об'єкти, які містять інші об'єкти), асоціації (зв'язки між сутностями). тощо. Ціллю вживання компонентів семантики є підвищення рівня абстракції при проектуванні, що робить модель більш універсальною. Абстракція в свою чергу - це в створення узагальнених моделей або класів, які представляють сутності в базі даних. Семантичний аналіз допомагає визначити, як саме дані будуть зберігатися та оптимізовані в базі даних. Він включає в себе вибір типів даних, індексування, нормалізацію/денормалізацію та інші аспекти проєктування бази даних. **Мета.** В дослідженні розглянуті питання семантичного аналізу даних при побудуванні реляційних моделей та реляційних баз даних. **Методи дослідження.** В статті поряд з теоретичним аналізом проблеми зібрано та проаналізовано за допомогою семантичного аналізу фактичний матеріал- структури даних з існуючих проєктів та хмарних додатків (кращі практики), зроблені висновки щодо можливостей підтримки семантики засобами існуючих реляційних моделей та реляційних баз даних. **Результати.** Було зроблено опис елементів семантики, проаналізована їхня роль в побудові моделі, виділено ряд семантичних шаблонів, як засобів подолання обмежень реляційної моделі.

Ключові слова: *семантичний аналіз даних, реляційна модель, концептуальні моделі даних, системи баз даних, моделі даних, логічний дизайн бази даних, семантичний аналіз предметної області.*

Як цитувати: Сузікова О. Г. Роль семантичного аналізу в подоланні обмежень реляційної моделі даних. *Вісник Харківського національного університету імені В.Н.Каразіна, сер. «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2024. вип. 62. С.55-69. https://doi.org/10.26565/2304-6201-2024-62-06

How to quote: O. G. Suzikova, " The role of semantic analysis in overcoming the limitations of the relational data model " *Bulletin of V.N. Karazin Kharkiv National University, series "Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp.55-69, 2024. [In Ukrainian] https://doi.org/10.26565/2304-6201-2024-62-06

1. Постановка проблеми

Концепція реляційної моделі була запропонована У.Коддом у 1969 році, це була модель зберігання даних у двомірних таблицях, де основною одиницею був кортеж, кортеж описував певну сутність, а поля кортежу описували атрибути цієї сутності [1]. Модель набула великої популярності саме тому, що була зручною, теоретично обґрунтованою та інтуїтивно зрозумілою. Розробники оцінили її ще й тому, що вона надала фахівцям методологію моделювання, не залежну від деталей фізичної реалізації. Але й по цей час багато розробників вважають, що реляційна модель не пропонує достатньо багатої концептуальної моделі для аспектів предметної галузі, що моделюється, які не відображаються на таблицях напряму[2].

З моменту створення реляційного підходу та концепції реляційних баз даних пройшло вже багато часу, але ще на зорі їх використання дослідники зазначали, що реляційна модель досить обмежена і не може відобразити область діяльності людей, що автоматизується, досить добре. У відповідь на цей виклик були зроблені численні спроби збагатити підхід і прилизувати можливості реляційних баз даних до можливостей об'єкт-орієнтованих мов, які оперують із більш складними структурами і, як результат, діяти на більш високому рівні абстракції.

Виникли об'єктно-орієнтована модель та об’єкт реляційна модель, що мали за мету надати більш досконалі засоби для відображення об'єктів реального світу, ніж реляційна модель[3].[4]. Їх спроби розширити традиційний реляційний підхід базувалися на намаганні реалізувати принципи з наступного списку:

-значеннями полів в таблицях (атрибутів відносин) можуть бути не тільки літеральне значення, а й складні об'єкти;

-значення атрибутів відносин не обов'язково є атомарними ;

-при побудові таблиць може використовуватися механізм успадкування;

-класи (таблиці) можуть включати певні операції (останній принцип вже імплементовано в механізмах вбудованих процедур в конкретних СУБД).

Але тут дослідники зіткнулися з наступною проблемою. Якщо системі управління базою даних закладено можливість оперування складними об'єктами, то процеси пошуку та перетворення інформації у базі даних істотно уповільнюються, оскільки самі об'єкти - складні і важкі. Зі зростанням потужності обчислювальної техніки ситуація покращується, але не радикально. Уповільнення процесів – це плата за складність об'єктів системи.

У той самий час цю проблему програмістам доводиться вирішувати щодня практично. У цьому плані виявився незаслужено забутим семантичний аналіз предметної галузі, який якраз у багатьох випадках дозволяє подолати протиріччя. Під семантичним аналізом мається на увазі змістовний аналіз даних та предметної області, результати якого дають можливість побудувати прийнятну та адекватну модель для реалізації в базі даних.

Метою цього дослідження є визначення ролі семантичного аналізу при подолання обмежень традиційної реляційної моделі даних.

Завданнями, що вирішуються, є вивчення сучасного стану проблеми, описання елементів семантики та засобів аналізу, та виділення семантичних шаблонів, як засобів подолання обмежень реляційної моделі.

2. Вивчення сучасного стану проблеми

Зараз словосполучення "семантичний аналіз" застосовується в основному в контексті великих даних та вирішення задачі аналізу природної мови. Але в даному випадку мається на увазі аналіз будь-якої предметної галузі, її сутностей та зв'язків між ними за змістом.

У відповідь на цей виклик з'явилися семантичні моделі даних. Як вказується в [2], певні розробки можна умовно класифікувати як "семантичні" моделі даних, оскільки їхньою відмінністю є те, що вони намагаються надати більше семантичного вмісту, ніж традиційна реляційна модель.

В центрі семантичного аналізу находиться певна предметна галузь, що автоматизується. Це частина реального світ та практичної діяльності людей, суть моделювання – у виділенні певних важливих об’єктів (сутностей) цієї області та їх аспектів з ціллю подальшого представлення їх таблицями бази даних та використання при побудові автоматизованого процесу. [2]

Структура або схема моделі бази даних утримує в собі як самі об’єкти, так і різні типи взаємозв’язки між ними. Об’єкти зазвичай не статичні, набір атрибутів описує певний стан такого об’єкту. Таким чином моделі задають певні структури даних для операцій моделювання, які використовуються для подальшого маніпулювання об’єктами схеми бази даних. Маніпуляції – це бізнес логіка того програмного продукту, що створюється для автоматизації.

Переваги реляційної моделі в тому, що вона, хоча й не здатні виражати всі зв’язки між об’єктами предметної області, але надає зручні засоби для зберігання та відображення даних фізично.

Для покращення моделі були запропоновані певні ідеї, першою була ідея про фізичну незалежність даних. Дослідники баз даних відчули, що користувач повинен бути вільним від деталей фізичної структури бази даних. Таким чином, користувач міг моделювати дані у спосіб, подібний до людського сприйняття програми.

Друга ідея передбачала захоплення додаткової семантики в процесі моделювання даних. Існуючі моделі були здатні визначати деякі семантики даних [5],[6],[7],[8].

Суть того, що розуміється під семантикою - це використання певних конструкцій та методів для того, щоб виражати певні особливості прикладного середовища, які залишаються поза традиційною реляційною моделлю.

Семантичний аналіз починається з розуміння бізнес-процесів та потреб користувачів системи. Це вимагає співпраці з представниками бізнесу та експертами предметної області для виявлення основних сутностей, взаємозв'язків і вимог до даних[9].

Після розуміння предметної області проводиться ідентифікація потрібних сутностей (об'єктів), їх зв'язків. що включає в себе визначення того, що є основними сутностями і як вони пов'язані одна з одною [9].

Для кожної сутності визначаються її атрибути, які відображають її характеристики та властивості. Це важливо для визначення як інформації буде зберігатися в базі даних. Це те, що відноситься до традиційної реляційної моделі даних "Entity-Relationship"[6],[9].

За останні кілька років новою парадигмою опису бізнес-процесів, у тому числі і для інформаційних систем, стала методологія, орієнтована не просто на сутності предметної галузі, а на артефакти бізнесу. Отже, бізнес-процеси описуються з погляду артефактів, якими бізнес маніпулює під час інформаційного обміну[10].

Інформація в системі це дані, визначені через артефакти, керовані бізнесом, разом з обмеженнями, що накладають умови на ці дані та які безпосередньо випливають із вимог предметної галузі організації[11].

Артефактно-орієнтоване моделювання ставить артефакти (значущі бізнесу об'єкти даних) та їх життєві цикли у фокус моделювання будь-яких бізнес-процесів. Цей тип моделювання за своєю суттю є декларативним: потік управління бізнес-процесом взагалі не моделюється, а випливає із життєвих циклів артефактів[10].

Зв'язування бізнес-процесів та їх даних, як і раніше, залишається відкритою проблемою, особливо при розгляді концептуальних моделей. Спільний розгляд моделей процесів та даних на концептуальному рівні дає багато переваг. зокрема, покращує розуміння всього інформаційного процесу в цілому, незалежно від інструментів, які використовуються для створення інформаційної системи[12].

Однією з причин, через яку артефактно орієнтована парадигма полегшує моделювання даних, є здатність моделювати відносини між об'єктами різної потужності.

Іншими словами семантичний аналіз дозволяє визначити більш складні відношення та взаємодії між сутностями, включаючи класифікації, агрегації (складні об'єкти, які містять інші об'єкти), асоціації (зв'язки між сутностями). тощо.

Також семантичний аналіз допомагає визначити, як саме дані будуть зберігатися та оптимізовані в базі даних. Це включає в себе вибір типів даних, індексування, нормалізацію/денормалізацію та інші аспекти проектування бази даних.

Ще одна функція семантичного аналізу - він допомагає визначити правила та обмеження, які забезпечують цілісність та безпеку даних в базі даних. Наприклад, обмеження вставки, видалення, зміни. Якщо об’єкти пов’язані через зв’язки, то вставка, видалення або модифікація одного об’єкта вплине на статус існування інших об’єктів, пов’язаних із ним.

Семантичний аналіз є важливою частиною проектування бази даних, оскільки він допомагає створити структуру даних, що точніше відображає реальну предметну область та більше відповідає потребам бізнесу та користувачів. Він також сприяє покращенню ефективності та продуктивності системи у подальшому використанні, полегшує поширення чи портування системи у майбутньому.

3. Елементи семантики для кращого розуміння структури даних

Ще в ранніх статях, що друкувались у 80-90х роках дослідники намітили компоненти семантики для майбутнього вдосконалення моделей даних. Ціллю вживання компонентів семантики є підвищення рівня абстракції при проектуванні, що робить модель більш універсальною. Абстракція в свою чергу - це в створення узагальнених моделей або класів, які представляють сутності в базі даних.

Абстракція дозволяє виділити головні характеристики сутностей та приховати деталі внутрішнього представлення. Абстракція у складних процесах чи системах дозволяє звести нетипові об'єкти або поведінку до типового, за рахунок чого модель суттєво спрощується[13].

Використання абстракцій дозволяє покращити модель даних і полегшити розуміння структури бази даних. Крім того, це дозволяє імплементувати більш зрозумілі та керовані інтерфейси для користувачів бази даних.

Наприклад, в статті Шміда та Свенсона [6] описані зв’язки та асоціації, а також пов’язана з ними семантика. В статті Сміта та Сміта [7] описаний шар абстракції в моделях, що використовувалися для моделювання бази даних. Тобто такі аспекти семантичного аналізу даних як абстракції, асоціації, класифікації, узагальнення та агрегації були відомі та описані теоретично вже досить давно. І саме вини є складовими тих моделей, що підпадають під категорію семантичних моделей даних.

Агрегація — включає об'єднання кількох об'єктів або сутностей в одну більш складну сутність. Це дозволяє моделювати комплексні відносини між даними та забезпечує більше структуровану та легко розуміючу базу даних. Це засіб, за допомогою якого зв’язки між типами низького рівня можна вважати типом вищого рівня. Реляційна модель даних може використовувати цю концепцію, наприклад, шляхом агрегування атрибутів для формування відношення. Таким чином семантичне моделювання даних дозволяє агрегувати типи сутностей (або відносини) для формування сутностей вищого порядку. За рахунок цього агрегація допомагає краще розуміти логічну структуру даних та розширює можливості моделювання складних об'єктів та їх взаємозв'язків.

Узагальнення —вказує на відносини між сутностями або класами, де одна сутність є більш загальною (батьківською), а інша - більш конкретною (дочірньою). Це засіб, за допомогою якого відмінності між подібними об’єктами ігноруються для формування типу вищого порядку, у якому можна підкреслити подібність. Узагальнення створює ієрархію, де батьківська сутність має загальні атрибути, які спільні для всіх дочірніх сутностей. Вона в цілому допомагає моделювати спадкування в базі даних, де конкретні сутності успадковують атрибути та характеристики від батьківської сутності. Це дозволяє використовувати загальні атрибути для всіх дочірніх сутностей.

Класифікація —включає в себе розділення сутностей або об'єктів на категорії чи класи на основі спільних характеристик. Це форма абстракції, у якій набір об’єктів вважається класом об’єкта вищого рівня. Класифікація дозволяє групувати подібні сутності разом та визначати загальні атрибути та властивості для кожної категорії. На практиці це допомагає створити більш організовану та структуровану базу даних, де сутності групуються за їхніми спільними характеристиками. Це полегшує пошук та аналіз даних.

Асоціація — асоціативні зв'язки вказують на взаємозв'язки між різними сутностями або об'єктами в базі даних. Це форма абстракції, у якій зв’язок між об’єктами-членами вважається об’єктом набору вищого рівня (Brodie, 1984)[14].Вона дозволяє представити, як об'єкти взаємодіють та співпрацюють один з одним. На практиці це сприяє створенню більш зрозумілих та реалістичних моделей даних[15].

Залежності – ще один важливий елемент семантики. Функціональні залежності - це концепція, яка визначає взаємозв'язок між атрибутами в базі даних. Вони грають важливу роль у визначенні того, як дані будуть зберігатися та організовані в таблицях.

Визначення функціональних залежностей - це один з видів аналізу семантики даних. Він є основою процедури нормалізації. Важливо правильно визначити функціональні залежності для кожної таблиці та атрибута в базі даних, оскільки це визначає структуру та забезпечує правильність операцій з базою даних.

Саме змістовний аналіз об'єктів дозволяє повісити рівень абстракції у програмі, що в результаті робить її більш компактною, універсальною, полегшує розширення функціоналу, портування та зміну коду. Оскільки семантика зв’язків втілена в операціях об’єднання, які явно вимагаються в реляційному запиті, запит буде мати більш компактний вираз у семантичній моделі.

Семантичні моделі даних зазвичай є розширеннями, тобто більш повними моделями в тому сенсі, що користувач здатний витягти повний спектр інформації з бази даних так само легко, як і в попередніх моделях. Однак усі ці моделі також забезпечують економію вираження, яку можна вважати сильнішою за повноту в такому сенсі: користувач не тільки може витягти точно таку саму інформацію, але більшу частину цієї інформації можна витягнути з більшою легкість.

Наприклад, з реляційною моделлю користувач повинен знати про атрибути, задіяні в неявному визначенні міжреляційних зв’язків, і виконувати складні операції саме над цими атрибутами, використовуючи проекції та об’єднання, щоб отримати інформацію через ці зв’язки. Якщо семантика відношення врахована в структурі даних, процес суттєво полегшується. У семантичній моделі операції чітко визначені у зв’язках. Таким чином, семантика вбудована в саму модель даних.

Підтримка цілісності. Традиційні моделі змушують користувача відстежувати зв’язки між об’єктами бази даних або підтримувати внутрішньо об’єктну узгодженість за допомогою навігації зв’язків на фізичному рівні. Семантичні моделі забезпечують механізми для визначення обмежень цілісності та водночас дозволяють користувачеві переглядати дані на рівні, відділеному від структури запису низького рівня[16].

4. Покращення реляційної моделі через абстракції

Абстракції представляють собою загальні та спрощені моделі або класи об'єктів, які можуть бути складними та різноманітними в реальному світі. Абстракції відкидають деталі та зосереджуються на ключових характеристиках.

Застосування абстракцій у реляційній моделі:

У реляційній моделі бази даних абстракції можуть бути представлені як окремі таблиці, де кожен рядок таблиці відображає конкретний об'єкт абстракції, а стовпці представляють атрибути абстракції.

Абстракції дозволяють спростити структуру даних, зробити її більш зрозумілою та дозволяють здійснювати операції над об'єктами абстракції на більш високому рівні абстракції.

Приклад використання абстракцій.

Розглянемо приклад створення абстракції для "користувача" в межах корпоративного порталу з великою кількістю ролей. Замість включення всіх можливих атрибутів користувача (ім'я, адреса, телефон, електронна пошта, тощо) в одну таблицю, можна створити абстракцію User. Ця абстракція міститиме базові атрибути, які є спільними для всіх користувачів, та можливо, посилання на інші таблиці для більш докладних даних. Це дозволить підтримувати однакову структуру для всіх користувачів і легше розширювати систему в майбутньому.

Значення використання абстракцій:

- Спрощення моделювання складних концепцій у реляційній моделі.
- Зменшення кількості дублювання даних та забезпечення їхньої єдності.
- Більша зрозумілість бази даних та легкість у підтримці.
- Підтримка вищого рівня абстракції для операцій над даними, що в цілому покращує продукт.

Використання абстракцій допомагає покращити якість та розуміння структури реляційної бази даних, зробити її більш модульною та підготовленою до розширення, що важливо при проектуванні баз даних для складних концепцій і систем.

Використання агрегацій та композицій

Агрегація в реляційній базі, як вже відмічалось, вказує на створення зв'язків між об'єктами, де один об'єкт (батьківський) містить в собі інші об'єкти (дочірні) як частини або підкомпоненти. Цей тип зв'язку дозволяє структурувати дані та представляти складні об'єкти як агрегати із відомими властивостями та залежностями.

Виділяють також композицію. Композиція - це особливий вид агрегації, де дочірні об'єкти пов'язані з батьківським об'єктом таким чином, що вони існують лише в межах цього батьківського об'єкта. Якщо батьківський об'єкт видаляється, то і всі його дочірні об'єкти також видаляються.

Використання агрегацій та композицій допомагає краще моделювати складні відносини між об'єктами та їхніми частинами в реляційній базі даних. Наприклад, якщо ви моделюєте автомобіль, агрегація дозволяє вам представити двигун, колеса, крісла та інші компоненти автомобіля як частини цього автомобіля.

Агрегація та композиція також грають важливу роль в моделюванні ієрархії підкласів в базі даних. Вони дозволяють структурувати дані та створювати зв'язки між батьківськими та дочірніми класами.

Наприклад, якщо у вас є базовий клас "Авто" і підкласи, такі як "Двигун" та "Ходова частина", агрегація допомагає представити зв'язок між "Авто" і "Двигун" де "Авто" є батьківським класом, а "Двигун" є дочірнім класом.

Значення використання агрегацій та композицій:

- Структурування та моделювання складних відносин між об'єктами в реляційній базі даних.
- Підтримка ієрархічних структур та відносин між батьківськими та дочірніми класами в моделях даних.

• Забезпечення можливості представляти складні об'єкти та їхні відносини в базі даних з точністю до деталей.

• Агрегація та композиція є важливими інструментами при моделюванні структури даних в реляційних базах даних, особливо при роботі з об'єктами, які мають складні відносини та ієрархії.

Крім того, до засобів семантичного аналізу можна віднести знання про структуру та складність об’єкта та функціональні залежності в системі атрибутів. Функціональні залежності, наприклад, широко використовуються для вдосконалення бази зданих за процедурою нормалізації, така процедура дозволяє запобігти дублюванню інформації. Зворотна процедура денормалізації навпаки дзволяє дублювати потрібні дані, якщо це відповідає вимогам проекту.

Використання класифікації та узагальнення

Як вже відмічалось, класифікація – це розбиття об’єктів на групи за певними характеристиками. У системі керування персоналом можна використовувати класифікацію для групування працівників за функціональними ролями, наприклад, "менеджери," "розробники," "QA-інженери" і т.

Узагальнення може бути використане для моделювання ієрархії, де є певна батьківська сутністю, а її дочірні сутності зі спільними атрибутами. Класифікація може бути використана для групування продуктів у магазині за їхніми категоріями, підкатегоріями тощо (наприклад, "побутова техніка"> "техніка для кухні"> "дрібна техніка для кухні").

Класифікація та узагальнення допомагають краще організувати та моделювати дані в базі даних, створюючи логічні зв'язки та ієрархії між сутностями. Це полегшує розуміння та управління даними в системі.

Значення використання класифікації та узагальнення:

• Спрощення структури даних шляхом групування сутностей та атрибутів за спільними ознаками.

• Зменшення дублювання даних та сприяння консистентності в базі даних.

• Можливість використовувати поліморфізм для роботи з різними класами через їх загальний суперклас.

• Забезпечення більшої гнучкості та розширюваності бази даних при зміні вимог до системи.

Використання асоціативних зв'язків

Асоціативні зв'язки в базах даних грають важливу роль у відображенні складних відносин між об'єктами. Ці зв'язки дозволяють з'єднувати два або більше об'єктів у базі даних та визначати їхні взаємовідносини. Це особливо корисно в ситуаціях, коли об'єкти мають складну систему відносин (тобто не один зв’язок).

Асоціативні зв'язки, наприклад, можуть містити додаткову інформацію про сам зв'язок. Наприклад, в асоціативному зв'язку між "Користувачами" і "Товарами" може бути включено інформацію про номер та дату ліцензиії або номер гарантійного зобов’язання, та граничну дату дії.

Асоціативні зв'язки можуть бути використані для вирішення різних завдань в базі даних:

Зв'язки багато-до-багатьох: асоціативні зв'язки часто використовуються для моделювання відносин багато-до-багатьох між об'єктами[10]. Наприклад, в базі даних для інтернет-магазину може бути асоціативний зв'язок між таблицею "Користувачі" і "Товари", де кожен користувач може мати багато товарів у своєму кошику, і кожен товар може бути у кошику декількох користувачів.

Запити і фільтрація: асоціативні зв'язки дозволяють виконувати складні запити, щоб знайти певні зв'язки або фільтрувати дані на основі цих зв'язків.

Відносини при семантичному моделюванні аналізуються з точки зору їх змісту та призначення. Концептуально конструкт відношення може відображатися в моделі як атрибут, сутність, незалежний елемент або функція. Відношення, втілене атрибутами, — це зв’язок, у якому атрибут одного об’єкта пов’язаний з іншим об’єктом, вказує на нього або походить від нього. Це традиційний підхід.

Альтернативно відносини також можна розглядати як незалежні об’єкти, відмінні від сутностей. Це не означає, що фізична база даних представляє зв’язки та сутності з різними структурами, але принаймні на концептуальному рівні сутності розглядаються окремо від зв’язків.

Зв’язок представляється як сутність, якщо зв’язок двох або більше об’єктів концептуально описує окремий об’єкт моделі.

Значення використання асоціацій:

- Можна відтворити більш складні зв’язки ("багато до багатьох")
- Можна передати додаткову інформацію
- Можна відтворити кілька типів зв’язку між тими самими об’єктами
- Це робить операції з базою даних більш ефективними і швидкими.
- Дозволяють уникнути дублювання даних та зберігати відносини між об'єктами зручним способом.

Використання функціональних залежностей

Функціональні залежності допомагають визначити, які атрибути залежать від інших, і які обмеження можна застосовувати до цих залежностей. Вони допомагають визначити ключові атрибути в таблицях бази даних. Ключові атрибути визначаються як ті, від яких залежать інші атрибути, і які можна використовувати для унікальної ідентифікації записів в таблиці.

Припустимо, у вас є таблиця "Замовлення" (Orders) у базі даних, і ця таблиця містить наступні атрибути:

- order_id (ідентифікатор замовлення)
- customer_id (ідентифікатор клієнта, який зробив замовлення)
- order_date (дата замовлення)
- lineitem_id (це рядок, що утримує товар, що замовляється, його кількість та ціну)
- total_amount (загальна сума замовлення)

У цій таблиці існує функціональна залежність між "order_id" та "customer_id". Це означає, що кожне "order_id" залежить від конкретного "customer_id". Іншими словами, кожен замовник (customer) може мати багато замовлень, і для кожного замовлення існує конкретний клієнт (customer).

Очевидно, що існує функціональна залежність між "order_id" та "order_date", бо кожне замовлення має свою унікальну дату замовлення.

Крім того, "total_amount" залежить від "order_id". Кожне замовлення має свою власну загальну суму.

Також "total_amount" залежить від "lineitem_id". Кожна сума замовлення формується за рахунок виду товару, ціни та кількості.

Ці функціональні залежності визначають взаємозв'язок між різними атрибутами в таблиці "Замовлення". Вона показує, як атрибути залежать один від одного та допомагають відобразити взаємозв'язок між клієнтами, замовленнями та їх даними в базі даних.

Функціональні залежності грають ключову роль в процесі нормалізації даних. Нормалізація полягає в розбитті таблиць на менші та більш атомарні структури для зменшення дублювання та заборони аномалій даних. Функціональні залежності визначають, які атрибути повинні бути в окремих таблицях для забезпечення нормалізації.

Розуміння функціональних залежностей допомагає покращити оптимізацію запитів в базі даних. Якщо ви знаєте, які атрибути залежать від інших, ви можете ефективніше створювати запити та індекси для прискорення доступу до даних.

Також функціональні залежності допомагають уникнути неправильного проектування бази даних та ризику помилок у даних. Вони надають ясний варіант того, як атрибути пов'язані між собою.

Значення використання функціональних залежностей:

- Зменшують кількість помилок
- Допомагають уникнути аномалій даних, таких як дублювання або втрати даних
- Дозволяють обмеження можливостей вставки, оновлення та видалення даних в таблицях
- Грають ключову роль в процесі нормалізації даних
- Допомагають ефективніше створювати запити та індекси для прискорення доступу до даних.

5. Семантичні шаблони як засоби подолання обмежень реляційної моделі з прикладами з проектів.

Патерни проектування баз даних - це загальні, перевірені часом рекомендації та рішення для моделювання та розробки баз даних. Як і в інших сферах програмування, семантичні патерни проектування баз даних служать як шаблони або рекомендації, які допомагають вирішувати типові завдання та проблеми при розробці та керуванні базами даних.

Основна ідея застосування патернів проектування баз даних полягає в тому, щоб використовувати так звані "best practice" вже перевірені та ефективні способи роботи з даними, замість того, щоб винаходити їх заново для кожного проекту. Саме патерни та шаблони пропонують шляхи подолання обмежень традиційної реляційної моделі даних. Поряд з цим патерни проектування допомагають забезпечити більшу стабільність, підтримуваність та розширюваність бази даних, а також знизити ризик помилок та неправильного проектування.

Одним з найбільш відомих дослідників патернів при проєктуванні реляційних баз даних є Мартін Фаулер [17]. Він виділив більш ніж 50 патернів для реляційних баз даних, що дозволяють покращити відображення предметної області.

Серед найбільш відомих його патернів є патерни для наслідування та відтворення асоціативних зв'язків.

Патерн "Наслідування конкретної таблиці"

Реляційні бази даних не підтримують успадкування – факт, який ускладнює об'єктно-реляційне відображення. Цей патерн використовує узагальнення для створення ієрархії таблиць в базі даних. В Concrete Table Inheritance існує таблиця для кожного конкретного класу в ієрархії успадкування, та кожна з них представляє собою конкретний об'єкт.

На основі шаблонів Фаулера можна створити в базі даних певні аналоги об'єктів та класів, що властиві об'єкт орієнтованим мовам програмування. Назвемо їх шаблонами.

Шаблон "Суперклас"("Superclass")

Цей підхід має на меті використання суперкласу для узагальнення спільних атрибутів, які є характерними для декількох класів. Наприклад, якщо у вас є класи "Користувачі" та "Команда" ви можете створити суперклас "Користувач", який містить загальні атрибути, такі як ім'я та електронна пошта, і нащадки цього класу будуть мати додаткові атрибути, наприклад, позиції для членів команди.

Приклад з проєкту Zendesk, де імплементовано суперклас User, до якого входять три підкласи організовані за різними ролями в системі (що означає, що в них різна логіка в системі) - це Admin, Agent, End-User. При чому адміністратори та агенти – це члени команди, а кінцевий юзер – це клієнт.

Всі дані про структуру об'єктів, їх поля та атрибути отримані за допомогою сервісу https://skyvia.com/

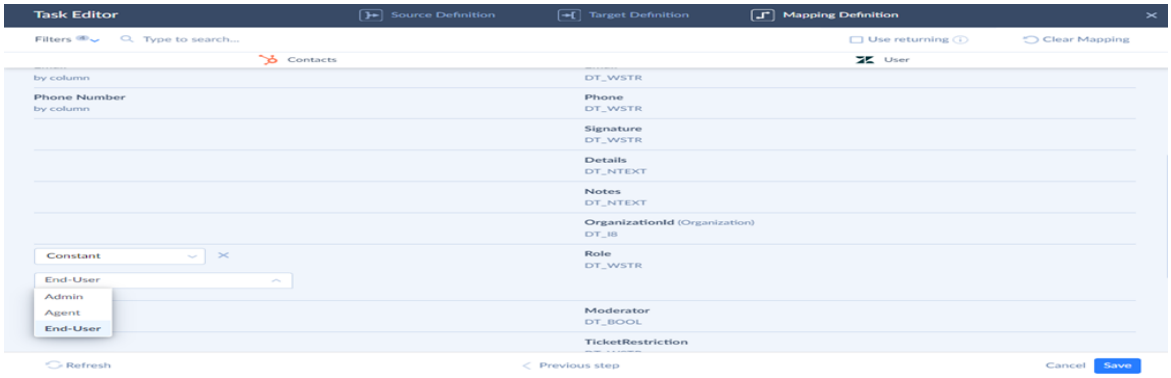


Рис.1.Імпорт в Zendesk, з вибором підкласу юзерів за роллю з суперкласу User.
Джерело: https://skyvia.com/

Множинне успадкування — це механізм, за допомогою якого об'єктам в ієрархії узагальнення/спеціалізації дозволяється успадковувати властивості від кількох об'єктів вищого рівня (наприклад, вони не можуть бути створені без ідентифікаторів об'єктів-батьків). Це зручно для деяких об'єктів, коли властивості, що успадковуються, не перетинаються. Проблема виникає, коли один спеціалізований об'єкт успадковує ту саму властивість від двох об'єктів вищого рівня.

В цьому випадку семантична модель може дозволяти або забороняти використання множинного успадкування, або пропонувати певні вбудовані механізми для обробки конфліктів, які можуть виникнути.

Приклад з платіжної системи Square. Сутність Invoice не може бути створена без попереднього створення сутностей Order та PaymentRequest. При чому інвойс (Invoice) від замовлення (Order) бере властивості, що пов'язані з товарами, кількістю, цінами, тобто з товарним наповненням

інвойсу. А від PaymentRequest успадковує параметри, що пов'язані з засобами та порядком оплати [18].

Патерн "Наслідування таблиці", зустрічається також з назвою "Table Inheritance", "Single Table Inheritance"

Цей патерн також використовує узагальнення та класифікацію для створення ієрархії таблиць в базі даних. На кожен клас або об'єкт створюється окрема таблиця, а головна таблиця містить загальні атрибути. Це дозволяє моделювати класи зі спільними характеристиками та використовувати агрегацію для посилання на окремі таблиці, які містять специфічні атрибути для кожного класу.

На основі цього патерну, наприклад, можна створити шаблон абстрактного класу.

Шаблон "Абстрактний клас" ("Abstract Class")

Абстрактний клас в об'єктних мовах програмування – це клас, що не може мати конкретних екземплярів, але екземпляри можуть бути в його класів-нащадків. Приклад - клас "Тварини", просто тварини в реальному світі не існують, але існують конкретні тварини, що відносяться до його класів-нащадків - Кішок або Собак.

Абстрактний клас в реляційній базі даних можна імплементувати через певні поля, що пов'язують батьківський клас та класи-нащадки. При чому, на відміну від просто наслідування таблиця, що відповідає абстрактному класу може не мати атрибутів крім ключового атрибуту (назви та/або ідентифікатора).

Приклад з реального проекту – платіжна система Square. В системі імплементовано абстрактний клас CatalogItem – це товари, продукти з каталогу, абстрактний продукт не має артікулу - SKU, ціни та валюти продажу, а ось його дочірня структура CatalogItemVaration має артікул -SKU, ціну та валюту, та не може бути створена без Id батьківської структури.

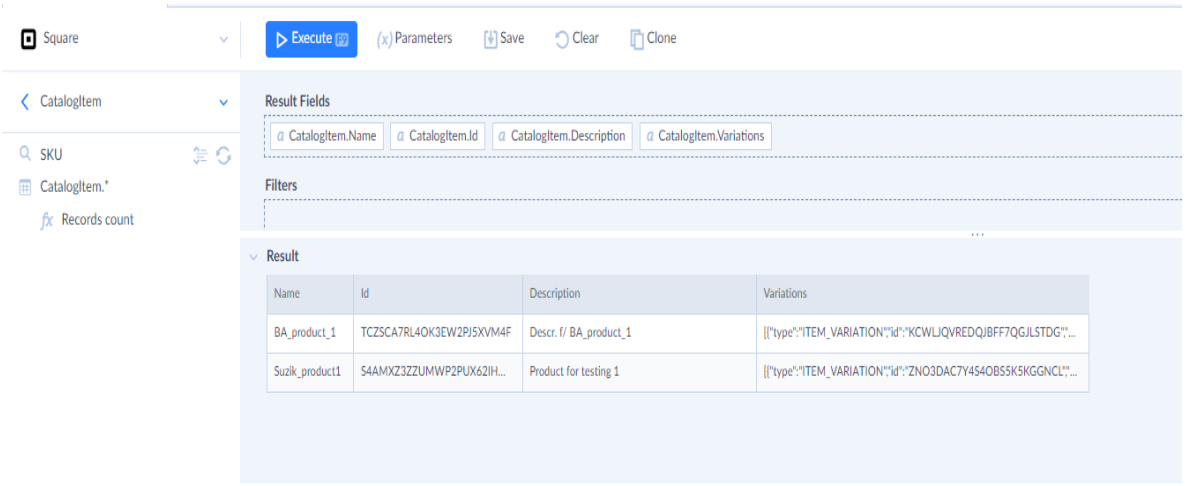


Рис 2. Абстрактний клас CatalogItem з платіжної системи Square (не має артікулу)
Джерело: https://skyvia.com/

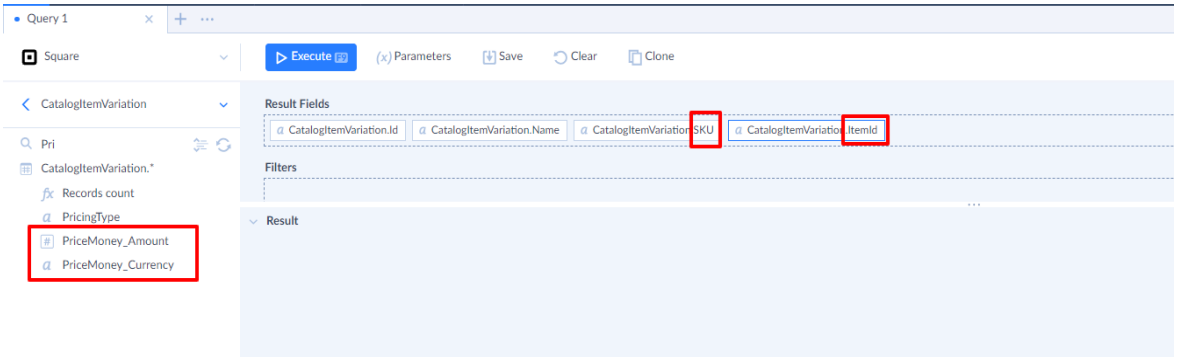


Рис 3. Клас CatalogItemVariation з платіжної системи Square (має артікул, ціну та валюту продажу), що не може бути створений без ідентифікатора батьківського об'єкту CatalogItem
Джерело: https://skyvia.com/

Патерн "Наслідування класу таблиці" ("Class Table Inheritance")

Цей патерн використовує агрегацію для створення ієрархії таблиць, при чому кожна таблиця представляє конкретний клас або об'єкт. В цьому випадку, агрегація дозволяє зв'язувати таблиці з загальними атрибутами, що спільні для всіх класів (наприклад, через ID об'єкта).

На основі цього патерну можна створити такі структури, як Агрегат та Компонент.

Шаблон "Агрегат" ("Agregate")

Це рішення має на увазі використання агрегації для створення складних але цільних структур даних, які включають у себе різні класи. Наприклад, агрегація може бути використана для створення складних об'єктів, що містять в собі інші об'єкти, де кожен об'єкт може належати до різних класів. Зв’язки відтворюється за допомогою зовнішніх ключів.

Прикладом Агрегату може бути сутність Country(країна) в екомерс системі Shopify, Країна утримує в собі незалежні Shipping Zone – зони постачання, від приналежності до певної зони залпежить вартість постачання. Як зони, так і країна можуть бути використані незалежно, і це реальні об’єкти предметної області.

Шаблон "Компонент"("Component")

Цей підхід використовує композицію для створення складних об'єктів, які складаються з менших компонентів. Кожен компонент може мати свої атрибути та пов'язані дані але обов’язково має посилання на складний об’єкт (наприклад, через ID об'єкта). В базі даних, композиція дозволяє структурувати дані об'єктів і їх компонентів у реляційній моделі. Особливості компонента - він існує не сам по собі, а лише як частка іншого об’єкту.

Особливості агрегату та компоненту такі, агрегат - це цільна структура, яка містить в собі колекцію об'єктів та має незалежний життєвий цикл, тоді як компонент - це частина цієї структури, яка існує та функціонує лише в контексті агрегата.

Прикладом компоненту, що сам по собі не існує може бути сутність Line, що використовується у багатьох платіжних та бухгалтерських системах (Stripe, Square, Xero, Quickbooks). Ця сутність представляє собою окремий рядок інвойсу. Сама по собі вона не існує, а є тільки часткою документу.

Патерн "Зіставлення Таблиці Асоціацій" (Association Table Mapping)

Ассоціативні зв'язки в реляційних базах даних відображають відносини між сутностями, коли ці відносини містять додаткову інформацію.

Цей патерн використовує окрему таблицю, яка містить зв'язки між двома або більше сутностями, а також можливі атрибути цього зв'язку. Наприклад, якщо ви маєте сутності "Користувач" і "Продукт," і ви хочете відстежувати взаємозв'язок між користувачами та продуктами, ви можете створити таблицю "UserProduct" з власним ідентифікатором зв’язку та полями, які вказують на користувача, продукт і, можливо інші дані, наприклад, серійний номер продукту, номер ліцензії або гарантійний термін.

Шаблон "Зв’язок як сутність"

В цьому шаблоні створюється спеціальна сутність (або сутності. Якщо є кілька типів зв’язків) для того, щоб здійснювати поєднання двох інших сутностей.

Приклад імплементації з проєкту Hubspot CRM. Раніше (в попередніх версіях API) зв’язки між сутностями були в системі імплементовані як поля, що утримують посилання на одну чи кілька пов’язаних сутностей. У випадку кількох таких зв’язків поле утримувало рядок з вмістом масиву ідентифікаторів. Зараз в поновленій версії API з’явився новий тип об’єктів – Association (Ассоціація), за допомогою цієї сутності можна поєднати два будь які об’єкти в системі, при чому між тими ж двома об’єктами може бути встановлено багато типів різних зв’язків.

Приклад такого зв’язку з системи Hubspot. Сутність CompanyEngagements зв’язує компанію та активності, щоб залучити її до числа клієнтів. Вона має власний ідентифікатор та посилання на сутності, які вона поєднує.

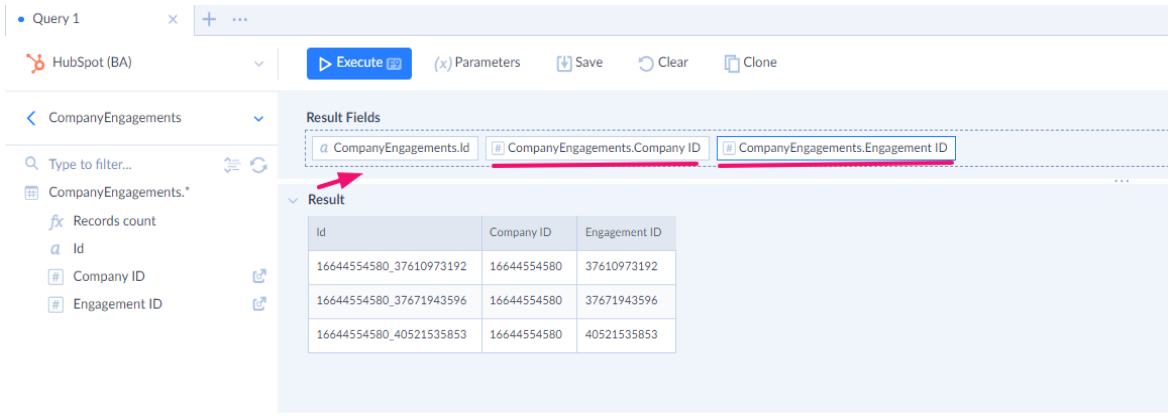


Рис 4. Таблиця CompanyEngagements Hubspot CRM
Джерело: https://skyvia.com/

Тобто, реалізація асоціативних зв'язків може варіюватися в залежності від специфікацій бази даних та потреб проекту.

Шаблон "Інкапсуляція" (Incapsulation)

Інкапсуляція, що є базовим принципом ООР, прямо суперечить принципам відкритості та вивільненості комбінування значень у кортежах SQL, що створює суттєві бар'єри для ефективного використання реляційної бази, як довготривалого сховища складених об'єктів класу (можливо, що це випливає із певних теоретичних неузгодженостей) .

Але рішення можливе, воно має на увазі створення додаткового об'єкту для заміни оригінального для того, щоб обмежити права доступу. Сервіс, який звертається за даними, отримує тільки ті дані, що відкриті для нього, а не до всіх. Фактично – це створення тимчасових або постійних таблиць, що утримують обмежену інформацію, що потрібна для користувача, при чому вся зайва інформація прихована та не є доступною.

Приклад з реального проєкту. В емейл-маркетинг системі Mailchimp нема прямого доступу до таблиці юзерів або контактів системи. Можливе звернення тільки до сутності ListMembers, це підписники з певного листу розсилок.

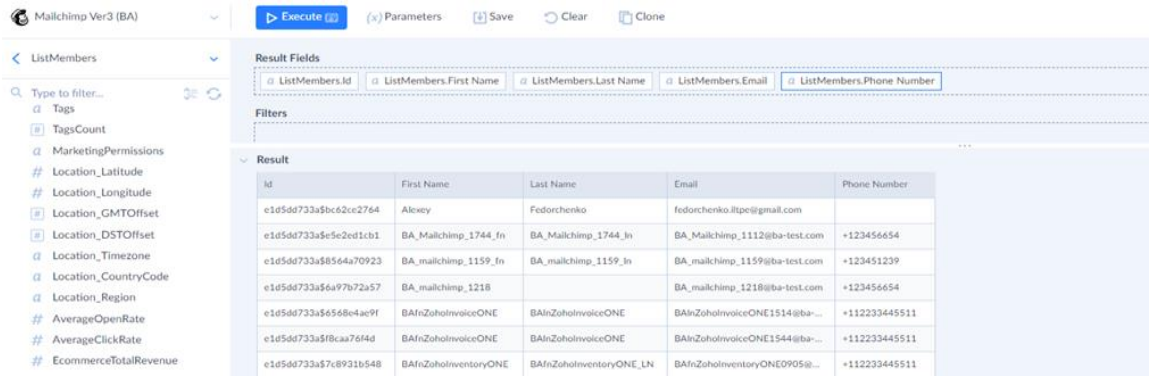


Рис 5. Вибірка з таблиці ListMembers з системи Mailchimp API V3.
Джерело: https://skyvia.com/

Для таблиць з функціональними залежностями широко відомими є підходи з нормалізації чи денормалізації таблиць, що зумовлено завданнями, що вирішуються.

Нормалізація та денормалізація

Відповідно до потреб певного ресурсу, що проєктується, ви можете вибрати паттерни нормалізації для зменшення дублювання даних або денормалізації для покращення швидкодії запитів. Нормалізаціія видвляє функціональну залежність між полями таблиці. Денормалізація створює певну функціональну залежність в системі даних.

Нормалізація включає в себе розбиття таблиць на менші та більш атомарні структури, за допомогою визначення первинного ключа (Primary Key) для кожної таблиці та встановлення зовнішніх ключів (Foreign Keys) для вираження залежностей між таблицями. Цей процес вимагає

аналізу структури даних та їх взаємозв'язків для визначення оптимальної організації даних в базі даних[18].

Нормалізація може бути поділена на кілька рівнів (результатом буде певна нормальна форма, від першої нормальної форми (1NF) до п'ятої нормальної форми (5NF)). Кожен рівень нормалізації має на увазі певні правила та вимоги щодо організації даних в таблицях.

Денормалізація – в противагу, це процес збільшення кількості дубльованих та зайвих даних у базі даних для підвищення ефективності запитів та покращення продуктивності системи. Вона протистоїть нормалізації, яка мінімізує дублювання даних та забезпечує консистентність даних.

Денормалізація застосовується у випадках, коли швидкість операцій з базою даних стає важливішою за ефективність витрат ресурсів на збереження даних.

Ці шляхи допомагають вдосконалити реляційні структури даних, роблять їх більш гнучкими та ефективними для моделювання різних відносин та ієрархій в базі даних. Розуміння цих паттернів допомагає проектувати більш ефективні та модульні бази даних.

Нормалізація – це доволі розповсюджена процедура при проєктуванні баз даних. Денормалізація більш рідкісна операція. Наведу приклад навмисно зробленої денормалізації:

Hubspot CRM дублює інформацію про асоціативні зв’язки між сутностями, наприклад, при створенні асоціації між компанією та квитком техпідтримки, фактично одна і та ж інформація складається в дві симетричні таблиці CompanyTickets та TicketCompanies для того, щоб прискорити пошук зв’язків.

В реальних проектах можна знайти менш фундаментальні, але не менш корисні рішення та лайфхаки для вирішення практичних завдань. Деякі з них до речі дозволяють обійти обмеження традиційних реляційних баз даних на атомарність полів та використовувати складні об’єкти в якості атрибутів.

Шаблон "Масив" ("Array").

Масив передається в поле як рядок та при отриманні розбирається за допомогою процедури парсінгу.

Приклад з системи Hubspot – сутність Deals має поля-рядки, що містять масиви ідентифікаторів.

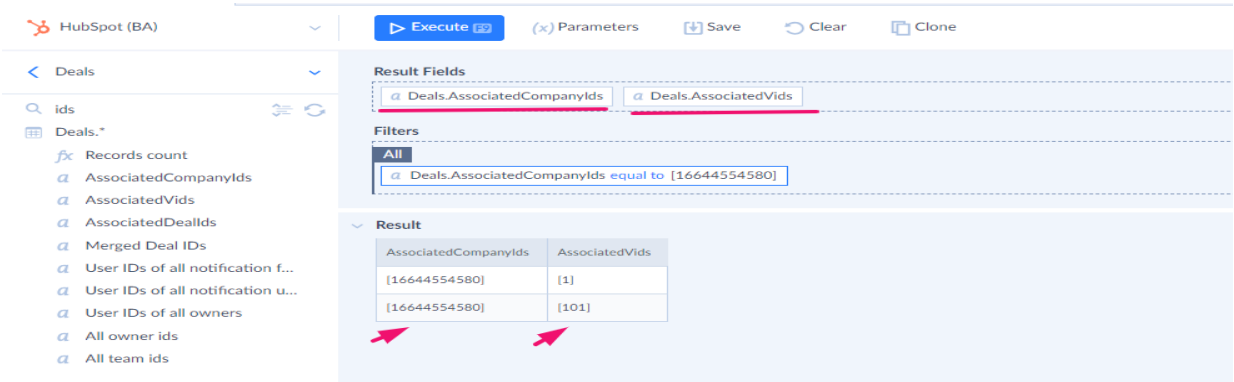


Рис 6. Масиви, що передаються в поля як рядки. Hubspot CRM.
Джерело: https://skyvia.com/

Шаблон "Складний атрибут"(мається на увазі робота з атрибутом – об’єктом з структурою JSON, XML, HTML).

Одною з проблем реляційної моделі є те, що вона не дозволяє використовувати атрибути-сутності, тобто не атомарні складні об’єкти. В деяких випадках, особливо для серійних складних структур даних, використовують структури JSON, XML для зберігання характеристик серійних об’єктів, або конструкції HTML для зберігання певних шаблонів. В цьому випадку атрибути-сутності представлені у вигляді такої структури можна помістити в рядок в одному зі стовпців таблиці.

Тобто, складний атрибут що по суті є конструкцією JSON, XML, HTML передається в поле у вигляді звичайного рядка, але впорядкованого по шаблону. Для відновлення об’єкту рядок розбирається по цьому шаблону.

Приклад з реального проекту: на малюнку ви можете бачити структуру складного атрибуту Line типу рядок об’єкту Invoice з хмарного застосунку Quickbooks Online, цей атрибут відповідає за рядки інвойсу, що утримують продукт, його деталі, одиниці виміру, вартість та підсумкову суму рядка.


```
[
  {
    "Id": "1",
    "Amount": 3500.0,
    "LineNum": 1.0,
    "Description": "Custom Design",
    "DetailType": "SalesItemLineDetail",
    "SalesItemLineDetail_ItemRefId": "4",
    "SalesItemLineDetail_ItemRefName": "Design",
    "SalesItemLineDetail_UnitPrice": 350.0,
    "SalesItemLineDetail_Qty": 10.0,
    "SalesItemLineDetail_ItemAccountRefId": "82",
    "SalesItemLineDetail_ItemAccountRefName": "Design income",
    "SalesItemLineDetail_TaxCodeRefId": "NON"
  },
  {
    "Amount": 3500.0,
    "DetailType": "SubTotalLineDetail"
  }
]
```


Рис 7. Вид атрибуту Line (типу рядок) об’єкту Invoice з хмарного застосунку Quickbooks Online.
Джерело: https://skyvia.com/

Шаблон "Важкий об’єкт"

В таблиці розміщується не сам об’єкт, а тільки його адреса. Для того, щоб фільтрувати або шукати певні об’єкти додаються метадані об’єктів за якими ведеться фільтрація, сортування або пошук.

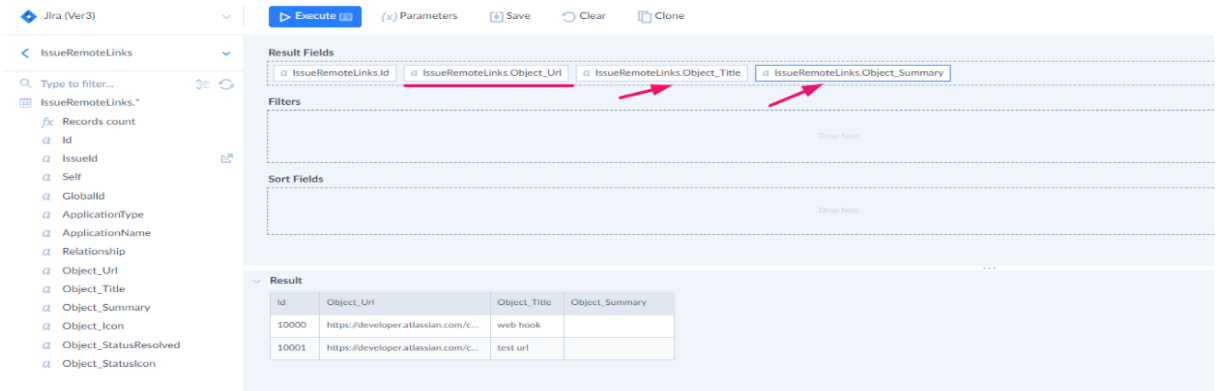


Рис 8. Адреса та метадані важкого об’єкта в Jira..
Джерело: https://skyvia.com/

Приклад з проєкту Jira. В версіях API V2, V3 використано сутність IssueRemoteLinks, що утримує інформацію про різні об’єкти, що відносяться до певної проблеми програмного забезпечення (Issue – це вид завдання, що описує технічну проблему, що треба вирішити). Об'єктами, що відносяться до такого завдання можуть бути файли, зображення, аудіо або відеоматеріали для опису проблеми. Сутність надає адресу такого об'єкту та метадані для пошуку, наприклад, назву та опис.

2 Висновки

При проектуванні баз даних однією з ключових складових є розуміння та відображення предметної області. Це завдання вимагає семантичного аналізу, який дозволяє створити таке відображення якомога більш точним, коректним і ефективним.

В той час як найбільш традиційні моделі даних забезпечують лише один засіб представлення даних. Семантичні моделі даних за допомогою абстракцій дозволяють користувачеві моделювати та переглядати дані на багатьох рівнях. Це надає розширені можливості для моделювання ситуацій "реального світу", оскільки перегляд даних на багатьох рівнях узгоджується з тим, як люди бачать світ.

Семантичний аналіз включає в себе наступні ключові аспекти:

- Використання абстракцій - класифікацій, узагальнень, агрегацій та асоціативних зв'язків - у проектуванні бази даних допомагає покращити структуру даних та зробити її більш зрозумілою та ефективною. Це полегшує роботу з базою даних, сприяє кращому розумінню предметної області та сприяє вдосконаленню якості та продуктивності системи у подальшому використанні.
- Використання знань про структуру об’єкту дозволяє вирішити проблему з атомарністю даних реляційних таблиць та наблизити їх можливості до можливостей об’єктних мов програмування.
- Використання найкращих практик, патернів проєктування, перевірених рішень, що дозволяють обійти певні обмеження реляційної моделі. При використанні таких стандартних підходів до проектування можна створити більш організовану та легко розширювану базу даних, що відповідає потребам вашого проекту та краще описує предметну область.

Вибір конкретного методу реалізації, патерну залежить від потреб користувачів та структури даних у проекті. Кожен з цих підходів має свої переваги та недоліки, і важливо враховувати їх при проектуванні конкретної бази даних.

REFERENCES

1. Codd, E. F. Extending the database relational model to capture more meaning. ACM Trans. Database Syst. 4, 4 (Dec.), 1979, pp.397-434.
2. Joan Peckham, Fred J. Maryanski. Semantic Data Models. ACM Comput. Surv. 20(3).1988, pp.153-189.
3. M. A. Garvey, M. S. Jackson. Introduction to Object-Oriented Database.Inf. and Software Technol.- 31, N 10.1989.pp.521-528
4. Woelk, D, Kim, W., and Luther, W.An object-oriented approach to multimedia databases. In Proceedings of the ACM SIGMOD Conference (Washington, D.C.). ACM, New York,1986, pp. 311-325.
5. Chen, P. 1976. The entity—relationship model: Toward a unified view of data. ACM Trans. Database syst. 1, 1 (Mar.), 1976, pp.9-36.
6. Chen, P., Ed. Entity—Relationship Approach: The Use of the ER Concept in Knowledge Representation. North-Holland, Amsterdam. 1985.
7. Schmid, H. A., Swenson, J. R. 1975. On the semantics of the relational data model. In Proceedings of the ACM SIGMOD Conference (San Jose, Calif.). ACM, New York, pp. 211—223.
8. Smith, J. M., Smith, D. C. P. Database abstractions: Aggregation and generalization. ACM Trans. Database Syst. 2, 2 (Mar.), 1977, pp 105-133.
9. Hey, D.C. Data Model Patterns: Conventions of Thought. David C. Hey. Dorset House Publishing, 1996.288 p.
10. Diana Borrego, Rafael M. Gasca, María Teresa Gómez-López, Automating correctness verification of artifact-centric business process models,т Information and Software Technology,Volume 62, 2015, pp187-197.
11. Xavier Oriol, Ernest Teniente, Simplification of UML/OCL schemas for efficient reasoning, Journal of Systems and Software, Volume 128, 2017, pp 130-149.
12. C. Combi, B. Oliboni and F. Zerbato, "Integrated Exploration of Data-Intensive Business Processes," in IEEE Transactions on Services Computing, vol. 16, no. 1, pp. 383-397, 1 Jan.Feb. 2023

13. David Chapela-Campa, Manuel Mucientes, Manuel Lama, Understanding complex process models by abstracting infrequent behavior, Future Generation Computer Systems, Volume 113, 2020, pp 428-440

14. Brodie, M. L. On the development of data models. In On Conceptual Modeling, Perspectives from Artificial Intelligence, Databases, and Programming languages, M. L. Brodie, J. Mylopouious, and J. W. Schmidt, Eds. Springer-Verlag, New York, 1984, pp. 19-48.

15. Silverstone, L. The data Model Resource Book, Vol. 3: Universal Patterns for Data Modeling . Len Silverston.Wiley Computer Publishing, 2009. 648 p.

16. Ambler, S. Refactoring Databases: Evolutionary Database Design. Scott W. Ambler, Premodkumar J. Sadalage .Addison-Wesley, 2006. 384 p.

17. Fowler, M. Patterns of Enterprise Application Architecture. Martin Fowler Addison-Weatley, 2003. 736 p.

18. Simsion, G.C., Witt, G.C. Data Modeling Essentials, Third Edition .Graeme C. Simsion, Graham C. Witt.Morgan Kaufmann Publishers, 2005.560 p.

Suzikova Olena G.

Candidate of Psychological Sciences. Department of Applied Mathematics
V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine.
e-mail: osuzikova@karazin.ua
https://orcid.org/0009-0002-1305-1256

The role of semantic analysis in overcoming the limitations of the relational data model

Relevance. The article is devoted to defining the role of semantic analysis in overcoming the limitations of the traditional relational data model. The need for semantic data analysis arises from the demand for more expressive and capacious conceptual data models. The problem that has not yet been fully resolved is that classical relational models do not directly support data semantics - relationships, data abstraction, inheritance, polymorphism, encapsulation, complex objects, and dynamic properties of objects. Semantics refers to the use of certain constructs and methods to express features of the application environment that remain outside the traditional relational model. Semantic analysis allows to define more complex relationships and interactions between entities, including classifications, aggregations (complex objects that contain other objects), associations (links between entities), etc. The purpose of using semantics components is to increase the level of abstraction in design, which makes the model more versatile. Abstraction, in turn, is the creation of generalized models or classes that represent entities in the database. Semantic analysis helps determine how data will be stored and optimized in the database. It includes the selection of data types, indexing, normalization/denormalization, and other aspects of database design. **Objective.** The study considers the issues of semantic data analysis in the construction of relational models and relational databases. **Research methods.** Along with the theoretical analysis of the problem, the article collects and analyzes the actual material - data structures from existing projects and cloud applications (best practices) - using semantic analysis, and draws conclusions about the capabilities of existing relational models and relational databases to support semantics**. Results.** We described the elements of semantics, analyzed their role in building the model, and identified a number of semantic patterns as a means of overcoming the limitations of the relational model.

Keywords: *semantic data analysis, relational model, conceptual data models, database systems, data models, logical database design, semantic analysis of the subject area.*

Надійшла у першій редакції 10.05.2024, в останній - 15.06.2024.
The first version has been received on 10.05.2024, the final version - on 15.06.2024.

DOI: https://doi.org/10.26565/2304-6201-2024-62-07

УДК (UDC) 004.8:342.9

Трусов Михайло Андрійович *Ведучий розробник програмного забезпечення, IT-компанія «EPAM Systems», Ukraine, вул. 23 Серпня, 33, Харків, Україна, 61045*
e-mail: trusov.michael@gmail.com

Узлов Дмитро Юрійович *к.т.н., доцент, в.о. декана факультету комп'ютерних наук Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022*
*e-mail:*dmytro.uzlov@karazin.ua
https://orcid.org/0000-00Q3-3308-424X

Перспективи використання моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях

Актуальність: впровадження моделей глибокого навчання для семантичної сегментації на автономних пристроях є перспективним напрямком розвитку інтелектуальних систем, здатних аналізувати візуальну інформацію без постійного підключення до зовнішніх ресурсів.
Метою роботи є дослідження можливостей та викликів використання моделей глибокого навчання для задач семантичної сегментації на автономних пристроях.
Методи дослідження включають теоретичний аналіз, систематизацію та узагальнення використання моделей глибокого навчання в автономних пристроях, а також параметрів, що впливають на обсяг пам'яті моделей, та особливостей імплементації натренованих моделей у власні програмні продукти.
Результати: виявлено значний потенціал технології глибокого навчання для створення автономних інтелектуальних систем. Визначено основні параметри, що впливають на ефективність роботи моделей на пристроях з обмеженими ресурсами. Запропоновано рекомендації щодо імплементації натренованих моделей у програмні продукти. Висвітлено сучасні підходи до шифрування моделей глибокого навчання.
Висновки: подальші розробки в області глибокого навчання для семантичної сегментації на автономних пристроях сприятимуть розвитку більш ефективних та автономних систем для широкого спектру застосувань, включаючи комп'ютерний зір, робототехніку тощо. Забезпечення безпеки доступу до навчених моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях вимагає комплексного підходу, що поєднує апаратні та програмні рішення.
Ключові слова: *глибоке навчання, семантична сегментація, автономні пристрої, оптимізація моделей, вбудовані системи, апаратне прискорення, шифрування даних.*

Як цитувати: Трусов М. А., Узлов Д. Ю. Перспективи використання моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2024. вип. 62. С.70-79. https://doi.org/10.26565/2304-6201-2024-62-07

How to quote: M. Trusov, D. Uzlov "Perspectives of using deep learning models for semantic image segmentation on autonomous devices", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 62, pp. 70-79, 2024. [In Ukrainian] https://doi.org/10.26565/2304-6201-2024-62-07

1. Вступ

Комп'ютерний зір є однією з найбільш динамічних і стрімко розвиваючихся галузей штучного інтелекту (ШІ), яка займається створенням технологій, здатних імітувати зорове сприйняття людини. Ця дисципліна об'єднує в собі елементи комп'ютерних наук, оптики, механіки та нейронауки для розробки алгоритмів, які можуть аналізувати, обробляти та інтерпретувати візуальні дані з різноманітних джерел [1]. Серед ключових напрямків цієї галузі особливе місце займає розробка та вдосконалення алгоритмів семантичної сегментації зображень [2]. Семантична сегментація, як інтегральна частина комп'ютерного зору, має за мету детальну класифікацію кожного пікселя на зображенні згідно з визначеними категоріями [3]. Вона відрізняється від процесу детекції об'єктів, який обмежується створенням контурів навколо них, надаючи більш глибоке розуміння форми та положення об'єктів у просторі зображення. Цей метод є одним з

трьох підходів у комплексному процесі сегментації зображень, що сприяє точнішому та ефективнішому розумінню візуальної інформації комп'ютерними системами. Моделі семантичної сегментації спрямовані на створення карти сегментації для вхідного зображення, яка є реконструкцією оригіналу, де кожен піксель кодується певним кольором відповідно до його семантичного класу [4]. Це дозволяє створювати маски сегментації, які виділяють окремі області зображення, відмежовані від інших областей. Так, наприклад, карта сегментації дерева в порожньому полі, ймовірно, міститиме три маски сегментації: для дерева, для землі та для неба на задньому плані, відповідно. Семантична сегментація, яка дозволяє комп'ютерним системам ідентифікувати та класифікувати об'єкти на зображеннях до відповідних категорій, є ключовою для створення інтелектуальних систем, здатних до глибокого розуміння візуального контенту. Підвищення точності та ефективності цих алгоритмів сприяє значному прогресу у вирішенні завдань, пов'язаних з обробкою великих обсягів візуальних даних, забезпечуючи більш точне та швидке узагальнення цільових об'єктів.

Проблематика семантичної сегментації є фокусом наукових досліджень ряду вітчизняних та зарубіжних вчених [5-7]. Зокрема, у вітчизняній науці дослідження зосереджені на адаптації та вдосконаленні сучасних методів семантичної сегментації для конкретних прикладних задач, таких як аналіз супутникових знімків, аналіз даних від БПЛА різних типів тощо. Окрім цього, такі галузі як робототехніка, медицина, автопромисловість в тому чи іншому вигляді використовують новітні технології, пов'язані з необхідністю швидкого та якісного аналізу зображення. У свою чергу, дослідження зарубіжних вчених охоплюють ширший спектр проблем та підходів, включаючи розробку нових архітектур нейронних мереж та методів навчання [8,9].

Загальноприйняті на сьогоднішній день концепції семантичної сегментації постулюють існування двох підходів до реалізації семантичної сегментації [10]. Зокрема, традиційні методи семантичної сегментації включають два основні процеси: виділення ознак (feature extraction) і класифікацію пікселів (pixel classification). Однак, традиційний підхід має кілька недоліків, зокрема високу залежність від доменної експертизи для визначення та виділення ознак, що може бути часозатратним та не завжди ефективним для складних зображень [11]. Крім того, цей підхід часто обмежений у своїй здатності адаптуватися до нових, невидимих даних, що призводить до зниження точності класифікації. У відповідь на ці обмеження, було розроблено методологію глибокого навчання, яка використовує глибокі нейронні мережі для автоматичного виявлення ознак, що значно підвищує точність та універсальність систем семантичної сегментації [1,2]. Завдяки своїй здатності до глибокого навчання на великих наборах даних, ці системи можуть ефективно адаптуватися до нових завдань, забезпечуючи високу точність навіть у складних візуальних умовах. Провідними архітектурами для семантичної сегментації з використанням глибокого навчання є Fully Convolutional Network (FCN), U-Net, DeepLab та PSPNet. Ці архітектури ефективно використовують згорткові нейронні мережі (CNN) для деталізованої екстракції характеристик та високоточної класифікації пікселів у складних візуальних сценах.
Однак, при використанні нейронних мереж для семантичної сегментації зображень, виникає питання про їх інтеграцію в автономні пристрої з обмеженим або відсутнім доступом до Інтернету. У світлі цього, метою даної роботи є дослідження потенціалу застосування моделей глибокого навчання в автономних пристроях, аналіз методів їх ефективної інтеграції та використання без постійного з'єднання з Інтернетом.

2. Використання моделей глибокого навчання в автономних пристроях для семантичної сегментації зображень

Аналіз останніх досліджень у галузі семантичної сегментації показав, що моделі глибокого навчання, зокрема повністю згорткові мережі (FCN) та інші типи нейронних мереж, можуть бути ефективно використані на автономних пристроях, які не мають доступу до Інтернету. Одним із способів реалізації цього є **деплой моделей** безпосередньо **на пристрої** (**on-device interference**), що дозволяє використовувати навчену модель для локального прогнозування без необхідності з'єднання з мережею. Це включає навчання моделі на потужному обладнанні, а потім її розгортання на кінцевому пристрої [12]. Таким чином, пристрій може виконувати локальні прогнози без необхідності постійного підключення до зовнішніх обчислювальних ресурсів. Це не тільки зменшує затримку при обробці даних, але й підвищує приватність, оскільки чутлива інформація не передається через мережу. Крім того, локальне виконання моделей дозволяє

працювати в умовах обмеженого або відсутнього підключення до Інтернету, що розширює можливості застосування ШІ в різних сценаріях.

Інша концепція, що базується на **граничних обчисленнях (edge computing)**, передбачає розміщення обчислювальних ресурсів ближче до джерела даних, що дозволяє обробляти інформацію локально. Наприклад, розумні камери можуть аналізувати відеопотоки безпосередньо на пристрої за допомогою попередньо навчених моделей, що значно зменшує залежність від постійного інтернет-з'єднання та сприяє обробці даних у реальному часі [13,14]. Окрім цього, граничні обчислення також дозволяють розподілити навантаження між різними пристроями, створюючи розподілену мережу систем ШІ, здатних взаємодіяти та обмінюватися інформацією для вирішення складних завдань.

Для ефективного виконання моделей нейронних мереж на автономних пристроях широко використовуються **спеціалізовані вбудовані системи**. Такі платформи, як NVIDIA Jetson, Google Coral та Intel Movidius, розроблені саме для цієї мети і можуть бути інтегровані в різноманітні автоматизовані пристрої для виконання складних завдань, включаючи семантичну сегментацію. Ці системи оптимізовані для виконання операцій, характерних для нейронних мереж, що дозволяє значно прискорити обчислення порівняно зі звичайними процесорами. Крім того, вони часто мають низьке енергоспоживання, що робить їх перспективною альтернативою для мобільних та автономних пристроїв [15].

Для забезпечення ефективної роботи нейронних мереж на пристроях з обмеженими ресурсами застосовується широкий спектр **методів оптимізації**. Ці техніки спрямовані на зменшення обчислювальної складності моделей при збереженні їх точності та продуктивності. Одним з ключових напрямків є стиснення моделей (model compression), яке включає в себе ряд підходів, таких як квантування, скорочення та дистиляція знань. Квантування передбачає зменшення точності представлення ваг та активацій нейронної мережі, що дозволяє суттєво скоротити обсяг пам’яті, необхідний для зберігання моделі, та прискорити обчислення [16]. Скорочення мережі полягає у видаленні надлишкових нейронів та зв'язків, що не мають значного впливу на кінцевий результат, тим самим зменшуючи кількість параметрів моделі. Дистиляція знань дозволяє передати "знання" від великої складної моделі до меншої, зберігаючи при цьому високу точність прогнозування.

Іншим важливим напрямком оптимізації є використання апаратного прискорення (hardware acceleration). Цей підхід передбачає застосування спеціалізованих обчислювальних пристроїв, таких як графічні процесори (GPU), тензорні процесори (TPU) або спеціалізовані мікросхеми для штучного інтелекту (ШІ-чіпи), які оптимізовані для виконання операцій, характерних для нейронних мереж. Графічні процесори, завдяки своїй паралельній архітектурі, здатні значно прискорити матричні обчислення, які є основою більшості операцій в нейронних мережах [17]. Тензорні процесори, розроблені спеціально для задач машинного навчання, забезпечують ще вищу ефективність за рахунок оптимізації під конкретні алгоритми глибокого навчання. Спеціалізовані ШІ-чіпи, такі як нейроморфні процесори, імітують структуру біологічних нейронних мереж, що дозволяє досягти надзвичайно високої енергоефективності при виконанні завдань штучного інтелекту [18].

Комбінація методів стиснення моделей та апаратного прискорення дозволяє досягти синергетичного ефекту, значно підвищуючи ефективність роботи нейронних мереж на вбудованих системах. Наприклад, квантовані моделі можуть бути ще більш ефективно виконані на спеціалізованих апаратних прискорювачах, оптимізованих під операції з низькою точністю. Крім того, розробляються нові архітектури нейронних мереж, які враховують особливості цільового апаратного забезпечення, що дозволяє максимально використовувати його можливості [19]. Такий комплексний підхід до оптимізації відкриває широкі перспективи для впровадження складних алгоритмів штучного інтелекту в мобільні та вбудовані пристрої, розширюючи сферу їх застосування та підвищуючи автономність та інтелектуальність edge-пристроїв.

Нарешті, ще одним аспектом сучасних вбудованих систем, що дозволяє пристроям виконувати складні завдання без постійного підключення до Інтернету, є так звана **обробка даних в автономному режимі (offline data processing)**. Цей підхід особливо корисний у ситуаціях, коли пристрій збирає дані та обробляє їх пакетами, виконуючи завдання сегментації в автономному режимі та лише час від часу підключаючись до Інтернету для оновлення або додаткової передачі даних. Такий метод забезпечує високу надійність, знижує затримку та покращує захист даних, оскільки чутлива інформація не передається через мережу. Однією з ключових переваг автономної

обробки даних є можливість виконання обчислень на місці, що зменшує залежність від зовнішніх обчислювальних ресурсів і підвищує швидкість реакції системи. Наприклад, у промислових застосуваннях, таких як автоматизовані виробничі лінії або системи контролю якості, локальна обробка даних дозволяє швидко виявляти дефекти та приймати рішення в реальному часі, що значно підвищує ефективність виробничих процесів. Крім того, автономні пристрої можуть працювати в умовах обмеженого або відсутнього підключення до Інтернету, що робить їх ідеальними для використання в віддалених або важкодоступних місцях.

Таким чином, можна стверджувати, що навчання моделей в автономному режимі та розгортання оптимізованих версій на периферійних пристроях або вбудованих системах є ефективним підходом для досягнення високоякісної семантичної сегментації без необхідності постійного підключення до Інтернету. Цей підхід має значні переваги, зокрема зниження затримок, підвищення надійності та забезпечення приватності даних, оскільки обробка здійснюється локально.

3. Параметри, які впливають на обсяг пам'яті, що займається натренованою моделлю

Мінімальний необхідний обсяг пам'яті для розгортання навчених моделей на пристрої залежить від кількох факторів, включаючи складність моделі, конкретну архітектуру та застосовані методи оптимізації. Розглянемо ці аспекти детальніше.

Розмір моделі є ключовим фактором. Прості моделі, такі як менші версії FCN або легкі архітектури (наприклад, MobileNet, SqueezeNet), можуть займати від кількох до десятків мегабайт. Натомість більші моделі, зокрема, повнорозмірні FCN або інші складні архітектури (наприклад, ResNet, VGG), можуть вимагати від сотень мегабайт до кількох гігабайт пам’яті [20].

Застосування методів оптимізації дозволяє суттєво зменшити розмір моделі. Квантизація, яка передбачає зниження точності ваги моделі (наприклад, з 32-бітної до 8-бітної), може зменшити розмір моделі до 4 разів. Обрізка (pruning) видаляє менш важливі ваги або нейрони, що також зменшує розмір моделі без значного впливу на продуктивність [21]. Дистиляція знань, при якій менша модель (учень) навчається імітувати більшу модель (вчитель), може призвести до значно меншої моделі з порівнянною продуктивністю.

Розглянемо конкретні приклади моделей. MobileNetV2, компактна модель, розроблена для мобільних пристроїв, зазвичай займає близько 10-15 МБ після квантизації [22]. Tiny YOLO, менша версія моделі ідентифікації об'єктів YOLO (You Only Look Once), займає приблизно 60 МБ. DeepLab, популярна модель для семантичної сегментації, має менші версії (наприклад, з використанням MobileNet як основи) розміром 20-30 МБ, а більші версії (наприклад, з використанням Xception або ResNet) – близько 100-200 МБ [23].

Окрім цього, певну увагу слід приділити додатковим вимогам. Середовище виконання або бібліотеки, необхідні для роботи моделі (наприклад, TensorFlow Lite, ONNX Runtime), також потребують певного обсягу пам'яті. Крім того, допоміжні дані, такі як мітки класів або конфігураційні файли, додають до загальних вимог до пам'яті.

Таким чином, можна оцінити вимоги до пам'яті наступним чином:

- для простих завдань та легких моделей: 10-50 МБ.
- для моделей середньої складності з деякою оптимізацією: 50-200 МБ.
- для складних, високопродуктивних моделей: від 200 МБ до кількох ГБ.

Наприклад, при розгортанні квантованої версії MobileNetV2 для семантичної сегментації можна очікувати наступні вимоги до пам'яті: розмір моделі - близько 10-15 МБ, накладні витрати фреймворку - 5-10 МБ, допоміжні дані - близько 1 МБ, що в сумі складає 16-26 МБ.

Таким чином, завдяки ефективним моделям та оптимізаціям, можливо розгортати нейронні мережі на пристроях з відносно невеликими вимогами до пам’яті, що робить можливим функціонування багатьох повністю автоматизованих пристроїв незалежно від підключення до Інтернету.

4. Особливості імплементації натренованих моделей у власні програмні продукти

Для використання навчених моделей у власних програмах необхідні бібліотеки або фреймворки, що полегшують взаємодію з цими моделями [24]. Розглянемо деякі поширені бібліотеки та фреймворки для різних платформ:

1. *TensorFlow Lite* – фреймворк, розроблений для розгортання моделей на мобільних та вбудованих пристроях. Він підтримує мови програмування Python, C++, Java та Swift. Ключовими

особливостями TensorFlow Lite є конвертація моделей TensorFlow у менший, оптимізований формат, надання API для інференсу на різних платформах (Android, iOS, Linux-based systems), а також підтримка апаратного прискорення (наприклад, через NNAPI, GPU).

2. *ONNX (Open Neural Network Exchange) Runtime* представляє собою кросплатформне високопродуктивне середовище для моделей формату ONNX. Він підтримує Python, C++, C#, Java, JavaScript та інші мови. ONNX Runtime дозволяє працювати з моделями з різних фреймворків (наприклад, PyTorch, TensorFlow), оптимізований для різних апаратних прискорювачів (GPU, TPU) і може використовуватися на різних платформах, включаючи Windows, Linux та macOS.

3. *PyTorch Mobile* призначений для розгортання моделей PyTorch на мобільних пристроях. Він підтримує Python, Java та C++. PyTorch Mobile надає інструменти для оптимізації та конвертації моделей PyTorch для мобільного розгортання, підтримує платформи Android та iOS, а також інтегрується з фреймворками розробки мобільних додатків.

4. *Core ML* є фреймворком машинного навчання Apple для додатків iOS та macOS. Він переважно використовується з Swift та Objective-C. Core ML інтегрується з Xcode для безперешкодного розгортання, оптимізований для апаратного забезпечення Apple (наприклад, Neural Engine) і підтримує конвертацію з різних форматів моделей (TensorFlow, PyTorch).

5. *NVIDIA TensorRT* представляє собою набор інструментів для високопродуктивного інференсу глибокого навчання на GPU NVIDIA. Він підтримує Python та C++. TensorRT оптимізує та розгортає моделі для інференсу на апаратному забезпеченні NVIDIA, підтримує різні фреймворки глибокого навчання (TensorFlow, PyTorch) і ідеально підходить для крайових пристроїв з модулями NVIDIA Jetson.

6. *OpenCV* – відкрита бібліотека комп'ютерного зору та машинного навчання. Вона підтримує C++, Python, Java та MATLAB. OpenCV надає можливості інференсу глибокого навчання через модуль DNN, підтримує моделі з різних фреймворків (Caffe, TensorFlow, PyTorch) і має кросплатформну підтримку (Windows, Linux, macOS, Android, iOS).

Нижче наведено приклад інтеграції з TensorFlow Lite (Python):

```
import tensorflow as tf
import numpy as np

# Load the TFLite model and allocate tensors.
interpreter = tf.lite.Interpreter(model_path="model.tflite")
interpreter.allocate_tensors()

# Get input and output tensors.
input_details = interpreter.get_input_details()
output_details = interpreter.get_output_details()

# Prepare input data
input_data = np.array(your_input_data, dtype=np.float32)

# Run inference
interpreter.set_tensor(input_details[0]['index'], input_data)
interpreter.invoke()

# Get output data
output_data = interpreter.get_tensor(output_details[0]['index'])
print(output_data)
```

Вибір конкретної бібліотеки або фреймворку залежить від специфіки проекту, цільової платформи та вимог до продуктивності.

5. Шифрування моделей глибокого навчання

При дослідженні використання моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях, критичного значення набуває забезпечення безпеки доступу

до навчених моделей. Розглянемо декілька сучасних підходів до забезпечення безпеки, які можуть бути ефективно застосовані у даному контексті:

1. Модулі апаратної безпеки (Hardware Security Modules, HSM)

HSM представляють собою спеціалізовані фізичні пристрої, які виконують криптографічну обробку та забезпечують надійне зберігання ключів [25]. Вони відіграють вирішальну роль у захисті моделей глибокого навчання, призначених для семантичної сегментації, шляхом зберігання та управління криптографічними ключами, що використовуються для шифрування та дешифрування моделей. Інтеграція HSM у автономні пристрої для реалізації семантичної сегментації зображень дозволяє проводити операції шифрування та дешифрування в безпечному середовищі [26]. Це запобігає несанкціонованому доступу до навчених моделей, оскільки вони можуть бути зашифровані та розшифровані лише всередині HSM. Такий метод значно підвищує захист інтелектуальної власності та конфіденційності даних, які використовуються під час навчання моделей.

2. Модуль довіреної платформи (Trusted Platform Module, TPM)

TPM є захищеним криптопроцесором, який може зберігати криптографічні ключі та виконувати криптографічні операції, створюючи надійну апаратну основу захисту [27]. Він є ключовим елементом у захисті моделей глибокого навчання для семантичної сегментації, оскільки забезпечує безпечне зберігання ключів, які є необхідними для доступу до цих моделей. Застосування TPM дозволяє шифрувати моделі та забезпечувати їх розшифрування лише за допомогою авторизованих операцій, що значно підвищує рівень безпеки. Такий підхід гарантує, що навіть при фізичному доступі до пристрою, несанкціоноване використання моделей стає вкрай утрудненим.

3. Комплексний підхід: шифрування та контроль доступу

Ефективний захист моделей глибокого навчання, які використовуються для семантичної сегментації зображень, вимагає комплексного підходу, що об'єднує якісне шифрування та ретельний контроль доступу. Шифрування моделей має бути здійснене за допомогою передових алгоритмів, таких як AES-256, що забезпечує надійний захист даних. Ключі для шифрування необхідно зберігати в безпечному середовищі, використовуючи модулі HSM або TPM, що гарантує їхню цілісність навіть у випадку несанкціонованого вилучення моделей з пристрою [28]. Такий комплексний підхід є ключовим для забезпечення високого рівня захисту моделей глибокого навчання в автономних пристроях.

6. Висновки

У роботі продемонстровано значний потенціал використання моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях. Ця технологія відкриває широкі можливості для розвитку інтелектуальних систем, здатних аналізувати візуальну інформацію без постійного підключення до зовнішніх обчислювальних ресурсів. Ключовим аспектом успішного впровадження таких моделей є оптимізація їх розміру та ефективності. Параметри, що впливають на обсяг пам'яті, займаний натренованою моделлю, включають складність архітектури, кількість шарів та нейронів, а також точність представлення ваг. Застосування методів оптимізації, таких як квантизація, обрізка та дистиляція знань, дозволяє значно зменшити розмір моделей без суттєвої втрати точності, що робить їх придатними для використання на пристроях з обмеженими ресурсами. Імплементація натренованих моделей у власні програмні продукти вимагає використання спеціалізованих бібліотек та фреймворків, таких як TensorFlow Lite, ONNX Runtime або PyTorch Mobile. Ці інструменти забезпечують ефективне розгортання моделей на різних платформах, включаючи мобільні пристрої та вбудовані системи. Важливим аспектом є також оптимізація моделей під конкретне апаратне забезпечення, що дозволяє максимально використовувати доступні обчислювальні ресурси. Критичним також є забезпечення безпеки доступу до навчених моделей глибокого навчання для семантичної сегментації зображень на автономних пристроях. Це вимагає комплексного підходу, що поєднує апаратні та програмні рішення.

Перспективи розвитку цієї галузі включають подальше вдосконалення архітектур нейронних мереж для підвищення ефективності семантичної сегментації, розробку нових методів оптимізації для зменшення обчислювальних вимог, а також створення спеціалізованих апаратних прискорювачів для ефективного виконання операцій глибокого навчання на автономних пристроях.

СПИСОК ЛІТЕРАТУРИ

1. R. Szeliski, Computer vision: algorithms and applications. Springer Nature, 2022. 925 p. https://link.springer.com/book/10.1007/978-3-030-34372-9.
2. Hao S., Zhou Y., Guo Y. A brief survey on semantic segmentation with deep learning. *Neurocomputing*. 2020. V. 406. P. 302-321. https://www.sciencedirect.com/science/article/abs/pii/S0925231220305476.
3. Guo Y., Liu Y., Georgiou T., Lew M. A review of semantic segmentation using deep neural networks. *Int. J. Multimed. Info Retr*. 2018. V. 7. P. 87-93. https://link.springer.com/article/10.1007/s13735-017-0141-z.
4. Yu H., Yang Z., Tan L., Wang Y., Sun W., Sun M., Tang Y. Methods and datasets on semantic segmentation: a review. *Neurocomputing*. 2018. V. 304. P. 82-103. https://www.sciencedirect.com/science/article/abs/pii/S0925231218304077.
5. Куцик А.Я. Аналіз супутникових знімків на основі семантичної сегментації, кваліфікаційна робота, КПІ ім. І. Сикорського, 2020. https://ela.kpi.ua/items/5c1bdf70-a6a6-45ab-8a63-312296420f6c.
6. Рябко А.В. Аналіз та оцінка методів сегментації супутникових зображень, Всеукр. Науково-технічна конференція «Сталий розвиток систем зв'яязку, навігації, спостереження та організації повітряного руху CNS/ATM - 2023», 29-30 листопада 2023 р., с. 4. https://it-visnyk.kpi.ua/?page_id=2165.
7. Глубока Ю.О. Дослідження якості методів сегментації зображення людини в умовах дії адитивних завад, кваліфікаційна робота, Харківський національний університет радіоелектроніки, 2022. https://openarchive.nure.ua/entities/publication/6bad8449-2d47-4c70-87dc-927980da23e7.
8. Hua Y., Marcos D., Mou L., Zhu X., Tuia D. Semantic segmentation of remote sensing images with sparse annotations. *IEEE Geoscience and Remote Sensing Letters*. 2022. V. 19. P. 1-5. https://arxiv.org/abs/2101.03492.
9. Zhang Y., Chi M. Mask-R-FCN: a deep fusion network for semantic segmentation. *IEEE Access*. 2020. V. 8. P. 155753-155765. https://ieeexplore.ieee.org/document/9151932.
10. O'Mahony N., Campbell S., Carvalho A., et al. Deep learning vs. traditional computer vision, *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC)*, Vol. 11, Springer International Publishing, 2020. P. 128-144. https://arxiv.org/abs/1910.13796.
11. Panella F., Lipani A., Boehm J. Semantic segmentation of cracks: data challenges and architecture. *Automation in Construction*. 2022. V. 135. P. 104110. https://www.sciencedirect.com/science/article/abs/pii/S0926580521005616.
12. Mairittha N., Mairittha T., Inoue S. On-device deep learning inference for efficient activity data collection. *Sensors (Basel)*. 2019. V. 19. P. 3434. https://www.mdpi.com/1424-8220/19/15/3434.
13. Cui T. Review of deep learning and mobile edge computing in autonomous driving. *Вісник Львівської політехніки*. 2022. V. 12. P. 208-218. https://science.lpnu.ua/sites/default/files/journal-paper/2023/jan/29757/221029maket-210-220.pdf.
14. Grigorescu S., Trasnea B., Cocias T., Macesanu G. A survey of deep learning techniques for autonomous driving. *J. Field Robotics*. 2020. V. 37. P. 362-386. https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21918.
15. Zhang Z., Li J. A review of artificial intelligence in embedded systems. *Micromachines*. 2023. V. 14. P. 897. https://www.mdpi.com/2072-666X/14/5/897.
16. Merone M., Graziosi A., Lapadula V., Petrosino L., d'Angelis O., Vollero L. A practical approach to the analysis and optimization of neural networks on embedded systems. Sensors. 2022. V. 22. P. 7807. https://www.mdpi.com/1424-8220/22/20/7807.
17. Helms D., Amende K., Bukhari S., et al., Optimizing neural networks for embedded hardware. *SMACD/PRIME 2021, International Conference on SMACD and 16th Conference on PRIME*, online. 2021. P. 1-6. https://ieeexplore.ieee.org/document/9547911.
18. Song W. Hardware accelerator systems for embedded systems. *Advances in Computers*, vol. 122. Elsevier, 2021. P. 23-49. https://www.sciencedirect.com/science/article/abs/pii/S0065245820300917.
19. Yesuf M., Assefa B. Model compression techniques in deep neural networks. *Pan African Conference on Artificial Intelligence*. Cham: Springer Nature Switzerland. 2022. P. 169-190. https://link.springer.com/chapter/10.1007/978-3-031-31327-1_10.

20. Lohn A., Scaling AI. *Technical report, Center for Security and Emerging Technology*, 2023. https://cset.georgetown.edu/publication/scaling-ai/.
21. Acun B., Murphy M., Wang X., et al. Understanding training efficiency of deep learning recommendation models at scale. *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE. 2021. https://arxiv.org/abs/2011.05497.
22. Dong K., Zhou C., Rian Y., Li Y. MobileNetV2 model for image classification. *2nd International Conference on Information Technology and Computer Application (ITCA)*. IEEE, 2020. P. 476-480. https://ieeexplore.ieee.org/document/9422058.
23. Berthelier A., Chateau T., Duffner S., et al. Deep model compression and architecture optimization for embedded systems: a survey. *J. Signal Processing Systems*. 2021. V. 93. P. 863-878. https://link.springer.com/article/10.1007/s11265-020-01596-1.
24. Hadidi R., Cao J., Xie Y., et al. Characterizing the deployment of deep neural networks on commercial edge devices. *IEEE International Symposium on Workload Characterization (IISWC)*, IEEE. 2019. P. 35-48. https://ieeexplore.ieee.org/document/9041955.
25. Mavrovouniotis S. Ganley M. *Hardware security modules. Secure Smart Embedded Devices, Platforms and Applications*. New York, NY: Springer New York, 2013. P. 383-405.
26. Vembu S. K., Chattopadhyay A., Saha S. Authenticating edge neural network through hardware security modules and quantum-safe key management. *2024 37th International Conference on VLSI Design and 2024 23rd International Conference on Embedded Systems (VLSID)*, Kolkata, India. 2024. P. 318-323. https://ieeexplore.ieee.org/document/10483401.
27. Ezirim K., Khoo W., Koumantaris G., et al. Trusted platform module – a survey. *The Graduate Center of The City University of New York*, 11. 2012. https://www.researchgate.net/profile/Kenneth-Ezirim/publication/287984174_Trusted_Platform_Module_-_A_Survey/links/567af54608ae197583812a7c/Trusted-Platform-Module-A-Survey.pdf.
28. Köylü T.Ç., Wedig Reinbrecht C.R., Gebregiorgis A., et al. A survey on machine learning in hardware security. *ACM Journal on Emerging Technologies in Computing Systems*. 2023. V. 19. P. 1-37. https://dl.acm.org/doi/10.1145/3589506.

REFERENCES

1. R. Szeliski, Computer vision: algorithms and applications. Springer Nature, 2022. 925 p. https://link.springer.com/book/10.1007/978-3-030-34372-9.
2. Hao S., Zhou Y., Guo Y. A brief survey on semantic segmentation with deep learning. *Neurocomputing*. 2020. V. 406. P. 302-321. https://www.sciencedirect.com/science/article/abs/pii/S0925231220305476.
3. Guo Y., Liu Y., Georgiou T., Lew M. A review of semantic segmentation using deep neural networks. *Int. J. Multimed. Info Retr*. 2018. V. 7. P. 87-93. https://link.springer.com/article/10.1007/s13735-017-0141-z.
4. Yu H., Yang Z., Tan L., Wang Y., Sun W., Sun M., Tang Y. Methods and datasets on semantic segmentation: a review. *Neurocomputing*. 2018. V. 304. P. 82-103. https://www.sciencedirect.com/science/article/abs/pii/S0925231218304077.
5. Kutsik A. Ya. Analysis of satellite imagery based on semantic segmentation, bachelor thesis. Igor Sikorsky KPI, 2020. https://ela.kpi.ua/items/5c1bdf70-a6a6-45ab-8a63-312296420f6c. [in Ukrainian]
6. Ryabko A.V. Analysis and evaluation of satellite image segmentation methods, Ukrainian Scientific and Technical Conference 'Sustainable Development of Communication, Navigation, Surveillance Systems, and Air Traffic Organization CNS/ATM - 2023», November 29-30. P. 4. https://it-visnyk.kpi.ua/?page_id=2165. [in Ukrainian]
7. Glyboka Yu.O. Investigation of the quality of human image segmentation methods under the influence of additive noise, master thesis, Kharkiv National University of Radio Electronics, 2022. https://openarchive.nure.ua/entities/publication/6bad8449-2d47-4c70-87dc-927980da23e7. [in Ukrainian]
8. Hua Y., Marcos D., Mou L., Zhu X., Tuia D. Semantic segmentation of remote sensing images with sparse annotations. *IEEE Geoscience and Remote Sensing Letters*. 2022. V. 19. P. 1-5. https://arxiv.org/abs/2101.03492.
9. Zhang Y., Chi M. Mask-R-FCN: a deep fusion network for semantic segmentation. *IEEE Access*. 2020. V. 8. P. 155753-155765. https://ieeexplore.ieee.org/document/9151932.

10. O'Mahony N., Campbell S., Carvalho A., et al. Deep learning vs. traditional computer vision, *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC)*, Vol. 11, Springer International Publishing, 2020. P. 128-144. https://arxiv.org/abs/1910.13796.
11. Panella F., Lipani A., Boehm J. Semantic segmentation of cracks: data challenges and architecture. *Automation in Construction*. 2022. V. 135. P. 104110. https://www.sciencedirect.com/science/article/abs/pii/S0926580521005616.
12. Mairittha N., Mairittha T., Inoue S. On-device deep learning inference for efficient activity data collection. *Sensors (Basel)*. 2019. V. 19. P. 3434. https://www.mdpi.com/1424-8220/19/15/3434.
13. Cui T. Review of deep learning and mobile edge computing in autonomous driving. *Вісник Львівської політехніки*. 2022. V. 12. P. 208-218. https://science.lpnu.ua/sites/default/files/journal-paper/2023/jan/29757/221029maket-210-220.pdf.
14. Grigorescu S., Trasnea B., Cocias T., Macesanu G. A survey of deep learning techniques for autonomous driving. *J. Field Robotics*. 2020. V. 37. P. 362-386. https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21918.
15. Zhang Z., Li J. A review of artificial intelligence in embedded systems. *Micromachines*. 2023. V. 14. P. 897. https://www.mdpi.com/2072-666X/14/5/897.
16. Merone M., Graziosi A., Lapadula V., Petrosino L., d'Angelis O., Vollero L. A practical approach to the analysis and optimization of neural networks on embedded systems. Sensors. 2022. V. 22. P. 7807. https://www.mdpi.com/1424-8220/22/20/7807.
17. Helms D., Amende K., Bukhari S., et al., Optimizing neural networks for embedded hardware. *SMACD/PRIME 2021, International Conference on SMACD and 16th Conference on PRIME*, online. 2021. P. 1-6. https://ieeexplore.ieee.org/document/9547911.
18. Song W. Hardware accelerator systems for embedded systems. *Advances in Computers*, vol. 122. Elsevier, 2021. P. 23-49. https://www.sciencedirect.com/science/article/abs/pii/S0065245820300917.
19. Yesuf M., Assefa B. Model compression techniques in deep neural networks. *Pan African Conference on Artificial Intelligence*. Cham: Springer Nature Switzerland. 2022. P. 169-190. https://link.springer.com/chapter/10.1007/978-3-031-31327-1_10.
20. Lohn A., Scaling AI. *Technical report, Center for Security and Emerging Technology*, 2023. https://cset.georgetown.edu/publication/scaling-ai/.
21. Acun B., Murphy M., Wang X., et al. Understanding training efficiency of deep learning recommendation models at scale. *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE. 2021. https://arxiv.org/abs/2011.05497.
22. Dong K., Zhou C., Rian Y., Li Y. MobileNetV2 model for image classification. *2nd International Conference on Information Technology and Computer Application (ITCA)*. IEEE, 2020. P. 476-480. https://ieeexplore.ieee.org/document/9422058.
23. Berthelier A., Chateau T., Duffner S., et al. Deep model compression and architecture optimization for embedded systems: a survey. *J. Signal Processing Systems*. 2021. V. 93. P. 863-878. https://link.springer.com/article/10.1007/s11265-020-01596-1.
24. Hadidi R., Cao J., Xie Y., et al. Characterizing the deployment of deep neural networks on commercial edge devices. *IEEE International Symposium on Workload Characterization (IISWC)*, IEEE. 2019. P. 35-48. https://ieeexplore.ieee.org/document/9041955.
25. Mavrovouniotis S. Ganley M. *Hardware security modules. Secure Smart Embedded Devices, Platforms and Applications*. New York, NY: Springer New York, 2013. P. 383-405.
26. Vembu S. K., Chattopadhyay A., Saha S. Authenticating edge neural network through hardware security modules and quantum-safe key management. *2024 37th International Conference on VLSI Design and 2024 23rd International Conference on Embedded Systems (VLSID)*, Kolkata, India. 2024. P. 318-323. https://ieeexplore.ieee.org/document/10483401.
27. Ezirim K., Khoo W., Koumantaris G., et al. Trusted platform module – a survey. *The Graduate Center of The City University of New York*, 11. 2012. https://www.researchgate.net/profile/Kenneth-Ezirim/publication/287984174_Trusted_Platform_Module_-_A_Survey/links/567af54608ae197583812a7c/Trusted-Platform-Module-A-Survey.pdf.
28. Köylü T.Ç., Wedig Reinbrecht C.R., Gebregiorgis A., et al. A survey on machine learning in hardware security. *ACM Journal on Emerging Technologies in Computing Systems*. 2023. V. 19. P. 1-37. https://dl.acm.org/doi/10.1145/3589506.

Trusov Mykhaylo *Chief software developer, Effective Programming for America, Ukraine, 23 Serpnya Str., 33, Kharkiv, Ukraine, 61045*

Uzlov Dmitro *Associate Professor of the Department of Theoretical and Applied Informatics, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine;*

Perspectives of using deep learning models for semantic image segmentation on autonomous devices

Relevance. The implementation of deep learning models for semantic segmentation on autonomous devices is a promising direction for the development of intelligent systems capable of analyzing visual information without constant connection to external resources. This enables the creation of more autonomous and efficient systems that can operate in real-time and under resource constraints. Such an approach is highly significant for various industries, including robotics, autonomous vehicles, medical diagnostics, and other fields where high accuracy and speed of image processing are required.
Goal. The goal of this work is to explore the possibilities and challenges of using deep learning models for semantic segmentation on autonomous devices. This includes analyzing the efficiency of the models, their adaptation to the limited resources of the devices, and developing methods to ensure the security of access to the trained models.
Research methods. The research methods include theoretical analysis, systematization, and generalization of the use of deep learning models in autonomous devices. Special attention is given to the parameters affecting the memory footprint of the models and the specifics of implementing trained models into proprietary software products. Additionally, modern approaches to encrypting models to ensure their security have been considered.
Results. A comparative analysis of traditional models and deep learning models for semantic segmentation of images has been conducted. Significant potential of deep learning technology for creating autonomous intelligent systems is identified. Various deep learning models currently used for semantic segmentation of images have been reviewed. The impact of key parameters on the efficiency of models on devices with limited resources has been determined, and the role of model size has been considered. Recommendations for implementing trained models into software products are presented, including optimizing models to reduce the size and increase the speed. Special attention has been paid to the analysis of encrypting trained models. It is shown that ensuring the security of access to trained deep learning models for semantic segmentation of images on autonomous devices requires a comprehensive approach that combines hardware and software solutions.
Conclusions. Further developments in the field of deep learning for semantic segmentation on autonomous devices will contribute to the development of more efficient and autonomous systems for a wide range of applications, including computer vision, robotics, and more. Ensuring the security of access to trained deep learning models for semantic segmentation of images on autonomous devices requires a comprehensive approach that combines hardware and software solutions. This not only protects the intellectual property of developers but also ensures the integrity and confidentiality of the data processed by autonomous devices when performing semantic segmentation tasks.

Keywords: *deep learning, semantic segmentation, autonomous devices, model optimization, embedded systems, hardware acceleration, data encryption*

Надійшла у першій редакції 01.05.2024, в останній - 10.06.2024.
The first version has been received on 01.05.2024, the final version – on 10.06.2024.

Наукове видання

Вісник Харківського національного університету імені В. Н. Каразіна

Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»

Випуск 62

Збірник наукових праць

Українською та англійською мовами

Комп'ютерне верстання О.О. Афанасьєва

Підписано до друку 21.06.2024 р.
Формат 60x84/8. Папір офсетний. Друк цифровий.
Ум. друк. арк. – 7,83.
Обл.– вид. арк. – 9,79.
Наклад 50 пр. Зам. № 56/2024
Безкоштовно

Видавець і виготовлювач
Харківський національний університет імені В. Н. Каразіна
61022, м. Харків, майдан Свободи, 4
Свідоцтво суб'єкта видавничої справи ДК №3367 від 13.01.09

Видавництво Харківський національний університет імені В. Н. Каразіна
тел.: 705-24-32