CIT 2024

WORKSHOP

ШКОЛА-СЕМІНАР

молодих вчених і студентів

КОМП'ЮТЕРНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

2 травня
2024 року

м. Тернопіль
вул. О. Теліги, 8

fcit.wunu.edu.ua

ОРГАНІЗАТОРИ

- Західноукраїнський національний університет
- Факультет комп'ютерних інформаційних технологій
- Асоціація фахівців комп'ютерних інформаційних технологій

Західноукраїнський національний університет
Факультет комп’ютерних інформаційних технологій
Асоціація фахівців комп’ютерних інформаційних технологій

МАТЕРІАЛИ ШКОЛИ-СЕМІНАРУ
МОЛОДИХ ВЧЕНИХ І СТУДЕНТІВ

КОМП'ЮТЕРНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

COMPUTER INFORMATION TECHNOLOGIES

2 травня 2024 року

CIT’2024

Тернопіль
ЗУНУ
2024

ББК 32.97
УДК 004.2-3+004.9+51.7+519.6-8

Організатори школи-семінару:

Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Асоціація фахівців комп'ютерних інформаційних технологій

32.97 Комп'ютерні інформаційні технології: Матеріали школи-семінару молодих вчених і студентів СІТ'2024. – Тернопіль: ЗУНУ, 2024.

У матеріалах семінару опубліковані результати наукових досліджень і розробок науковців та студентів факультету комп'ютерних інформаційних технологій ЗУНУ з таких напрямків: математичні моделі об'єктів та процесів, комп'ютерні мережеві технології; спеціалізовані комп'ютерні системи; системи штучного інтелекту; інженерія програмного забезпечення; комп'ютерні технології інформаційної безпеки та управління проектами. Матеріали призначені для наукових співробітників, викладачів, інженерно-технічних працівників, аспірантів та студентів.

Відповідальний за випуск:
Пукас А.В., д. т. н., професор, завідувач кафедри комп'ютерних наук

Відповідальність за достовірність, стиль викладення та зміст надрукованих матеріалів несуть автори.

ЗМІСТ

УДК 004.62

ІНФОРМАЦІЙНО-АНАЛІТИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ЕФЕКТИВНОГО МОНІТОРИНГУ ФІНАНСОВОЇ ДІЯЛЬНОСТІ ТА ПЛАНУВАННЯ РОЗВИТКУ УНІВЕРСИТЕТУ

Шпінталь М.Я.[1)], Крисак Д.М.[2)]
Західноукраїнський національний університет
1) к.т.н., доцент; 2) магістрант

I. Постановка проблеми

Ефективне управління фінансовими ресурсами є критично важливим для стабільного розвитку та процвітання будь-якої освітньої установи, зокрема університетів. У контексті зростаючої конкуренції та змін у фінансуванні освіти, університетам необхідно забезпечити ефективний моніторинг і планування своєї фінансової діяльності. Це включає аналіз доходів, витрат, інвестиційних проектів та інших фінансових показників, що впливають на стратегічне рішення та оперативне управління. Проте, багато університетів стикаються з проблемами у цьому процесі через застарілі системи управління даними, відсутність інтеграції між різними джерелами інформації та обмежені аналітичні можливості. Така ситуація може призводити до неефективного використання ресурсів, помилкового планування, а також упущених можливостей для розвитку та інновацій.

II. Мета роботи

Основною метою цієї роботи є розробка та впровадження інформаційно-аналітичного забезпечення, яке дозволить університетам ефективно моніторити та планувати свою фінансову діяльність, що сприятиме стратегічному розвитку та підвищенню загальної ефективності управління. Ця система має на меті адресувати ключові виклики, ідентифіковані в постановці проблеми.

III. Основна частина

Дослідження поточного стану інформаційно-аналітичних систем університетів виявило ключові недоліки: застаріле програмне забезпечення, обмежена інтеграція даних та відсутність гнучкості для адаптації до змінюваних умов. Ці недоліки перешкоджають ефективному моніторингу та стратегічному плануванню, що підкреслює потребу в розробці вдосконаленої системи. Розроблена концепція передбачає інтегровану платформу з можливостями глибокого аналізу даних, прогнозування фінансових показників та підтримки стратегічного рішення.

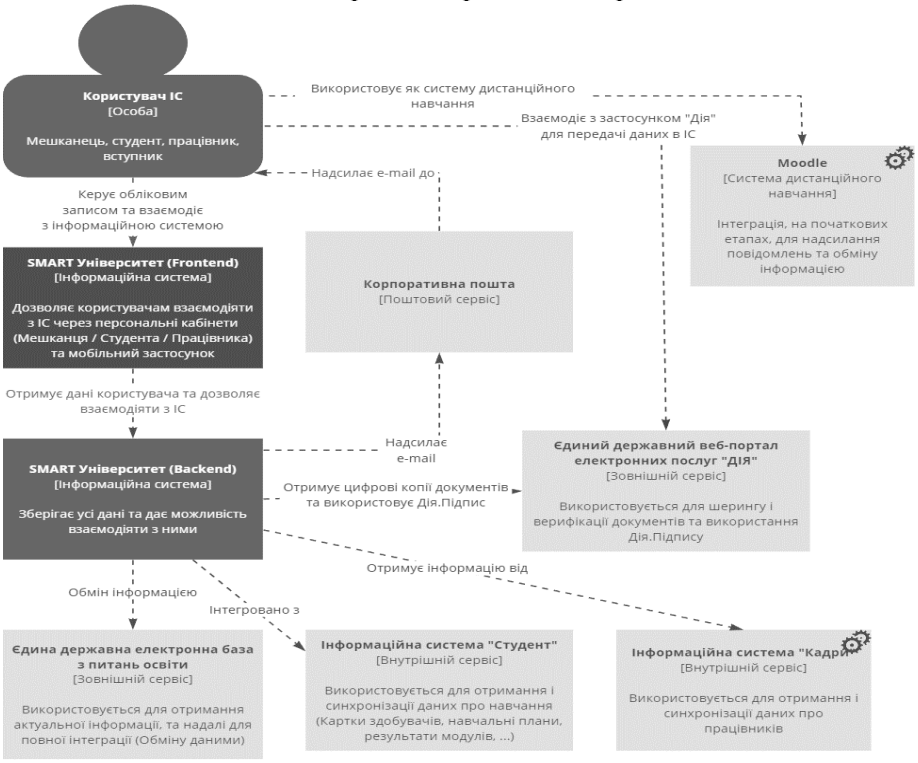


Рисунок 1 – Контекстна діаграма інформаційної системи "SMART Університет"

Система об'єднує дані з різних джерел, включаючи бухгалтерський облік, студентські внески та дослідницьке фінансування, забезпечуючи єдину правдиву картину фінансового стану університету. Побудова на модульній архітектурі, що дозволяє легко інтегрувати нові функції та джерела даних. Використання хмарних технологій забезпечує масштабованість та доступність системи з будь-якого місця та на будь-якому пристрої.

План впровадження включає пілотні проекти в окремих факультетах, з подальшим розгортанням по всьому університету. Важливим аспектом є тренінги та семінари для співробітників, щоб забезпечити високий рівень використання та ефективності системи.

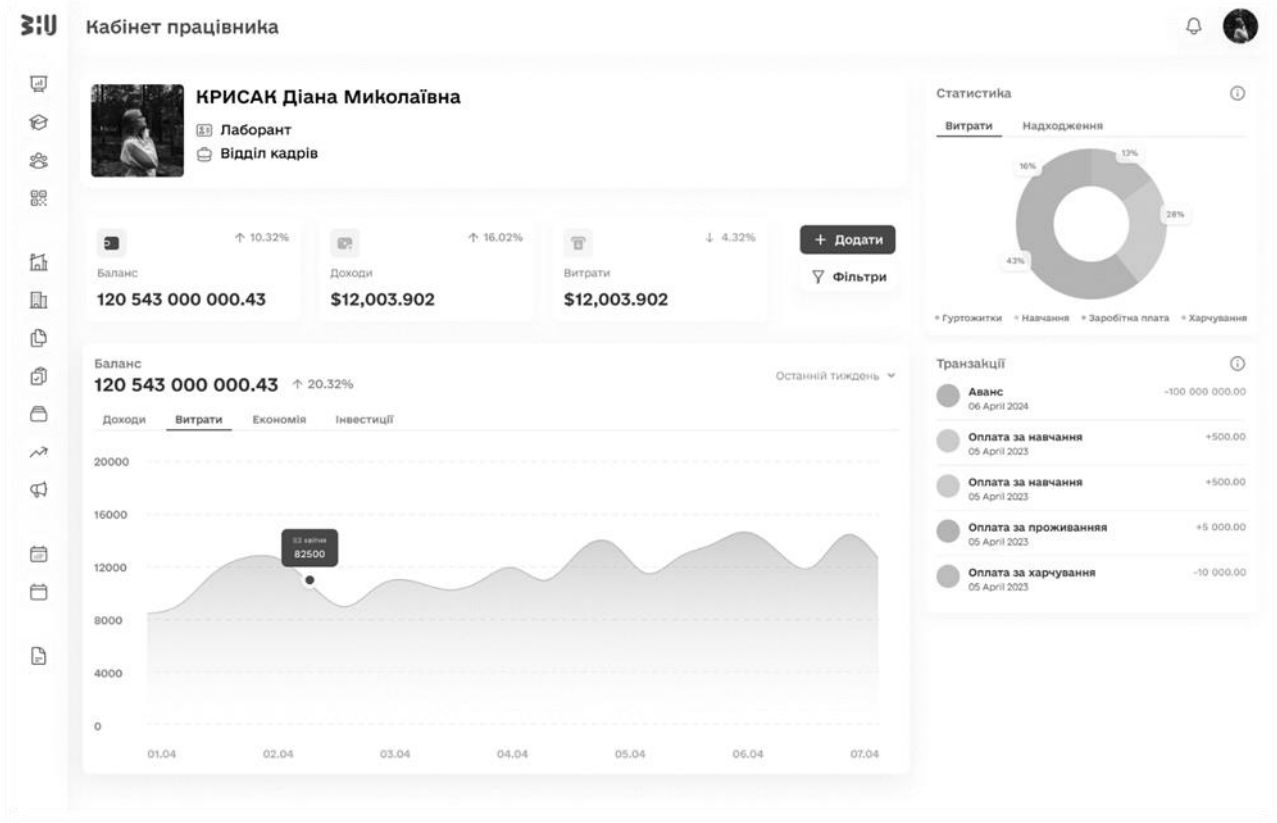


Рисунок 2 – Аналітичний дешборд працівника університету.

Впровадження системи має забезпечити значне підвищення ефективності фінансового управління університету, зокрема через оптимізацію витрат, підвищення прозорості фінансових операцій та покращення прийняття рішень на основі даних. За допомогою цієї системи університети зможуть прогнозувати майбутні фінансові потреби, ефективно керувати ресурсами та адаптуватися до змінюваних умов ринку.

Висновок

Розроблена концепція системи включає інтеграцію даних, передові аналітичні інструменти, підтримку прийняття рішень та можливості для гнучкого прогнозування і планування. Це не лише сприятиме оптимізації фінансового управління, але й надасть університетам інструменти для адаптації до швидко змінюваних умов, підвищуючи їхню конкурентоспроможність і стійкість.

Ефективна інформаційно-аналітична система може стати вирішальним фактором у забезпеченні стабільного розвитку та процвітання університетів. Вона не лише підвищує прозорість та контроль над фінансовими процесами, але й сприяє стратегічному плануванню та інноваціям у навчальному процесі та наукових дослідженнях.

У результаті, розробка та впровадження такої системи є кроком вперед у забезпеченні ефективного та прозорого фінансового управління в університетах, що відповідає сучасним викликам та очікуванням у сфері вищої освіти.

Список використаних джерел

1. Бєлялов Т. Е. Стратегія розвитку як ключовий фактор підвищення ефективності діяльності університету. *Journal of strategic economic research*. 2021. №2. С.8–17.

2. Medzhybovs'ka N. S. Formation of information support system to improve the efficiency ofenterprise management. *Ekonomika: realiichasu*, 2013.Vol. 4, pp. 26-30.

УДК 004.4

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЧНОГО ГЕНЕРУВАННЯ ТЕСТОВИХ ДАНИХ ДЛЯ ДИНАМІЧНО ТИПІЗОВАНИХ МОВ ПРОГРАМУВАННЯ

Шерстюк Ю.Ю.[1)], Порплиця Н.П.[2)]
Західноукраїнський національний університет
[1)] магістрант; [2)]к.т.н., доцент

I. Постановка проблеми

Розробка програмних продуктів є конкурентною діяльністю, а час, доступний для виведення продукту на ринок, часто обмежений. Крім того, складність та розмір програмних систем зростають у останні роки. Щоб уникнути невдачі на ринку, важливо також досягти високої якості. Разом ці тенденції ставлять великі вимоги до розробників ПЗ: вони повинні розробляти більші системи та швидше. Це особливо викликає складнощі заходів спрямованих на підвищення якості, наприклад, тестування програмного забезпечення [1-3].

Пошукове тестування програмного забезпечення (Search-basedsoftwaretesting- SBST) може зменшити час і зусилля, автоматично генеруючи відповідні та достатні тестові випадки. SBST було успішно застосовано в структурному тестуванні. Хоча попередні дослідження показали застосовність SBST для генерації тестових даних для структурного тестування, вони часто обмежувалися простими типами вхідних даних, такими як числові значення [4-6]. Числа є дуже поширеними вхідними даними, але є багато інших типів вхідних даних, які часто використовуються, особливо в об'єктно-орієнтованих програмах. Часто параметри є об'єктами, які самі зберігають внутрішній стан, або складні та складені структури даних, які вимагають належної ініціалізації даних. У таких ситуаціях генерація тестових даних для досягнення певної мети стає набагато складнішою, ніж для простих числових значень [7].

Іншим аспектом є генерація тестових даних для динамічних мов програмування, які стали популярними в останні роки. Динамічна мова - це мова програмування високого рівня, яка дозволяє програмі змінювати свою поведінку динамічно під час виконання. Часто вони не обмежуються на типі об'єктів, які передаються в якості аргументів або створюються під час викликів методів. Таким чином, можна змінювати частини програми, поки система все ще працює, і легко адаптувати програмні системи [8, 9]. Крім того, динамічні мови часто розглядаються як дуже гнучкі та продуктивні. У попередніх дослідженнях були розроблені інструменти SBST, переважно для статично типізованих мов, таких як C/C++ та Java. На нашу думку, SBST ще мало застосовувався до динамічних мов програмування, і актуальною є задача, як SBST може бути адаптований для цього [9, 10].

II. Мета роботи

У статті запропоновано RTG (Ruby Testcase Generator), інструмент, написаний мовою Ruby, який може створювати тестові випадки для вихідного коду Ruby. Метою роботи, відповідно, є підвищення ефективності автоматичного генерування тестових даних для досягнення повного покриття операторів. У статті приділено увагу двом аспектам:

- застосування SBST для динамічних мов програмування для генерації тестових даних;
- засоби генерувати різноманітних тестових вхідних даних, таких як об'єкти та складні структури даних.

III. Проектування програмної системи

У цьому розділі представлено інструмент RTG, автоматичний генератор тестових даних для Ruby. На рисунку 1 показано основну структуру та основні компоненти інструменту. Робота системи починається з отримання вихідного коду (Sourcecode) та обраного класу для тестування (CUT). Маючи ці передумови, аналізатор (Analyser) динамічно завантажує CUT та шукає корисну інформацію, необхідну для генератора тестових даних (Testcase generator). Генератор тестових даних формує набір тестових індивідів, формуючи послідовність створення об'єктів та викликів методів. Вхідні дані для параметрів генеруються за допомогою генератора даних(Data input generator). Після повної ініціалізації окремих тестових даних вони виконуються на програмі, яку тестують (Test caseexecutor). З цього генератор тестових даних отримує зворотний зв'язок, який впливає на подальші кроки пошуку. Після досягнення цілі пошуку інструмент постачає набір сценаріїв тестових даних (Testscenario).

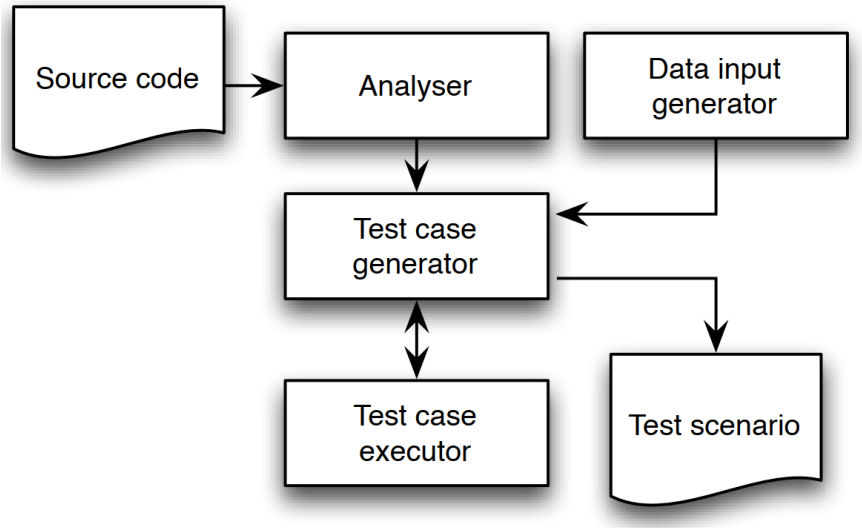


Рисунок 1 – Структура автоматичного генератора тестових даних

Аналізатор. Аналізатор видобуває інформацію, яка використовується пізніше в процесі генерації та адаптації тестових даних. Оскільки Ruby є рефлексивною та динамічною мовою, аналізатор зосереджується на виконанні своєї задачі під час виконання та виконує лише мінімальний аналіз статичного коду. Це означає, що CUT динамічно завантажується в систему та досліджується. Аналізатор надає об'єкт CUTInfo, який містить інформацію про конструктор та його аргументи. Крім того, він зберігає список методів, які оголошені в межах цього класу-цілі. Кожен метод стає методом під тест (MUT), і, таким чином, пов'язується з об'єктом MUTInfo. Цей об'єкт, з свого боку, містить інформацію про MUT, таку як його список аргументів та методи, викликані для кожного окремого аргумента. Крім того, він відстежує покриття, досягнуте під час процесу пошуку, крім адекватних або відкинутих генераторів даних.

Генератор даних. Генератор даних створює вхідні значення, які передаються як аргументи для викликів методів. Знаходження відповідних даних є дуже важливою, але важкою задачею. Існує великий набір можливих типів вхідних даних, і область значень вхідних даних для певного типу може бути досить великою. Оскільки одному генератору даних важко охопити всі різні можливості, основне рішення проекту полягає у наявності різних генераторів для конкретних проблем. Це надає користувачу можливість визначати нові генератори, які можуть створювати дані, що відповідають контексту. Таким чином, набір генераторів даних повинен бути змінюваним, додаванням або вилученням визначених користувачем генераторів. Однак це потребує загального спільного інтерфейсу, такого, що програма може самостійно працювати, не змінюючи свою поведінку для кожного генератора. Мати специфічні для проблеми генератори даних поліпшує також пошук відповідних значень тесту. Припустимо, що у нас є метод, який приймає рядок як вхід і перевіряє, чи є він дійсним ISBN-кодом чи ні. Якщо ми застосуємо простий генератор рядків, який випадковим чином створює будь-яку послідовність символів, то буде дуже важко, якщо не неможливо, знайти дійсний рядок ISBN. З іншого боку, якщо ми визначимо генератор рядків, який виробляє лише значення згідно з попередньо визначеним шаблоном, то шанс успіху у знаходженні дійсного вхідного значення збільшиться.

Виконавець тестових даних. Згенеровані тестові випадки виконуються для отримання інформації, такої як покриття коду. Виконавець тестових даних відстежує покриття, досягнуте попередніми тестовими сценаріями. Таким чином, можливо визначити, чи сприяє поточний тестовий випадок покриттю коду, і, отже, чи він стане частиною кінцевого набору тестових сценаріїв. Тестові випадки поділяються на три основні частини, а саме конструктор, послідовність викликів методів для зміни стану об'єкта та виклик поточного методу під тест. У випадку, якщо виконання тестового сценарію призводить до виникнення винятку, можна визначити відповідну частину. Це робиться для того, щоб уникнути помилкової оцінки в процесі пошуку.

Генератор тестових даних. Генератор тестових даних є ядром RTG і відповідає за створення тестових сценаріїв. Є дві основні задачі: знаходження відповідних вхідних значень та формування розумної послідовності викликів методів. Обидва ці завдання не можуть бути виконані одразу. Фактично, для пошуку можливих тестових даних використовується генетичний алгоритм (ГА). Таким

чином, під час процесу пошуку зберігається популяція тестових особин, яка еволюціонує в напрямку пошуку 'хороших' тестових даних.

Особина сама по собі не може бути виконана, але містить всю необхідну інформацію для створення повного тестового випадку. Цю інформацію можна розділити на три різні категорії, а саме конструктор, послідовність викликів методів та виклик методу під тест. Представлення особини можна побачити на рисунку 2.

Алгоритм пошуку починається з випадково ініціалізованої популяції особин. Кожна особина трансформується в виконуваний тестовий випадок та оцінюється. Оцінка базується на наступній функції придатності:

$$f_{fitness} = (\text{cov} \cdot p) + \frac{executed_{cs}}{total_{cs}} \cdot (1 - p) \quad (1)$$

де p – значення між 0 і 1, cov –показник покриття коду, досягнутого тестовим випадком, $executed_{cs}$ – кількість виконаних структур керування, $total_{cs}$ –загальна кількість існуючих структур керування. Отже, значення придатності є значенням між 0 і 1, де значення, близьке до 1, вказує на кращих особин, ніж значення, близьке до 0.

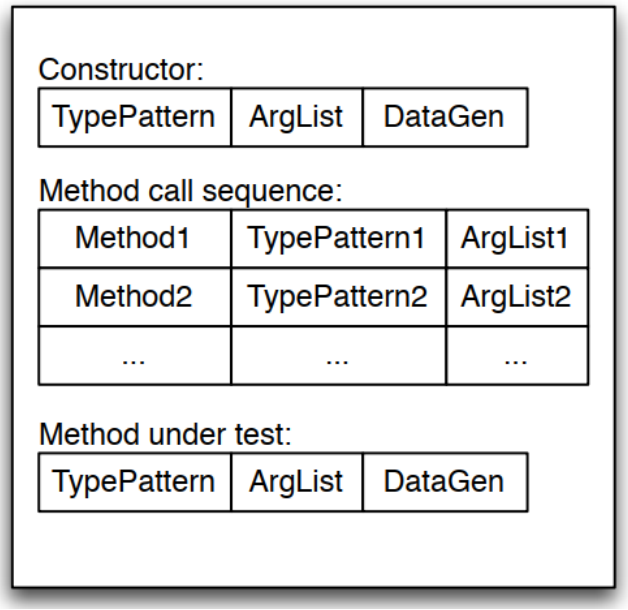


Рисунок 2 – Представлення особин

Особини вибираються з популяції, причому більш пристосовані особини мають більшу ймовірність вибору, ніж інші. Потім вони комбінуються і в кінцевому підсумку мутуються, щоб сформувати нове покоління популяції. Ці операції застосовуються по-різному на кожну частину особини.

Комбінація двох особин для конструктора та методу під тест стосується списку аргументів. У цьому випадку інформація про список аргументів обмінюється на випадково обраній позиції. З іншого боку, комбінація послідовності викликів методів відбувається шляхом вибору випадково двох позицій для кожної особи, на яких замінюється послідовність. Аналогічно випадок з мутацією, яка застосовується з попередньо визначеною ймовірністю до випадково обраних особин. Для конструктора та методу під тест існують дві можливості мутації, а саме генерація нового типового шаблону або виробництво нового вхідного значення для одного з існуючих аргументів. Генерація нового типового шаблону застосовується для охоплення комбінацій типів, які інакше не були б протестовані. Окрема таблиця ведеться для відстеження вже виконаних комбінацій типів. У разі, якщо всі можливі типові шаблони були протестовані принаймні один раз, тоді мутація стосується лише значення аргументу. Для мутації послідовності викликів методів випадково вибирається позиція, на якій метод додається або вилучається з поточної послідовності.

Оскільки Ruby підтримує поліморфізм, неможливо визначити з сигнатури методу, які типи параметрів можуть бути застосовані. Крім того, через комбінацію та мутацію особин може виникнути

нові комбінації типів. Така можлива ситуація показує наступний приклад. Припустимо, що у нас є метод, який приймає на вхід два параметри і застосовує оператор +. У цьому випадку можливі типові шаблони вхідних даних - (*Fixnum, Fixnum*) або (*String, String*), але будь-яка інша комбінація, така як (*Fixnum, String*) або (*String, Fixnum*), призводить до винятку.

Для зменшення кількості винятків система вивчає різницю між придатними та непридатними типовими шаблонами. Під час процесу пошуку операції можуть призводити до нових комбінацій типів. Така комбінація перевіряється на придатність, і в разі непридатності операція може повторюватися для знаходження кращого рішення. Система також вивчає різницю якості генераторів даних. Таким чином, якщо для конкретного типу потрібне нове вхідне значення, то вибір виконується на користь кращих генераторів. Оцінка генераторів даних ґрунтується на значенні придатності, що призначається тестовим випадкам. Генератори даних, які виробляли вхідні значення, які призвели до кращих кандидатів рішення, призведуть до більшої оцінки.

Висновок

У роботі наведено особливості реалізації RTG, інструменту для автоматичного створення тестових випадків для динамічної мови програмування Ruby. RTG може використовуватися для різних типів вхідних значень.

Встановлено різні види складності, які мають значний вплив на генерацію тестових випадків. Це самостійно визначені типи, складні та складені структури даних, вхідні дані з невеликим простором рішень та послідовності методів для зміни внутрішнього стану об'єкта. Складність послідовностей методів є чутливим фактором у автоматичному створенні тестових випадків. Чим складніше стає послідовність методів, тим складніше знаходити можливі рішення. Дуже складні та надійні послідовності методів можуть відбуватися нечасто, але в таких ситуаціях обидва генератори тестових випадків ймовірно не зможуть досягти високого покриття коду.

Метою RTG є знаходження тестових сценаріїв для охоплення якомога більшої кількості коду. Поточна версія RTG шукає придатні комбінації типів, а потім для кожного типу вибирає відповідний генератор даних. Цей проміжний крок не є обов'язковим. Більш ефективним рішенням було б безпосереднє використання набору наявних генераторів даних. У цьому випадку система шукала б комбінацію придатних генераторів замість типів даних, що може покращити якість, так і продуктивність створених тестових випадків.

Список використаних джерел

1. M. Alshraidehand L. Bottaci. Search-based software test data generation for string data using program-specific search operators: Research articles. Software Testing, Verification&Reliability,16(3):175–203, 2006.
2. A. Bouchachia. An immunegenetic algorithm for software test data generation. In HIS ’07: Proceedings of the 7th International Conference on Hybrid Intelligent Systems (HIS 2007), pp. 84–89, Washington, DC, USA, 2007. IEEE Computer Society.
3. M. Harmanand P. McMinn. A theoretical&empirical analysis of evolutionary testing and hill climbing for structural test data generation. InISSTA ’07: Proceedings of the 2007 International Symposium on Software Testing and Analysis, pp.73–83, NewYork, NY, USA, 2007,ACM.
4. S. Stinckwichand S. Ducasse. Introduction to the small talk special issue. Computer Languages, Systems&Structures, 32(2-3):85–86, 2006.
5. S. Wapplerand I. Schieferdecker. Improving evolutionary classtesting in the presence of non public methods. In ASE ’07: Proceedings of the twenty-second IEEE/ACM International Conference on Automated Software Engineering, pp.381–384, NewYork, NY, USA, 2007. ACM.
6. S. Wapplerand J. Wegener. Evolutionary unit testing of object-oriented software using a hybrid evolutionary algorithm. In CEC’06: Proceedings of the 2006 IEEE Congress on Evolutionary Computation, pp. 851–858. IEEE, 2006.
7. S. Wapplerand J. Wegener. Evolutionary unit testing of object-oriented software using strongly typed genetic programming. In GECCO ’06: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, pp. 1925–1932,NewYork, NY, USA, 2006. ACM.
8. A. Watkinsand E. M. Hufnagel. Evolutionary test data generation: a comparison of fitness functions: Research articles. Software Practice and Experience, 36(1):95–116, 2006.
9. A. Windisch, S. Wappler, and J. Wegener. Applying particle swarm optimization to software testing. In GECCO ’07: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, pp.1121–1128, NewYork, NY, USA,2007. ACM.
10. R. Zhaoand Q. Li. Automatic test generation for dynamic data structures. In SERA ’07: Proceedings of the 5th ACIS International Conference on Software Engineering Research, Management&Applications, pp. 545–549, Washington, DC,USA, 2007. IEEE Computer Society.

UDC 004.4

WEB APPLICATION FORUM

Honchar L.[1)], **Matsuk A.**[2)], **Smoliak O.**[3)], **Serediak R.**[4)], **Drichyk V.**[5)]

West Ukrainian National University
[1)]PhD;[2-4)]master; [5)]bachelor

I. Statement of the problem

The development of a Web Forum Application presents an opportunity to leverage modern technologies and approaches to enhance the user experience and functionality of online community platforms. The primary objective is to create a robust and scalable forum application that facilitates engaging discussions, knowledge sharing, and community building. Key challenges include designing an intuitive user interface [2], implementing efficient data management and storage solutions, and ensuring security and performance optimization. The successful development of the Web Forum Application requires a comprehensive understanding of client-server architecture [1], web technologies, database management, and user interface design principles. It also involves thorough testing, user feedback integration, and continuous improvement to deliver a high-quality, user-friendly forum experience.

II. The purpose of the work

The purpose of work is to create a modern and functional tool for communication and information exchange on the Internet based on such a platform as a forum. The work is aimed at researching and implementing the best practices of web development, using modern technologies and tools to build a convenient and effective user interface, ensuring the security and speed of the application, as well as creating a favorable environment for communication and information exchange in the online environment. The result of the work will be a functional Forum web application that will be able to meet the needs of users in communication and information exchange on the Internet.

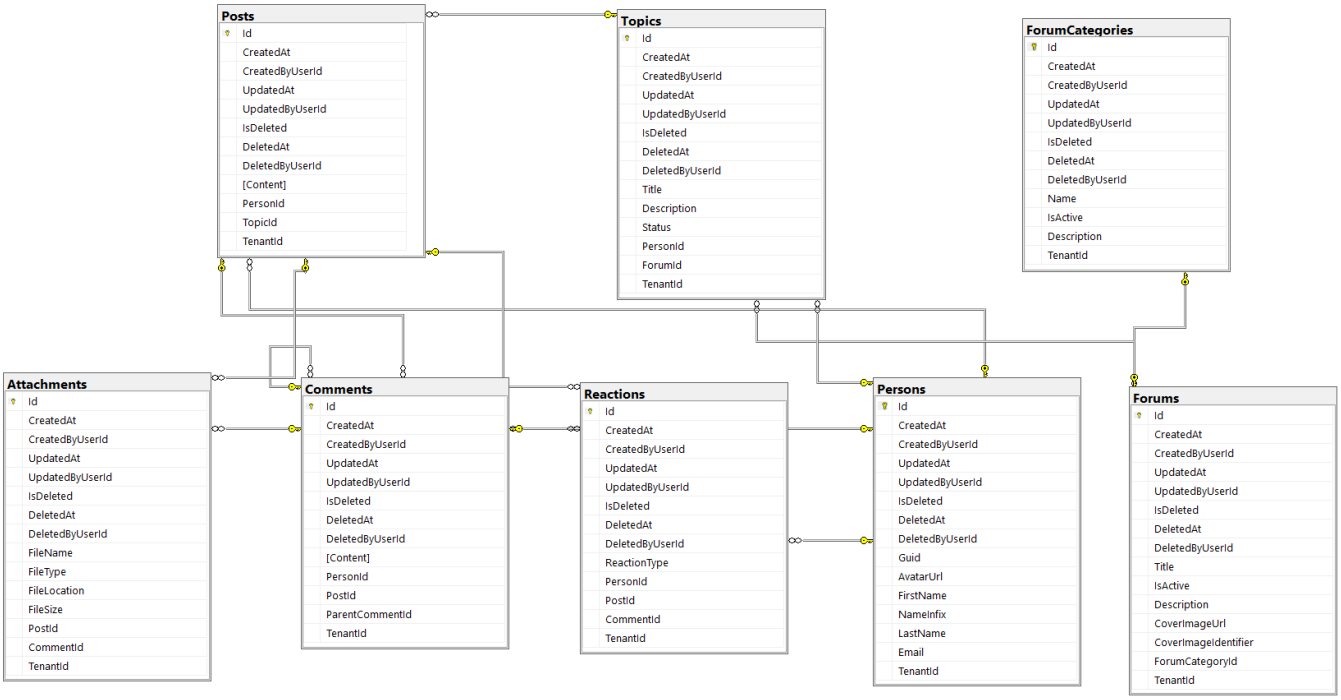
III. Main part

For the development of the Web Application Forum, a variety of technologies have been chosen to ensure its functionality, security, and user-friendliness. These technologies include:

- **Client-Server Architecture:** The application is built on a client-server architecture, which allows for efficient communication between the client-side and server-side components. This architecture is essential for handling user requests, managing data, and ensuring a seamless user experience.
- **.NET 8.0 and ASP.NET:** These frameworks were chosen to write the backend of application and create endpoints. With the help of ASP.NET, it is very convenient to build and support applications according to the MVC model. These frameworks provide a robust foundation for building web applications. They offer a wide range of features and tools that simplify the development process and improve application performance [3].
- **Entity Framework Core 8.0:** This ORM (Object-Relational Mapping) framework simplifies database interactions, making it easier to work with database entities in the application code. It ensures efficient data management and enhances the application's scalability[4]. EntityFramework, together with the Linq library, is a powerful tool in the hands of a developer when writing fast, easy, and understandable data base queries.
- **Refit:** Refit is used to simplify the consumption of RESTful APIs. It provides a clean and type-safe way to define API interfaces, making it easier to integrate external services into the application [5].We use it to integrate our app with a CloudFlare API.
- **Swagger:** Swagger is used for API documentation, providing a way to document and test APIs. It helps understand the application's API structure and use it effectively[6].
- **Vue.js:**We use Vue.js to write the user interface (front-end), because it is easy and clear to use. It is also one of the top front-end frameworks at the moment. It provides a reactive and component-based framework that makes it easy to create dynamic and interactive UI components [7].
- **Vue-i18N:** Vue-i18N is used for internationalization in the application, allowing it to support multiple languages and regions. This ensures that the application is accessible to users from different parts of the world.

- **Bootstrap-Vue:** Bootstrap-Vue is used for building responsive and mobile-first UI components. It ensures that the application looks great and works well on a variety of devices and screen sizes.
- **Date-Fns:** Date-Fns is used for date manipulation and formatting in the application. It ensures that dates are displayed correctly and consistently across different parts of the application.
- **CloudFlare:**For image storage, our choice fell on CloudFlare, since storing a lot of photos in a database is not a good idea. We chose CloudFlare because it has good API documentation that makes it easy to integrate and use, as well as a great community and support.CloudFlare is used for CDN (Content Delivery Network) services, which improve the performance and reliability of the application. It caches content and optimizes delivery to users, ensuring a fast and seamless user experience [8].
- **MSSQLServer:**MSSQLServer is used as the database management system for the application. It provides a reliable and scalable database solution for storing and managing the application's data [9].

These technologies are essential for the development of a modern and functional Web Application Forum. They ensure that the application is secure, efficient, and user-friendly, providing a seamless experience for users.



Picture1 – Database diagram

Conclusion

The work aimed to research and implement best practices in web development, utilizing modern technologies and tools to build a user-friendly interface, ensuring application security and speed, and fostering a conducive environment for online communication and information exchange.

The result of this endeavor will be a functional Forum web application designed to meet users' needs for online communication and information exchange. Through the application of best practices and modern technologies, the developed forum platform is expected to provide an effective and convenient platform for users to engage in discussions, share knowledge, and build communities online.

List of used sources

1. Client-server architecture.https://www.britannica.com/technology/client-server-architecture
2. 3 Key Principles for Creating an Intuitive User Interface.https://bootcamp.uxdesign.cc/3-key-principles-for-creating-an-intuitive-user-interface-6189a6165134
3. Introduction to .NET https://learn.microsoft.com/en-us/dotnet/core/introduction
4. Entity Framework Core https://learn.microsoft.com/en-us/ef/core/
5. Using Refit in .NET.https://medium.com/p/0843bb199987
6. API Documentation https://swagger.io/solutions/api-documentation/
7. The Progressive JavaScript Framework https://vuejs.org/
8. Cloudflare API https://developers.cloudflare.com/api/
9. https://www.microsoft.com/en-us/sql-server

УДК 004.4

ОПТИМІЗАЦІЯ ПРОЦЕСУ ПОШУКУ ТОВАРІВ НА МАРКЕТ ПЛЕЙСАХ НА ОСНОВІ JSOUP

Манжула В.В.[1)], Крепич С.Я.[2)], Шерстюк Ю.Ю.[3)]
Західноукраїнський національний університет
[1),3)] магістрант; [2)]к.т.н., доцент

І. Постановка проблеми

Ефективний пошук елементів на HTML сторінці за допомогою Java залежить від багатьох факторів. Основна проблема полягає в здатності ефективно виявляти та вибирати бажані елементи зі складного структурного дерева HTML. Це означає розуміння специфіки HTML-коду та здатність адаптувати алгоритм до різних форматів сторінок та їхньої різноманітності.

Однією з ключових задач є ефективне використання jsoup для навігації та вибору елементів. Для досягнення цієї мети потрібно ретельно вивчити документацію бібліотеки та розробити стратегію пошуку, що враховує різноманітність структур HTML. Додатково, виникає завдання оптимізації алгоритму для швидкого та ефективного пошуку на великих обсягах даних.

ІІ. Мета роботи

Метою даної роботи є розробка алгоритму пошуку елементів на HTML сторінці з використанням мови програмування Java та бібліотеки jsoup. Головними завданнями є створення ефективного та надійного алгоритму, здатного вибирати бажані елементи з будь-якої HTML-структури, а також оптимізація процесу пошуку для забезпечення швидкості та ефективності на великих обсягах даних. Ця робота спрямована на вдосконалення інструментів для обробки та аналізу веб-контенту, що є важливим у контексті розвитку сучасних інтернет-технологій та веб-аналітики.

ІІІ. Проектування алгоритму пошуку елементів

Jsoup– це бібліотека для обробки HTML та XML даних в Java. Вона надає зручні методи для отримання, зміни та видалення елементів на HTML сторінці. Ефективні методи для пошуку елементів на HTML сторінці за допомогою Jsoup забезпечує використання таких засобів:

- CSS-селекторів.Jsoup дозволяє використовувати CSS-селектори для вибору елементів. Надає можливість використовувати метод select(), передавши до нього CSS-селектор, щоб отримати список знайдених елементів;
- ID або класів. Можливість шукати елементи за їх ID або класами, використовуючи CSS-селектори;
- методів getElementsBy...().Jsoup також має набір методів getElementsBy...(), які дозволяють вам шукати елементи за їх тегом, класом, ID тощо;
- Пошук елементів за атрибутами;
- Поєднання кількох методів пошуку. Для складних запитів можна поєднувати кілька методів пошуку, наприклад, спочатку вибрати всі елементи з певним тегом, а потім вже серед них вибирати тільки ті, що мають певний клас.

Дані засоби Jsoup надають можливість здійснювати ефективний пошук елементів на HTML сторінці та виконувати різноманітні маніпуляції з ними.

Схематично, реалізацію алгоритму пошуку елементів можна зобразити як на рисунку 1.

Отримання HTML-коду сторінки. Цей етап включає в себе взаємодію з веб-сервером або файловою системою для отримання HTML-коду веб-сторінки. При взаємодії з веб-сервером ми виконуємо HTTP-запит до веб-сторінки і отримуємо відповідь, яка містить HTML-код сторінки. При роботі з локальними файлами ми читаємо збережений HTML-файл з файлової системи.

Створення об'єкту Document з використанням jsoup. Після отримання HTML-коду ми використовуємо бібліотеку jsoup для його парсингу та створення об'єкту типу Document. Цей об'єкт представляє собою дерево DOM сторінки і дозволяє легко навігуватися по її структурі та взаємодіяти з елементами.

Jsoup.connect("http://example.com").get();

*Використання селекторів для пошуку елементів.*Jsoup дозволяє використовувати CSS-подібні селектори для пошуку конкретних елементів на сторінці. Селектори можуть бути використані для вибору елементів за їх тегами, класами, ідентифікаторами, атрибутами тощо. Наприклад, select("div.content") вибирає всі елементи з класом "content".

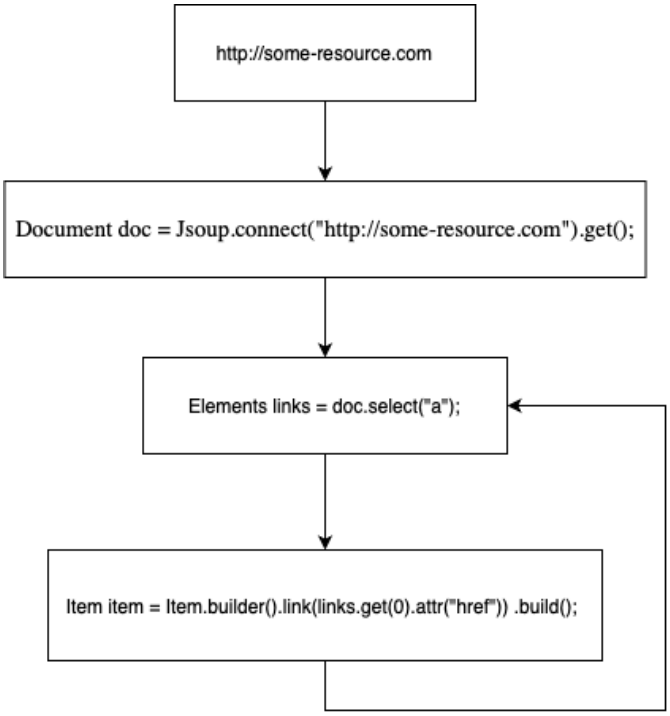


Рисунок 1 – Алгоритм пошуку елементів на сторінці

Обробка результатів пошуку. Після виконання пошуку ми отримуємо колекцію елементів, які задовольняють наш селектор. Ми можемо ітерувати цю колекцію та виконувати різні операції з кожним елементом, такі як отримання текстового вмісту, значень атрибутів, додавання класів, зміна стилів тощо. Згідно з цілями нашого алгоритму, додаємо текстовий вміст елементів або значення їхніх атрибутів до моделі Item.

```
Elementslinks = doc.select("a");
for (Elementlink : links) {
        Itemitem = Item.builder().
                .link(link.attr("href"))
                .build();        }
```

Висновок

У даній роботі розроблено алгоритм пошуку товарів на українських маркет плейсах засобами Jsoup, яке дозволяє здійснювати швидкий та глобальний пошук товарів. Ефективність запропонованого алгоритму обумовлена перевагами використання Jsoup. Зокрема,простий та зрозумілий API, що робить роботу з HTML даними легкою та зручною для розробників; ефективний пошук елементів,завдяки механізму використання CSS-селекторів та методів для отримання елементів за їх тегами, класами, ID та атрибутами; підтримка обробки складних HTML документів; підтримка функцій парсингу та обробки HTML і XML. Загалом, використання Jsoup є ефективним способом роботи з HTML даними в Java, що дозволяє швидко та зручно вибирати, змінювати та аналізувати вміст веб-сторінок.

Список використаних джерел

1. Lanet [Електронний ресурс]. — Режим доступу: https://lanet.click/ppc/reklama-v-prais-ahrehatorakh (2022).
2. JSoup[Електронний ресурс]. — Режим доступу: https://jsoup.org (2022).
3. uk.wikipedia.org – Вільна енциклопедія [Електронний ресурс]. – Режим доступу: http://uk.wikipedia.org для доступу до інформаційних ресурсів авторизація не потрібна — Назва з екрану. Вікіпедія. Вільна енциклопедія.

УДК 711.4:004.9

РОЗУМНЕ МІСТО: ОЦІНЮВАННЯ РОЗУМНОСТІ МІСТА

Бонар В.О. [1)]
Тернопільський національний технічний університет імені Івана Пулюя
[1)]магістрант

І. Постановка проблеми

Сучасний світ швидкий і динамічний. З кожним роком урбанізація населення зростає. Це створює нові виклики, оскільки старі методи управління містом можуть не працювати, або потребують масштабування. Міста відіграють важливу роль в забезпеченні благополуччя населення включаючи освіту, працевлаштування, соціальну стабільність. Велику роль у розвитку міста відіграють суспільство і влада.

Уряд міста покладається на інформаційно-комунікаційні технології (ІКТ), щоб покращувати життя резидентів. Сучасні ІКТ включають в себе різні інформаційні сенсори, вбудовані в світлофори, дороги і транспортні засоби, аналіз великих даних і мережеві технології. Зібрана інформація може бути використана для розуміння реалій життя громадян, що в свою чергу допомагає у створенні “смарт” рішень.

ІІ. Критерії для оцінки рівня розумності міста

Для оцінки рівня розумності міста потрібно обрати комплекс критеріїв, за якими буде проводитись оцінка. Це завдання має свої виклики, оскільки індикаторів може бути дуже багато і вони можуть мати різні вагові коефіцієнти. При виборі критеріїв потрібно керуватись такими правилами:

- Можливість об’єктивно оцінити критерій.
- Надійність - критерій має бути чітко визначений і бути однозначним.
- Актуальність - критерій повинен бути актуальним для міста.
- Інтуїтивність - критерій має бути інтуїтивно зрозумілим для людей.
- Ексклюзивність - критерій не має повторюватись.

За метою індикатори поділяють на групи: оцінювання, дослідження та валідація. Оцінювання використовують для розуміння ефективності й стійкості міста. Для оцінки поточного стану якогось окремого виміру міста, наприклад середовища, зазвичай використовують дослідження. Для підтвердження результатів певних дій проводять валідацію.

За методом розрахунку критерії поділяють на одиничні, агреговані, композитні та індекси. Одиничні індикатори показують одну властивість, яка вимірюється в певній одиниці. Наприклад: концентрація викидів CO2 і кількість приватних автомобілів у місті. Агреговані базуються на двох або більше властивостях, які вимірюються в одній одиниці. Прикладом є інтенсивність викидів CO2 за типом джерела (виробництво, транспорт, побутові споживачі). Композитні критерії об’єднують кілька ознак заданих характеристик розумного міста. У результаті виходить єдине значення в загальній одиниці. Прикладом композитного індикатора є вимірювання незадоволення, що розраховується як відсоток населення, яке страждає від шуму, запаху та візуального забруднення. Індекси — це індикатори, які агрегують кілька вимірів розумного міста в одному безрозмірному значенні. Індекс людського розвитку, наприклад, є підсумковим показником середнього досягнення в трьох вимірах людського розвитку, пов’язаних із здоров’ям, освітою та рівнем життя.

За типом оцінки критерії поділяють на три групи: дохід, результат і процес. Дохід - кількісна або якісна оцінка таких властивостей як кількість безробітних, приріст ВВП тощо. За результатом оцінюються наслідки людської діяльності. Наприклад, ефективне використання води або енергоресурсів з відновлювальних джерел. Людська активність або ініціатива, такі як відвідування музеїв і бібліотек оцінюються процесом.

Дуже важливо в критеріях оцінки підібрати одиницю виміру. Основне для цього - це її сумісність з різними містами та проектами, які можуть відрізнятись розміром, мати різні цілі тощо. Тому замість використання абсолютних показників рекомендують використовувати відносні, наприклад, відсотки.

Для спрощення оцінки рівня розумності міста критерії можна категоризувати: “смарт” природа, “смарт” життя, “смарт” транспорт, “смарт” управління, “смарт” люди та “смарт” економіка [1].

Основні категорії розумної природи міста: вода, енергія, забрудненість, відходи, земля, зелене середовище. Кожна категорія є важливою для благополуччя міста і має свої критерії оцінки. В оцінці води беруть до уваги ефективність використання води, загальне споживання води, якість води, ефективність постачання води. Для оцінки енергоспроможності міста використовують такі метрики: споживання енергії різними секторами, частота та тривалість вимкнень електроенергії, енергоефективність. Забрудненість визначається якістю повітря та рівнем шуму. Основні причини забрудненості: викиди CO2, важка техніка, будівництва, генератори, виробництва. Оцінки можливості міста справлятися з відходами включають: кількість відходів, які переробляють, спалюють чи закопують. Земля оцінюється ефективністю її використання, рівнем її деградації з часом, опустелення тощо. Екологічність міста є одним з основних показників сталого розвитку. Зелене середовище оцінює стратегії для покращення зовнішнього середовища і відповідність міжнародним стандартам.

Розумне управління. Прозорість і зрозумілість політики управління містом є основним фактором для втілення концепції розумного міста. Основні критерії оцінки: рівень корупції, прозорість бюрократії, відкритість влади, кількість та якість соціальних послуг, рівність різних верств населення.

Розумна економіка. Економіка має чи не найважливіший вплив на рівень людського життя. Оцінювання економічного благополуччя міста включає в себе не тільки загальне багатство і статки міста, а ще й як воно розподілене між населенням, рівень залученості резидентів в економіку, перспективи економічного зростання, міжнародну залученість.

Розумне життя. Розумне життя об’єднує різні аспекти людського життя: здоров’я, освіту, безпеку, побут, культуру, туризм та архітектуру. Основні оцінки: доступність медичних послуг, тривалість життя, ризик смерті, доступність шкіл, умови життя, вартість оренди, кількість злочинів, якість нових будівель тощо.

Розумні люди. Стратегія розвитку міста повинна включати в себе програми, що підтримують демографічний рівень. Відношення працездатного населення до дітей та людей пенсійного віку є основним критерієм цього. Також важливим фактором розвитку будь-якого міста є освіченість людей, які в ньому проживають.

Розумний транспорт. Великі міста дедалі більше зіштовхуються з проблемою транспорту. Керування дорожнім рухом є серйозним викликом. Наявність, якість та доступність громадського транспорту оцінюють його доступністю в різних частинах міста, кількістю річних поїздок на душу населення, зручністю транспорту. Кількість та інтенсивність заторів ж оцінюють часом подорожі в пікові години.

ІІ. Оцінка розумності міст

Україна стоїть лише на початку свого шляху розвитку розумних міст. Хоча вже є міста, які в тому чи іншому вигляді реалізують окремі концепції розумних міст, нам бракує стратегічного бачення і синергії всіх систем. В умовах війни розвиток дещо сповільнився і з’явились нові проблеми.

Згідно з дослідженням IMD за оцінками SmartCityIndex [2] у 2021 місто Київ було на 82-у місці у рейтингу найрозумніших міст світу (див.табл. 1).

Таблиця 1 - Оцінка розумності м. Київ у 2021 р.[2]

Критерій	Оцінка (0-100)	Критерій	Оцінка (0-100)
Базові санітарні умови відповідають потребам найбідніших районів	54.7	Більшість дітей має можливість навчатися у якісній школі	62.5
Громадська безпека	46.3	Послуги з переробки	39.0
Забруднення повітря не є проблемою	28.0	Меншини не відчувають утиску	49.7
Пошук житла з орендною платою в розмірі 30% або менше місячної зарплати не проблема	31.3	Доступ до онлайн оголошень про роботу значно спростив пошук зайнятості	82.7
Медичні послуги	41.1	Школи добре навчають ІТ навичкам	54.6
Можливість онлайн запису на прийом до лікаря	63.1	Затори не є проблемою	18.7

Підприємці стимулюють створення нових робочих місць	55.3	Місцеві установи забезпечують можливість навчання протягом усього життя	61.4
Онлайн звернення про проблеми з утримання міста швидко вирішуються	55.0	Міські онлайн сервіси спростили процес відкриття нового бізнесу	52.7
Мешканцям зручно віддавати непотрібні речі за допомогою веб-сайту або додатку	50.7	Поточна якість Інтернету відповідає потребам	74.1
Доступ до міських послуг покращився завдяки безкоштовному громадському Wi-Fi	58.0	Інформація про рішення місцевої влади є легкодоступною	61.0
Камери відеоспостереження дозволяють жителям почуватися в безпеці	53.0	Занепокоєння щодо корупції в міській владі відсутнє	23.4
Мешканці мають можливість моніторити забруднення повітря за допомогою веб-сайту або додатку	46.1	Мешканці беруть участь у прийнятті рішень місцевої влади	33.7
Мешканці надають зворотній зв'язок про проєкти місцевого самоврядування	55.0	Онлайн доступ мешканців до міських фінансів сприяв зниженню рівня корупції	34.4
Громадський транспорт задовільний	50.3	Додатки каршерингу зменшили затори	43.4
Інтернет голосування забезпечило більшу кількість виборців	48.6	Онлайн платформа для збору ідей мешканців покращила життя міста	54.7
Додатки, які допомагають знайти вільне паркувальне місце, скорочують час подорожі	59.8	Опрацювання ідентифікаційних документів онлайн скоротила час очікування	72.9
Затори зменшилися завдяки прокату велосипедів	48.6	Культурна діяльність (шоу, бари, музеї)	72.9
Користування громадським транспортом стало простішим завдяки онлайн розкладу та продажу квитків онлайн	70.0	Можливість онлайн купівлі квитків на вистави та в музеї полегшила їхнє відвідування	83.0
Інформація про затори надається містом через мобільні телефони	62.1	Послуги з пошуку роботи доступні	70.4
Зелені насадження	65.6		

Найгірші показники у Києві пов'язані зі заторами (18.7), корупцією міської влади (23.4), пошуком житла з орендною платою в розмірі 30% або менше місячної зарплати (31.3), залученістю мешканців у прийнятті рішень місцевої влади (33.7).

Найкраще Київ реалізував різні онлайн сервіси: онлайн доступ до оголошень про роботу (82.7), онлайн купівля квитків на вистави та в музеї (83.0), опрацювання ідентифікаційних документів онлайн (72.9), швидкість і надійність Інтернету (74.1). Загалом решта показників або близькі до середнього (50), або вищі.

Висновок

В даній роботі розглянуто важливість оцінки розумності міста, методи та критерії, які використовують для цього. Надано перелік вимог для критеріїв оцінки та наведено практичні приклади. Суспільство і влада мають докладати багато зусиль для розвитку сучасного розумного міста і моніторити результати та зміни згідно з оцінками наданих критеріїв. Надано оцінку реалізації концепції розумного міста на прикладі Києва.

Список використаних джерел

1. Smart Cities Evaluation – A Survey of Performance and Sustainability Indicators, Dessislava Petrova-Antonova, Sylvia Ilieva, https://www.researchgate.net/publication/328460614
2. Smart City Index 2021, , p. 66 https://www.imd.org/smart-city-observatory/home/2021

УДК 004.896.8

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗНАХОДЖЕННЯ АНОМАЛІЙ В ТЕХНОЛОГІЧНИХ ДАНИХ

Поворозник Б.С.[1)],**Марценюк Є.О.**[2)], **Куривчак Д.І.**[3)], **Мархоцький Н.Ю.**[4)], **Даньків А.В.**[5)]
Західноукраїнський національний університет
[1)]магістрант;[2]к.т.н., доцент;[3-4)]магістрант;[5)]аспірант

I. Постановка проблеми

Розуміння та використання даних, які генеруються технологічними системами, стає все важливішим для ефективного управління та забезпечення надійності виробничих процесів.

Однією з ключових переваг кіберфізичних систем є їх здатність інтегрувати фізичні об'єкти з програмним забезпеченням, що дозволяє збирати, аналізувати та діяти на основі зібраних даних в реальному чи близькому до реального часу. Прогрес у цій області може значно підвищити ефективність, надійність та безпеку виробничих процесів [1].

Щодо складнощів у впровадженні інтелектуальних систем аналізу даних у виробничому середовищі, то згадані проблеми є цілком актуальними. Врахування нестаціонарності, складних залежностей та унікальної природи технологічних даних вимагає розробки високоефективних алгоритмів аналізу та моделей, які здатні працювати в реальному часі та реагувати на зміни виробничих умов .

Проте, розвиток науки про дані та швидкий прогрес в області штучного інтелекту та машинного навчання відкривають нові можливості для подальшого розвитку і вдосконалення систем виявлення аномалій у виробничих процесах. Із поглибленням розуміння цих складних проблем та застосуванням передових технологій, можна досягти значних досягнень у покращенні безпеки та продуктивності виробничих систем.

II. Мета роботи

Метою дослідження є розробка математичного та програмного забезпечення для знаходження аномалій в технологічних даних.

III. Метод виявлення аномалій в технологічних даних

Складність виявлення аномалій у технологічних сигналах полягає у їхній "нестаціонарності". При цьому з урахуванням постановки завдання виявлення аномалій повинно відбуватися без вчителя, тобто без будь-якої апріорної інформації про властивості сигналу. У той же час, математичний апарат системи повинен мати інваріантні властивості аналізованих даних. Ці фактори вимагають від алгоритму виявлення аномалій адаптивних властивостей, що дозволяють витягувати інформативні ознаки із сигналів без додаткової інформації.

Тому пропонується підхід до побудови алгоритму G з використанням перетворення Гільберта для отримання амплітудно-частотного спектра та подальшого його аналізу. Основна ідея пропонованого підходу полягає у розкладанні досліджуваного сигналу $X(t)$ в амплітудний спектр та застосуванні моделі виявлення M аномалій до кожного рівня розкладання.

При цьому модель виявлення аномалій як вхідну інформацію приймає миттєву амплітуду, одержану шляхом застосування перетворення Гільберта до внутрішніх мод сигналу (IMF), отриманих в результаті емпіричної модової декомпозиції сигналу (метод Хуанга). Зауважимо, що для аналізу використовується саме миттєва амплітуда для відповідної IMF, а миттєві частота і фаза не беруться до уваги через низьку локалізацію особливостей сигналу. Результати виявлених аномалій кожного рівня розкладання представляються як підсумковий вектор міток. Нижче представлений формальний покроковий опис алгоритму:

Крок 1. Згідно з постановкою задачі сигнал $X(t)$ поділяється на навчальну $X(t)^{train}$ та тестову вибірку $X(t)^{test}$.

Крок 2. Для кожної частини сигналу виконується процедура вилучення $IMF(t)^{train}$ та $IMF(t)^{test}$ таким чином, що:

$$X(t) = \sum_{j=1}^{n} IMF_j(t) + r_n(t) \quad (1)$$

де n - кількість емпіричних мод, $r_j = r_{j-1} - IMF_j(t)$.

Крок 3. Виявляється мінімальний рівень $T_{\min}$ розкладання IMF через порівняння числа рівнів розкладання для $IMF(t)^{train}$ та $IMF(t)^{test}$:

$$T_{\min} = \min(T^{train}, T^{test}) \tag{2}$$

Крок 4. Для $IMF(t)^{train}$ і $IMF(t)^{test}$ застосовується перетворення Гільберта, після чого обчислюються образи Гільберта $H(t)^{train}$ та $H(t)^{test}$:

$$H(t)^{train} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{IMF(\tau)^{train}}{t-\tau} d\tau \tag{3}$$

$$H(t)^{test} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{IMF(\tau)^{test}}{t-\tau} d\tau \tag{4}$$

де інтеграли в (3,4) розуміються у сенсі головного (P.V. – principal value) значення Коші.

Крок 5. Модель M навчається на миттєвій амплітуді $\left|H(t)_k^{train}\right|$, де $k = 1..T_{\min}$:

$$M(\varphi, \theta, |H(t)_k^{train}|) \rightarrow M' \tag{5}$$

де M' – навчена модель.

Крок 6. Далі для $H(t)_k^{test}$ виконується процедура виявлення аномалій та формується вектор прогнозу F:

$$M'(\varphi, \theta, |H(t)_k^{test}|) \rightarrow F \tag{6}$$

На рисунку 1 представлено графічне, деталізоване представлення запропонованого методу. Підхід складається із двох блоків. Перший блок виконує процедуру перетворення Гільберта для аналізованого сигналу, другий блок навчає та тестує моделі для кожного отриманого образу Гільберта $H(t)$. При цьому як моделі виявлення можуть застосовуватись різні моделі M для кожного сигналу. Далі прогноз кожної моделі об'єднуються у спільний вектор F оцінок аномалій. Відмінною рисою даного підходу є те, що кожен елемент спектра сигналу аналізується окремою моделлю M_k. У разі виникнення аномалії в сигналі, на певному рівні розкладання, дана аномалія локалізується вираженим шляхом і буде виявлена відповідною моделлю. При цьому рівень розкладання контролюється автоматично за рахунок обчислення $T_{\min}$, що дозволяє аналізувати максимальну кількість рівнів розкладання. Це дозволяє підвищити як точність виявлення, а й інтерпретованість підходу.

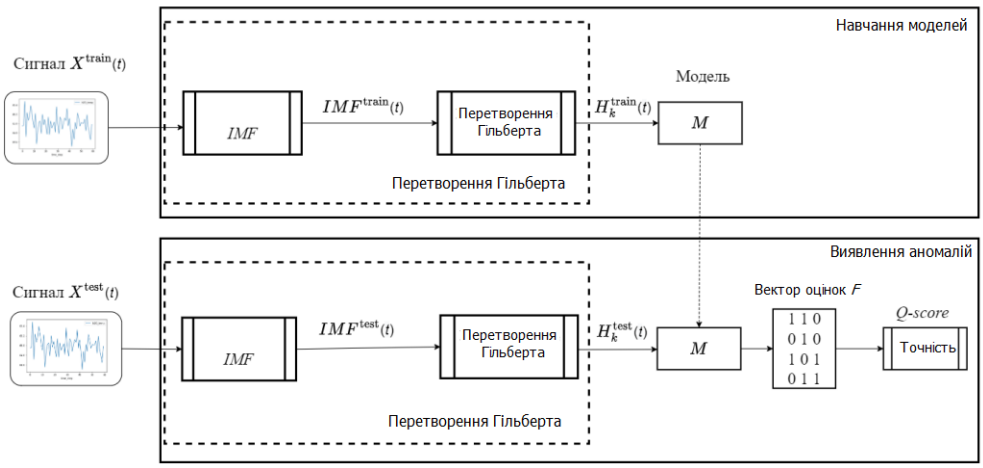


Рисунок 1– Схема методу виявлення аномалій

Висновок

Запропонований підхід до виявлення аномалій на основі перетворення Гільберта відкриває нові можливості для ефективного аналізу сигналів і виявлення незвичайних подій. Використання цього методу дозволяє локалізувати аномалії в елементах спектру сигналу та аналізувати їх окремо, що сприяє більш точному виявленню та ідентифікації. Основна новизна цього підходу полягає у тому, що кожен елемент амплітудного спектра сигналу аналізується окремо моделлю виявлення. Це означає, що кожен компонент сигналу розглядається індивідуально, що дозволяє ефективно виявляти аномалії, які можуть відбуватися на різних частотах або в різний час.

Список використаних джерел

1. В.П. Шкодирєв, К.І. Ягафаров, В.А. Баштовенко, Є.Е. Ільїна. Огляд методів виявлення аномалій в потоках даних. URL: http://ceur-ws.org/Vol-1864/paper_33.pdf

УДК 004.896.3

МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ НЕЧІТКОЇ КЛАСТЕРИЗАЦІЇ ДАНИХ В ПАРАЛЕЛЬНІЙ СИСТЕМІ УПРАВЛІННЯ БАЗАМИ ДАНИХ

Варчук Д.А.[1)], **Пришляк О.В.**[2)], **Виноградова Д.Р.**[3)], **Дементьєв Р.В.**[4)], **Марценюк М.О.**[5)]
Західноукраїнський національний університет
[1)]магістрант;[2)]аспірант;[3-5)]студент

I. Постановка проблеми

У сучасному інформаційному суспільстві одним із ключових феноменів, що має значний вплив на область обробки даних, є великі дані [1,2]. У зв'язку з розвитком сучасних технологій, таких як соціальні мережі, електронні бібліотеки, геоінформаційні системи та інші, виникає широкий спектр додатків, в яких накопичуються неструктуровані дані.

Системи управління базами даних (СУБД), побудовані на основі реляційної моделі даних залишаються основним і найбільш популярним інструментом для управління даними на сьогоднішній день. Ефективна обробка та аналіз великих сховищ даних вимагає використання паралельних СУБД на високопродуктивних обчислювальних системах[1]. Паралельні СУБД будуються на основі концепції фрагментного паралелізму, що передбачає розбиття таблиць баз даних на горизонтальні фрагменти, які можуть оброблятися незалежно на різних вузлах багатопроцесорних систем.

II. Мета роботи

Метою дослідження є розробка математичного забезпечення для нечіткої кластеризації даних в паралельній системі управління базами даних.

III. Метод нечіткої кластеризації даних в паралельній СУБД

Нечітка кластеризація даних у СУБД виконується за допомогою інтеграції алгоритму Fuzzy c-Means (FCM) [2] в паралельну СУБД відповідно до наступного сценарію. Вихідні дані, проміжні результати обчислень і вихідні дані алгоритму подаються у вигляді реляційних таблиць, які входять до бази даних зі спеціально розробленою схемою. Обчислення реалізуються у вигляді запитів SQL до відповідних таблиць. Таблиці бази даних піддаються горизонтальній фрагментації та розподіляються по вузлах кластерної обчислювальної системи. Кожен екземпляр паралельної СУБД запускається на власному вузлі і обробляє власний фрагмент таблиці.

У подальшому описі використовуються такі позначення:
$d \in N$ - розмірність простору об'єктів, що кластеризуються; $l \in N : 1 \le l \le d$ - номер координати об'єкта, що кластеризується; $n \in N$ - кількість об'єктів, що кластеризуються; $X \subset R^d$ - множина об'єктів, що кластеризуються; $i \in N : 1 \le j \le k$ - номер об'єкта; $x_i \in N$ - і-й об'єкт; $k \in N$ - кількість кластерів; $i \in N : 1 \le j \le k$ - номер кластера; $C \subset R^{k \times d}$ - матриця, що містить центри кластерів (матриця центроїдів); $c_j \subset R^d$ - центр кластера j; $x_{il}, c_{jl} \in R$ - координати об'єктів x_i, c_j відповідно; $U \subset R^{n \times k}$ - матриця ступенів приналежності, де число $u_{jj} \in R$ показує ступінь належності об'єкта x_i кластеру j і виконуються такі властивості:

$$\forall i, j[u_{ij}] \in [0,1] \forall i \sum_{j=1}^{k} u_{ij} = 1 \quad (1)$$

$\rho : R^d \times R^d \to R_+ \cup 0$ - функція відстані, яка використовується для визначення близькості об'єкта x_i та центроїда c_j (параметр алгоритму);

$m \in R : m > 1$ - показник нечіткості цільової функції (параметр алгоритму);

J_{FCM} – цільова функція.

Алгоритм FCM виконує мінімізацію цільової функції J_{FCM}, яка має такий вигляд:

$$J_{FCM}(X, k, m) = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{i,j}^m, \rho^2(x_i, c_j) \quad (2)$$

Нечітке розбиття вихідної множини об'єктів досягається за допомогою мінімізації цільової функції (2). На кожній ітерації відбувається оновлення матриці центроїді C та матриці степенів належності U за такими формулами:

$$\forall i, lc_{jl} = \frac{\sum_{i=1}^{N} u_{i,j}^{m} x_{il}}{\sum_{i=1}^{N} u_{i,j}^{m}} \quad (3)$$

$$u_{i,j} = \sum_{t=1}^{K} \left(\frac{\rho^2(x_i, c_j)}{\rho^2(x_i, c_t)} \right)^{\frac{2}{1-m}} \quad (4)$$

Нехай s - номер ітерації, $u_{i,j}^{s}$ та $u_{i,j}^{s+1}$ елементи матриці U на s-му та $(s+1)$-му кроках відповідно, $\varepsilon \in (0,1) \subset R$ - порогове значення зупинки алгоритму.

Вхідними даними алгоритму FCM є множина об'єктів $X = (x_1, x_2, ..., x_n)$ кількість кластерів k, ступінь нечіткості m і граничне значення зупинки ε. Алгоритм повертає матрицю степенів належності U.

Множина наборів реалізується за допомогою класу vector зі стандартної бібліотеки класів C++ (Standard TemplateLibrary). Даний клас реалізує масив елементів, що належать одному типу даних та забезпечує операції додавання, видалення елементів, доступ до елемента за його індексом та ітерацію елементів масиву. Кожен із векторів *DASHED* і *SOLID* забезпечує зберігання двох множин наборів алгоритмів: *DashedCircle* і *DashedBox* та *SolidCircle* і *SolidBox* відповідно.

Зберігання двох множин у рамках одного вектора забезпечує менші витрати, ніж виділення одного вектора на кожну множину елементів: для переміщення елемента з одної множини до іншої достатньо змінити значення поля **shape**.

Набір елементів, таким чином, реалізується як структура з наступними основними полями: *Mask* - бітова маска набору; *k*- кількість елементів у наборі; *stop* - лічильник частини транзакцій, в яких перевірено наявність даного набору; *supp* - підтримка цього набору; *shape* - вид цього набору (*BOX* або *CIRCLE*, або *NULL*).

Алг. PDIC(IN $\mathcal{B}$, *minsup*, M; OUT $\mathcal{L}$)

1: $SOLID.init()$; $DASHED.init()$
2: **for all** $i \in 0..m-1$ **do**
3: $I.shape \leftarrow$ NIL
4: $I.mask \leftarrow SetBit(I.mask, i)$
5: $I.stop \leftarrow 0$; $I.supp \leftarrow 0$; $I.k \leftarrow 1$
6: $SOLID.push\_back(I)$
7: **end for**
8: $k \leftarrow 1$
9: $stop_{max} \leftarrow \lceil \frac{n}{M} \rceil$; $stop \leftarrow 0$
10: FIRSTPASS($SOLID$, $DASHED$)
11: **while not** $DASHED.empty()$ **do**
12: $stop \leftarrow stop + 1$
13: **if** $stop > stop_{max}$ **then**
14: $stop \leftarrow 1$
15: **end if**
16: $first \leftarrow (stop-1) \cdot M$; $last \leftarrow stop \cdot M - 1$
17: $k \leftarrow k + 1$
18: COUNTSUPPORT($DASHED$)
19: PRUNE($DASHED$)
20: MAKECANDIDATES($DASHED$)
21: CHECKFULLPASS($DASHED$)
22: **end while**
23: $\mathcal{L} \leftarrow \{I \mid I \in SOLID \wedge I.shape =$ BOX$\}$
24: **return** $\mathcal{L}$

Рисунок 1–Паралельна реалізація пошуку наборів

Розпаралелювання піддаються наступні стадії алгоритму: підрахунок підтримки, знаходження та відкидання наборів-кандидатів, які свідомо не є частими та перевірка завершення перегляду наборів-кандидатів по всій базі даних транзакцій. Підрахунок виконується за допомогою двох вкладених циклів: зовнішній розпаралелюється і виконується за наборами-кандидатами, а внутрішній - за транзакціями.

Висновок

Запропоновано метод нечіткої кластеризації даних за допомогою паралельної СУБД. Виконано проектування схеми бази даних, що дозволяє зберігати вихідні та проміжні дані в реляційних таблицях і виконувати обчислення за допомогою агрегатних функцій SQL. Результати проведених обчислювальних експериментів показують, що запропонований алгоритм для СУБД на великих обсягах даних є більш ефективним у порівнянні з традиційною реалізацією нечіткої кластеризації, що передбачає використання оперативної пам'яті.

Список використаних джерел

1. Рудакевич Т. М. Розширення функціональних можливостей СКБД Postgresql для оптимізації пошуку інформації у файлах баз даних / Т. М. Рудакевич, Г. Г. Цегелик // Вісник Національного університету "Львівська політехніка". – 2010. – № 673 : Інформаційні системи та мережі. – С. 181-194.
2/ Garcia, A.J.; Toril, M.; Oliver, P.; Luna-Ramirez, S.; Ortiz, M. Automatic Alarm Prioritization by Data Mining for Fault Management in Cellular Networks. ElsevierSci. DirectExpertSyst. Appl. 2020, 158, 113526.

УДК 004.891.3

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ПРОВЕДЕННЯ АДАПТИВНОГО НАВЧАННЯ СТУДЕНТІВ

Курівчак Д.І.[1)], **Пришляк О.В.**[2)], **Хаєцький М.В.**[3)], **Варчук Д.А.** [4)], **Ділай П.Р.**[5)]
Західноукраїнський національний університет
[1)]магістрант;[2]аспірант;[3-5)]магістрант

І. Постановка проблеми

Застосовані в сучасній освіті форми навчання, будь то очна або дистанційна, орієнтовані насамперед на "усередненого" студента і практично не враховують індивідуальні особливості та потреби студентів: їх рівень знань, здатність до навчання, мотивацію, особисті переваги і т.д. При цьому ефективність навчального процесу визначається також технологією розробки та застосування навчально-методичних матеріалів, за якими учні освоюють нові знання та навички. Вирішенням проблеми реалізації індивідуального навчання є адаптивне навчання, спрямоване на підвищення ефективності придбання нових компетенцій за допомогою адаптивних навчальних матеріалів (у загальному випадку - адаптивного контенту)[1].

Сучасна інтерпретація поняття "адаптивне навчання" передбачає реалізацію навчального процесу на основі застосування електронних систем навчання (цифрових навчальних платформ, систем дистанційного навчання), зміст навчального контенту яких підбирається в автоматичному режимі таким чином, щоб врахувати характеристики та здібності конкретного учня.

Реалізація інструментальної навчальної системи у технології адаптивного навчання ґрунтується на застосуванні моделей учня, моделі навчального контенту, і навіть моделі адаптації [2]. Розробка нових або вибір існуючих моделей є питанням наукового дослідження через їх безпосередній вплив на функціональні можливості системи та адекватність одержуваного результату.

ІІ. Мета роботи

Метою дослідження є розробка математичного та програмного забезпечення для проведення адаптивного навчання студентів.

ІІІ. Модель забування інформації

Для екстраполяції рівня залишкових знань на момент закінчення курсу використовувалася модель, що базується на швидкості забування інформації. Інші обґрунтовані моделі, придатні для чисельного прогнозування рівня знань учнів у майбутні періоди часу на основі їх попередньої історії навчання, практично не існують.

Використання байєсівських мереж не призводить до значного підвищення точності прогнозування (за межами завдань адаптивного тестування), але вимагає значних обчислювальних ресурсів. Технологія машинного навчання, як, наприклад, Snappet, забезпечує хорошу швидкість прийняття рішень, але для досягнення достатньої точності прогнозування у навчальному курсі з великою кількістю модулів потрібна база даних з великою кількістю пройдених траєкторій навчання. Тому на початковому етапі впровадження навчального курсу рекомендується використовувати статистичні моделі (наприклад, байєсівські мережі довіри) або моделі, що базуються на швидкості забування інформації.

Основна ідея адаптивного навчання полягає в побудові оптимальної траєкторії вивчення модулів курсу. Розглянемо докладніше, у чому складається це завдання.

Нехай задано множину компетенцій K, необхідних для освоєння студентами, а також множина модулів M, що реалізують K при цьому кожна компетенція K_j може реалізовувати більше одного модуля.

У кожного модуля є принаймні одна вихідна компетенція та нуль або більше вхідних. За рахунок того, що одна й та ж компетенція для одного модуля є вхідною, а для іншого - вихідною, множина модулів та компетенцій утворюють граф $G=(V,C)$, що складається з вершин модулів та вершин компетенцій $V=\{M,K\}$, і множини зв'язків C між ними. Приклад такого графа представлено на рисунку 1.

Нехай поставлений час t, відведений на навчання. При освоєнні модуля M_i студент набуває знання з відповідного цього модуля вихідний компетенції із набору K, що вимірюється за умовною

шкалою від 0 до 100%. Нехай R – сумарне значення рівнів залишкових знань усіх компетенцій на даний момент закінчення навчання.

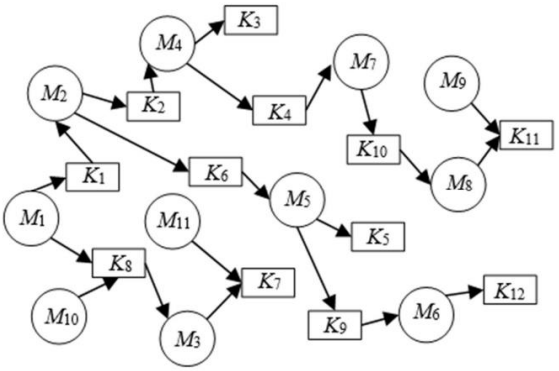


Рисунок 1 – Приклад графа навчальних модулів

Завдання системи – знайти такий шлях на графі , щоб на момент закінчення курсу R мало максимальне значення, тобто

$$R(P,t) \to \max \qquad (1)$$

Обмеження задачі (1):

1. Повинна виконуватись умова несуперечності проходження модулів. Наприклад, на рисунку 2 допустимим фрагментом шляху буде послідовність $M_1 \to M_2$, але не навпаки. При цьому можливою і допустимою також є послідовність $M_1 \to M_{11} \to M_2$, незважаючи на те, що прямий зв'язок між M_1 та M_{11} не заданий.

2. При завершенні освоєння студентом модуля та переході до наступного, до щойно придбаних компетенцій застосовується закономірність $R(t) = \begin{cases} 1, & t < 1; \\ \min\left(\frac{k}{\lg t + c}, 1\right), & t \geq 1 \end{cases}$, тобто з часом рівень знань починає спадати.

Відповідно до концепції ітеративного навчання, той самий модуль може брати участь у P більш ніж один раз. Оскільки модуль характеризується часом, відведеним на його вивчення, то порядок модулів різної тривалості (враховуючи умови несуперечності проходження), а також повторне включення P раніше вивчених модулів надають прямий вплив на R до моменту закінчення навчання.

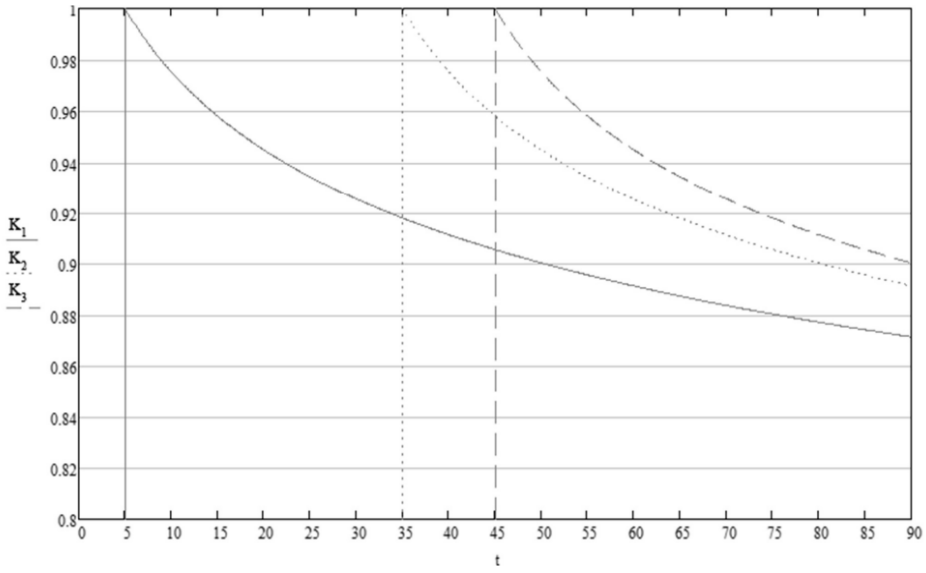


Рисунок 2– Криві забування компетенцій для послідовності $M_1 \to M_2 \to M_3$

Висновок

Для реалізації адаптивного навчання було отримано метод, який застосовує модель навігації "повне керівництво". На кожному кроці студенту пропонується до освоєння тільки один модуль. Після проходження тестування, спрямованого на вимірювання рівня знань з щойно вивченою компетенцією, алгоритм здійснює нове обчислення траєкторії P з обліком поточного стану моделі студента S, що дозволяє кожному просуватися індивідуальною траєкторією.

Список використаних джерел

1. Pardamean, B.; Suparyanto, T.; Cenggoro, T.W.; Sudigyo, D.; Anugrahana, A. AI-Based Learning Style Prediction in Online Learning for Primary Education. IEEE Access 2022, 10, 35725–35735.
2. Wang, S.; Wu, H.; Kim, J.H.; Andersen, E. Adaptive Learning Material Recommendation in Online Language Education. In Artificial Intelligence in Education. AIED 2019. Lecture Notes in Computer Science; Isotani, S., Millán, E., Ogan, A., Hastings, P., McLaren, B., Luckin, R., Eds.; Springer: Cham, Switzerland, 2019; Volume 11626

УДК 004.738.5

ІНФОРМАЦІЙНО-ОЗНАКОВА МОДЕЛЬ ШКІДЛИВОЇ ІНФОРМАЦІЇ В СОЦІАЛЬНІЙ МЕРЕЖІ

Судейченко Д.В.[1)], **Даньків А.В.**[2)], **Биц С.С.**[3)], **Могильська І.М.**[4)], **Олійник А.П.**[5)]
Західноукраїнський національний університет
[1)]магістрант;[2]аспірант;[3)]магістрант;[4)]студент;[5)]аспірант;

І. Постановка проблеми

Глибина проникнення соціальних мереж у повсякденне життя значна, і їхні переваги полягають у можливості учасників комунікації оперативно висловлювати свою думку великій групі людей та публікувати медіа файли. Сьогодні соціальні мережі вже не лише засіб спілкування, але й інструмент розповсюдження інформації. Проблема шкідливої інформації стала очевидною проблемою інформаційної безпеки сучасного суспільства. Терористичні та злочинні угруповання все частіше використовують засоби інформаційного впливу для розширення своєї сфери впливу та залучення нових адептів через соціальні мережі. Тому моніторинг, аналіз та активна протидія шкідливій інформації в соціальних мережах стає надзвичайно важливим для забезпечення інформаційної безпеки держави [1-3].

Проте в наш час існує недостатньо науково-технічних рішень для протидії шкідливій інформації. Існуючі засоби виявлення та протидії шкідливій інформації в соціальних мережах не відповідають вимогам до швидкості, повноти, точності та адекватності прийнятих рішень. Це зумовлено кількома причинами, включаючи розділеність системи на два не пов'язаних модулі - моніторинг та протидію, складну структуру соціальних мереж та потребу в обробці великих обсягів повідомлень в реальному часі [4-5].

Отже, основна складність протидії шкідливій інформації в соціальних мережах випливає з сучасних тенденцій розвитку інформаційної сфери, таких як збільшення обсягу та швидкості поширення шкідливої інформації. Це ставить перед науковцями та фахівцями з інформаційної безпеки виклик з підвищення ефективності протидії шкідливій інформації у соціальних мережах, зокрема за рахунок оперативності та обґрунтованості.

ІІ. Мета роботи

Метою дослідження є розробка інформаційно-ознакової моделі шкідливої інформації в соціальній мережі.

ІІІ. Інформаційно-ознакова модель шкідливої інформації в соціальній мережі

Соціальні мережі *(SN)* – сукупність взаємозалежних вузлів, якими є аканти спільноти, сторінки, пости, вкладення тощо, а зв'язки між об'єктами – це однорівневі відносини (складаються "у друзях", перебувають у співтоваристві, тощо) та відносини вкладеності (стіна містить пост, сторінка облікового запису містить посилання на пост і т.п.). Соціальні мережі асоціюються із графами, де частина об'єктів інформаційні, і є вершиною графа, а частина – зв'язки між такими об'єктами є ребра між вершинами [1-2].

У процесі аналізу основних структур даних соціальних мереж можна виділити загальні атрибути, які можуть бути застосовні для опису взаємозв'язку між джерелом, реципієнтом та шкідливою інформацією. Нехай у момент приєднання до соціальної мережі суб'єкт проходить процес реєстрації та створює мережевий профіль - обліковий запис (*account*), який належить множині *ACCOUNT*.

У процесі реєстрації облікового запису суб'єкт отримує унікальний ідентифікатор (*id*), який належить множині *ID*. Суб'єкт завжди до нього прив'язаний, хоча знаходиться на межі між віртуальним та реальним світом. Суб'єкт може не додавати про себе жодної інформації після реєстрації, проте це не заважає йому створювати та передавати інформацію в соціальній мережі [3].

Далі суб'єкт заповнює обліковий запис шляхом внесення даних про себе у мережному профілі, і тоді він заповнює свою власну сторінку (*pageac)*. Сторінка облікового запису належить множині PAGEac. Суб'єкт може створити спільноту, і в цей момент спільнота отримує свою унікальну адресу та профіль (*group*). Співтовариства в соціальних мережах утворюють множину *GROUP*, після заповнення профілю спільноти формується його сторінка ($pageg$). Сторінки груп належать множині *PAGEg*. При цьому

$$PAGE = PAGEac \cup PAGEg \quad (1)$$

У процесі аналізу будь-якого повідомлення в соціальних мережах можна визначити сторінку, де воно опубліковано. Сторінка, на якій опубліковано повідомлення із шкідливою інформацією – це джерело *(source)*. Усі джерела в соціальних мережах утворюють множину *SOURCE*. Поширення шкідливої інформації в соціальних мережах можливе шляхом публікації повідомлення (*message*). Всі повідомлення соціальних мереж утворюють множину *MESSAGE*. Повідомлення – це будь-який пост на стіні акаунта, на стіні групи, запис у коментарях тощо. Суб'єкт створює повідомлення та публікує його у відкритому доступі від імені свого облікового запису або від імені групи; він (суб'єкт) є автором (*author*). Усі пишучі суб'єкти соціальних мереж утворюють множину *AUTHOR*.

Нехай *MIO* – це інформаційний об'єкт (шкідливий інформаційний об'єкт), який містить ознаки, що дозволяють прийняти рішення про те, що інформація завдає шкоди суспільству, особистості, державі чи бізнесу. При цьому ознаки T (токени) інформаційної загрози встановлює експерт, залежно від умов. Наприклад, коли використовується система батьківського контролю, батько сам вибирає обмеження для дитини. Якщо ж представник бізнесу зацікавлений у захисті конфіденційної інформації, тоді він сам задає їх. Отже, теоретико-множинна модель шкідливої інформації у соціальній мережі включає такі базові елементи:

- *IO*- інформаційний об'єкт,
- *T*- інформаційна загроза,
- *MIO* – шкідливий інформаційний об'єкт,
- *Token*– ознака інформаційної загрози, що міститься у шкідливому інформаційному об'єкті,
- Feature – ознака наявності інформаційного об'єкта, зв'язок між об'єктами.

Теоретико-множинна модель формально представлена наступним чином:

$$IO = \{io\};\ MIO = \{io\};\ MIO_i = \{io\}$$

$$MIO \subset IO;\ \forall io \in MIO : io \in IO$$

$$MIO_i \subseteq MIO;\ \forall io \in MIO_i;\ io \in MIO$$

$$Token \subset T;\ Token = \{t\}$$

$$CheckFeature(io, t) = \{True; False\}$$

$$io \in MIO_i \Leftrightarrow \exists Token_m io_i : CheakFeature(uo, t) = True.$$

де *IO* – множина інформаційних об'єктів, io_i – один інформаційний об'єкт, *T*- множина всіх можливих ознак інформаційної загрози, t_n – одна ознака інформаційної загрози, *MIO* – множина шкідливих інформаційних об'єктів, *MIOi* - окремий клас шкідливої інформації, $Token_m$ - множина ознак характеризуючих *MIO*.

Таким чином, для протидії шкідливій інформації необхідно задати набір ознак, характерних для інформаційної загрози. Відмінною особливістю є те, що модель припускає наявність дискретних ознак у наборі ознак, таких як: дата створення інформаційного об'єкта, зв'язок інформаційного об'єкта з іншими об'єктами у соціальній мережі, частота повторення ознаки та ін.

Протидія шкідливому інформаційному об'єкту може здійснюватися лише на рівні повідомлень чи рівні джерел. Необхідно виділити такі погрози та їх інформаційні ознаки повідомлення у соціальній мережі, що характеризують його як шкідливе. Загалом під інформаційно-ознаковою моделлю [3,4] у дослідженні розуміється впорядкована сукупність відомостей про зв'язки повідомлень та їх інформаційних ознак із змістом повідомлень. У свою чергу, під інформаційними ознаками повідомлень розуміється окремі властивості повідомлень, а точніше їх зміст:

- інформаційна загроза – задається оператором системи;
- шкідлива інформація в соціальній мережі – задається оператором шляхом формування набору ключових слів;
- інформаційні ознаки, що формують множину усіх можливих ознак t.

На рисунку 1 зображено співвідношення різних рівнів інформаційно-ознакової моделі шкідливої інформації. Показано, що автор формує повідомлення та розміщує його в джерелі розповсюдження, на сторінці або обліковому записі, або групі. Повідомлення можуть містити чи не містити ознаки шкідливої інформації. Ознаки формують рівень інформаційних загроз.

Таким чином, зібравши інформацію на сторінці будь-якого джерела, можливо визначити, які з цих повідомлень належать до шкідливих. З урахуванням виявлених загроз та їх кількості може бути ухвалено рішення про протидію повідомленню або джерелу.

Розроблена інформаційно-ознакова модель шкідливої інформації дозволяє сформувати вихідні дані для протидії шкідливої інформації.

Розроблений комплекс моделей складається з моделі соціальної мережі, моделі джерела інформації, моделі шкідливої інформації та інформаційно-ознакової моделі шкідливої інформації Кожна з моделей, що входять до комплексу, містить унікальні, запропоновані автором комплексу множини, атрибути та відношення між елементами. Одночасно зцим комплекс моделей дозволяє сформувати вимоги до алгоритмів аналізу та оцінки джерел та вибору контрзаходів.

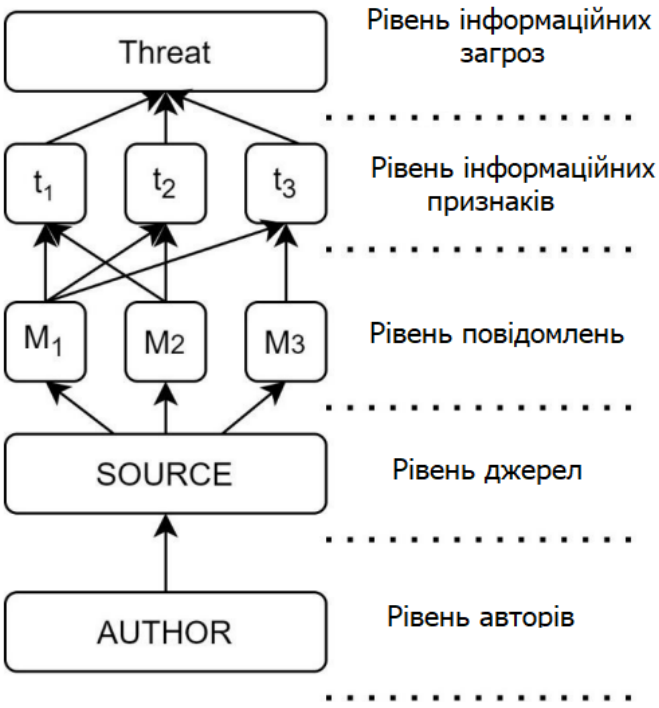


Рисунок 1 –Графічне представлення інформаційно-ознакової моделі шкідливої інформації

Висновок

В рамках цього дослідження було розроблено модель соціальної мережі, що включає повідомлення, джерела та зв'язки між ними, яка відрізняється наявністю нових структурних елементів та зв'язків. Також розроблено модель джерела, в якій вперше враховуються такі параметри, як індекс активності, потенціал, індекс впливовості та індекс перегляду. Розроблено інформаційно-ознакову модель шкідливої інформації, що складається з взаємопов'язаних об'єктів та ознак шкідливої інформації, разом формують шкідливо-інформаційні об'єкти.

Список використаних джерел

1. Войтович О. П. Модель та засіб для виявлення фейкових облікових записів у соціальних мережах / Войтович О. П., Дудатьєв А. В., Головенько В. О. // Вчені записки таврійського національного університету ім. В.І. Вернадського. Серія: Технічні науки. – Частина 1. – 2018. – № 1. –Том 29 (68). – С. 112-119.
2. Monti, F.; Frasca, F.; Eynard, D.; Mannion, D.; Bronstein, M.M. Fake news detection on social media using geometric deep learning. arXiv 2019, arXiv:1902.06673, 2019.
3. Silva, A.; Han, Y.; Luo, L.; Karunasekera, S.; Leckie, C. Propagation2Vec: Embedding partial propagation networks for explainable fake news early detection. Inf. Process. Manag. 2021, 58, 102618.
4. Barnabò, G.; Siciliano, F.; Castillo, C.; Leonardi, S.; Nakov, P.; Martino, G.D.S.; Silvestri, F. Deep active learning for misinformation detection using geometric deep learning. Online Soc. Netw. Media 2023, 33
5. Shu, K.; Mahudeswaran, D.; Wang, S.; Liu, H. Hierarchical propagation networks for fake news detection: Investigation and exploitation. In Proceedings of the International AAAI Conference on Web and Social Media, Limassol, Cyprus, 5–8 June 2020; Volume 14, pp. 626–637.

УДК 004.43

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ІНТЕЛЕКТУАЛЬНОЇ ПІДТРИМКИ АНАЛІЗУ ТА ТЕСТУВАННЯ ПРОГРАМ

Мархоцький Н.Ю.[1)], **Олійник А.П.**[2)], **Судейченко Д.В.**[3)], **Костик Б.П.**[4)]
Західноукраїнський національний університет
[1)]магістрант;[2]аспірант;[3)]магістрант;[4)]аспірант

І. Постановка проблеми

Тестування відносять до динамічних методів верифікації, який дозволяє перевірити програму у процесі її виконання. Підходи до тестування та розроблення тестів використовуються для чіткого визначення мети тестування програми чи її компонента/модуля. Тестові набори для перевірки програми формуються на основі застосування методів верифікації або надаються після використання альтернативного підходу (білої скриньки)[1,2].

У той же час поточні наукові дослідження показують, що існують підходи, розробка та застосування яких може значно поліпшити якість тестів, що генеруються, виражається в ступені покриття ними коду, що тестується. Особливий інтерес представляють підходи на основі аналізу коду програм (тестування білої скриньки), оскільки витягнута з тексту програми інформація може сприяти подальшому збільшенню показника якості покриття коду [2,3].

Тому наразі більш реалістичним і ефективним для практичного використання у компаніях, що займаються виробництвом ПЗ, є динамічний підхід, заснований на фактичному виконанні програми при деяких значеннях вхідних змінних та наступному аналізі потоків даних [3].

Еволюційна основа, що лежить в основі генетичного алгоритму (ГА), використовує велику кількість випадкових тестових даних, згенерованих на початковому етапі, після чого проводиться послідовна "еволюція" даних з метою поліпшення якості покриття коду, що тестується. У зв'язку з вищезазначеним виникає припущення про можливість адаптації генетичного алгоритму для реалізації ідеї еволюційного покращення тестових даних з точки зору максимізації покриття ними коду, що тестується. У зв'язку з вищезазначеним виникає припущення про можливість адаптації генетичного алгоритму для реалізації ідеї еволюційного покращення тестових даних з точки зору максимізації покриття тестованого коду.

ІІ. Мета роботи

Метою дослідження є розробка математичного та програмного забезпечення для інтелектуальної підтримки аналізу та тестування програм.

ІІІ. Застосування основних еволюційних операцій в задачах генерації тестових даних

Першим етапом роботи ГА, як було сказано вище, є формування початкової популяції. Стандартно у ГА початкова популяція визначається випадковим чином, тобто перша популяція наповнюється хромосомами з рівномірно розподіленими випадковими значеннями тестових змінних в межах заздалегідь заданих областей визначення. Незважаючи на те, що існують і інші методи ініціалізації, саме випадковий метод формування початкової популяції використовується у цій роботі, оскільки він дозволяє забезпечити достатню різноманітність отримуваних тестових варіантів.

Після того, як було сформовано початкову популяцію, починається послідовне виконання основних еволюційних операцій відповідно до алгоритму ГА. Далі розглянемо особливості їх застосування у завданні генерації тестових даних, використаних у цій роботі.

Відбір (селекція). Хромосоми у популяції сортуються відповідно до значення функції пристосованості. Сортування дозволяє розділити хромосоми на найкращі (з більшим значенням функції) та гірші (з меншим значенням функції), що є необхідним критерієм для відбору.

Перед етапом відбору частина найкращих хромосом переноситься до нового покоління без змін, тим самим формуючи пул найкращих хромосом. Інші хромосоми нового покоління будуть отримані в результаті схрещування. Саме для визначення пулу придатних для схрещування хромосом проводиться операція відбору. Пул кращих хромосом (P_{best}) та пул хромосом для схрещування (P_{cross}) можуть як збігатися, так і не збігатися.

Методи відбору мають свої переваги та недоліки. Найбільший інтерес представляють методи турнірного відбору та метод випадкового визначення пар. Перший метод дозволяє розділити популяцію на окремі підгрупи, з яких відбувається відбір батьків, а другий – досягти більшого

розмаїття, що у завданні по генерації тестових даних відіграє важливу роль. Тому у цій роботі пропонується гібридний метод відбору на основі цих методів.

Перш ніж приступати безпосередньо до відбору, необхідно визначити, яка частина нового покоління буде отримана перенесенням найкращих хромосом, а яка - за допомогою схрещування. Емпірично для завдання генерації тестових даних було визначено такі параметри. В якості кращих хромосом для перенесення в нове покоління без змін вибираються 20% хромосом з найбільшою функцією пристосованості, що становлять пул кращих P_{best}. Інші 80% хромосом нового покоління будуть відібрані з використанням гібридного методу:

- 50% хромосом для схрещування вибираються тільки з пулу P_{best};
- 50% хромосом відбиратимуться з популяції загалом, тобто значення функції пристосованості не враховується.

Схрещування серед усіх хромосом необхідне для досягнення більшого різноманіття популяції. Таким чином, гібридний метод дозволяє краще досліджувати тестовий код та відбирати хромосоми з метою досягнення більшої різноманітності. Після відбору пари батьківських хромосом вони формують одного нащадка з використанням еволюційної операції схрещування.

Схрещування. Схрещування дозволяє отримувати нові тестові набори перемішуванням значень вже існуючих. У термінах генетичного алгоритму, дві батьківські хромосоми обмінюються генами для отримання нащадка. Основна мета використання схрещування полягає в ітераційному покращенні покоління послідовним схрещуванням найкращих хромосом для отримання найкращих рішень. У цій роботі схрещування використовується також для збільшення різноманітності популяції гібридним методом відбору.

Існують різні методи схрещування. При їх описі, для зручності, визначимо одну з батьківських хромосом як материнську (x_{mother}), а другу – як батьківську (x_{father}).Схрещування за однією позицією. Для отримання нащадків у батьківських хромосом вибирається одна позиція, за якою відбувається обмін генами. Стандартно використовується одна статична позиція для схрещування, більш просунутим методом є визначення позиції випадковим чином. Графічно схрещування двох хромосом з 6 вхідними змінними та позицією схрещування $r = 4$ показано на рисунку 1.

Застосовуючи схрещування по одній позиції, можна отримати дві нові хромосоми, але вони не дуже відрізняються від батьківських. У разі якщо одна з частин хромосоми робить набагато більший вплив на результат, то нові тестові набори, отримані таким чином, не будуть вносити жодного різноманітності в популяцію, що негативно позначається на роботі алгоритму загалом. Одноточкове схрещування є простим для розуміння та реалізації, але має надто суттєві недоліки.

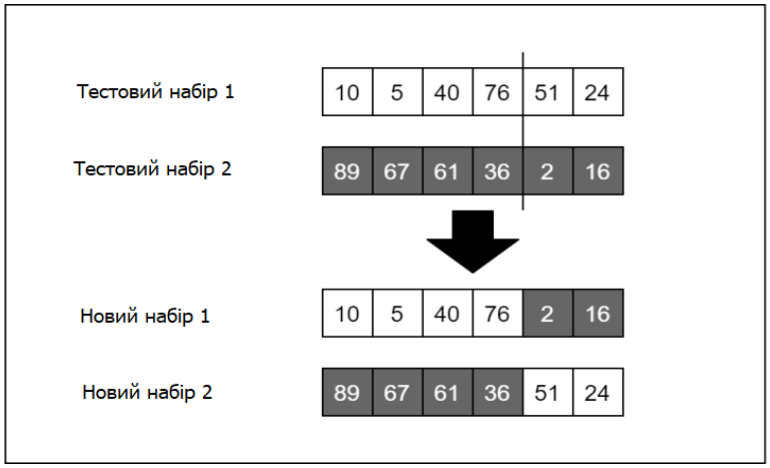


Рисунок 1– Приклад одноточкового схрещування тестових наборів

Так, збільшення кількості позицій для схрещування може сприяти отриманню більш відмінних тестових наборів, оскільки дозволяє більш глибоко і різноманітно комбінувати гени в батьківських хромосомах. Уніфіковане схрещування є одним з типів схрещування, яке відрізняється від інших у підході до отримання нащадків. У цьому методі кількість генів та кількість позицій для схрещування збігаються. Кожен ген нащадка випадково успадковується від одного з батьків. Оскільки батьки у генетичному алгоритмі рівнозначні, кожен ген нащадка має однакову ймовірність вибору від одного або іншого батька. Це призводить до того, що кожен ген нащадка може успадкувати значення або від одного, або від іншого батька з ймовірністю 0.5.

Щоб нівелювати недоліки безперервного ГА, пропонується використовувати еволюційну операцію змішування, яка дозволяє вводити у популяцію нові значення в залежності від значень генів батьків. Як було сказано раніше, змішування необхідне для отримання нових значень генів у популяції, що є істотним для генерації нових значень у безперервному варіанті ГА.

Змішування є механізмом, що дозволяє отримати значення генів у нащадків, відмінне від батьків. Можна виділити такі методи змішування:

- Пряме змішування значень. Нове значення гена обирається випадковим чином у проміжку між батьківськими значеннями.

- Середні значення. Значення нового гена отримується усередненням значень батьківських генів. Очевидно, що такий метод погано підходить для збільшення різноманітності, але може використовуватися, наприклад, для пошуку відповідних меж вхідних змінних.

Мутація є еволюційною операцією, яка дозволяє вносити в популяцію нові значення за допомогою випадкової зміни генів. Мутація застосовується тільки до нащадків, тому найкращі тестові набори, що передаються у нове покоління безпосередньо (без операції схрещування), не змінюватимуться. Дуже важливо задати вірну ймовірність мутації, оскільки занадто невеликий шанс може нівелювати переваги мутації в цілому, і алгоритм може не підібрати тестові набори для виходу більш складними шляхами. Занадто високий шанс, навпаки, непередбачувано змінюватиме нащадків, через що нівелюються переваги використання ГА порівняно з випадковим пошуком.

Мутація застосовується до кожного гена окремо. Залежно від реалізації випадкового шансу мутації, вона може бути не застосована до хромосом зовсім або змінити тільки один ген (рисунок 2) або змінити кілька.

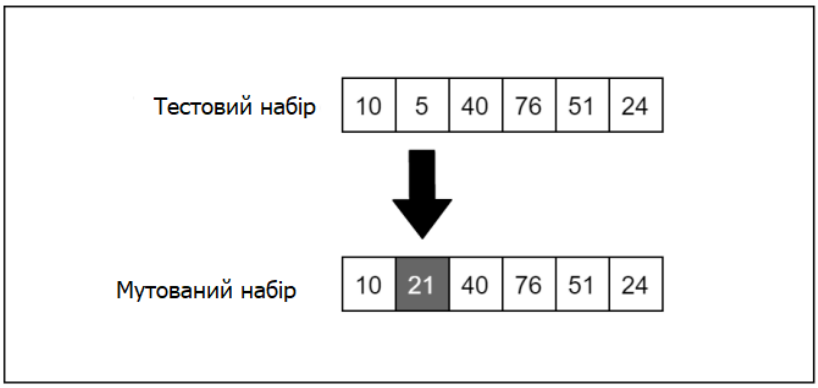


Рисунок 2 – Приклад одноточкової мутації тестового набору, в якомуз начення другої змінної змінилося

Хоча мутація є досить потужним інструментом для забезпечення різноманітності популяції, необхідно користуватися нею обережно. Мутація може як допомогти у знаходженні складніших шляхів, так і ускладнити пошук. Емпірично для завдання генерації тестових даних було визначено оптимальну ймовірність мутації на рівні 5% (0.05), яка, в сукупності з змішуванням, дозволила забезпечити достатню різноманітність популяції. Таким чином, ітераційно виконуючи раніше розглянуті еволюційні операції алгоритм забезпечує пошук тестових даних.

Висновок

Основні поняття генетичного алгоритму, такі як ген, хромосома і популяція, були адаптовані для вирішення задачі генерації тестових даних. Показано застосування безперервного варіанта ГА, визначено поняття невід'ємних хромосом, описано особливості реалізації еволюційних операцій для розв'язання даної задачі. Описано основні етапи генетичного алгоритму. Пропонується новий метод відбору, що ґрунтується на виборі кращих хромосом, випадковому формуванні пар та турнірному відборі. Для забезпечення більшої різноманітності популяції використовується еволюційна операція змішування, яка дозволяє вводити нові значення у популяцію.

Список використаних джерел

1. Jamil, M.A.; Nour, M.K.; Alhindi, A.; Awang Abhubakar, N.S.; Arif, M.; Aljabri, T.F. Towards Software Product Lines Optimization Using Evolutionary Algorithms. Procedia Comput. Sci. 2019, 163, 527–537.
2. Alsewari, A.A.; Zamli, K.Z.; Al-Kazemi, B. Generating t-way test suite in the presence of constraints. J. Eng. Technol. (JET) 2015, 6, 52–66.
3. Малихіна М. П., Частікова В. А., Власов К. А. Дослідження ефективності роботи модифікованого генетичного алгоритму в задачах комбінаторики // Сучасні проблеми науки та освіти. – 2013. – No 3.

УДК 004.04

ПІДХІД ДО АНАЛІЗУ УНІКАЛЬНОСТІ ПРАКТИЧНИХ РОБІТ СТУДЕНТІВ

Крепич С.Я.[1)], **Глухов О.В.**[2)], **Співак І.Я.**[3)]
Західноукраїнський національний університет
[1)] к.т.н., доцент; [2)] магістрант; [3)] к.т.н., доцент

I. Постановка проблеми

В університетському середовищі проведення лабораторних робіт є важливою складовою академічного процесу. Проте, однією з основних проблем, з якою зіштовхуються викладачі, є виявлення плагіату та копіювання студентами результатів робіт. У сучасному освітньому середовищі для перевірки унікальності випускових кваліфікаційних робіт часто використовуються спеціалізовані сервіси, такі як Turnitin, Unicheck та інші. Ці сервіси пропонують платні плагіат-детектори, які автоматично перевіряють текстові документи на унікальність, порівнюючи їх з великою базою даних наукових статей, книг та Інтернет-ресурсів. Проте, вони зазвичай використовуються лише для архівних робіт, таких як випускові кваліфікаційні роботи, і не завжди доступні для перевірки лабораторних або практичних завдань. У той час, як унікальність лабораторних чи практичних робіт зазвичай перевіряється викладачем самостійно, що може бути часо- та працевитратним процесом. Викладачеві доводиться аналізувати та порівнювати кожен студентський проект з іншими текстами, що може бути важко і неефективно, особливо у великих групах. Для вирішення цієї проблеми необхідно розробити ефективний додаток, який забезпечить можливість виявлення збігів у лабораторних роботах та інших завданнях. Для цієї цілі найбільше підходить розробка ПЗ на комп'ютер, на базі операційної системи Windows.

II. Мета роботи

Мета роботи – створення програмного забезпечення, яке дозволить виявляти збіги у лабораторних роботах та інших завданнях, що надасть викладачам засоби для виявлення плагіату та копіювання в академічному середовищі.

III. Огляд методів

Метод TF-IDF (Term Frequency-Inverse Document Frequency): Цей метод використовується для оцінки важливості кожного слова у документі, порівнюючи його частоту в документі з його частотою в корпусі документів. Він може бути використаний для визначення ступеня схожості між документами на основі їхнього змісту.

		m	e	i	l	e	n	s	t	e	i	n
	0	1	2	3	4	5	6	7	8	9	10	11
l	1	1	2	3	3	4	5	6	7	8	9	10
e	2	2	1	2	3	3	4	5	6	7	8	9
v	3	3	2	2	3	4	4	5	6	7	8	9
e	4	4	3	3	3	3	4	5	6	6	7	8
n	5	5	4	4	4	4	3	4	5	6	7	7
s	6	6	5	5	5	5	4	3	4	5	6	7
h	7	7	6	6	6	6	5	4	4	5	6	7
t	8	8	7	7	7	7	6	5	4	5	6	7
e	9	9	8	8	8	7	7	6	5	4	5	6
i	10	10	9	8	9	8	8	7	6	5	4	5
n	11	11	10	9	9	9	8	8	7	6	5	4

Рисунок 1 - Алгоритм Левенштейна

Алгоритм Левенштейна: Цей алгоритм використовується для визначення відстані між двома рядками шляхом підрахунку мінімальної кількості операцій (вставка, видалення, заміна символів), необхідних для перетворення одного рядка на інший. Він може бути використаний для визначення рівня схожості між текстовими документами.

Метод N-грам: Цей метод використовується для аналізу тексту шляхом розбиття його на послідовності з n символів (зазвичай n = 2 або 3) і порівняння цих послідовностей між двома документами. Він може виявляти схожість між документами навіть у випадках, коли слова переставлені або змінено порядок слів.

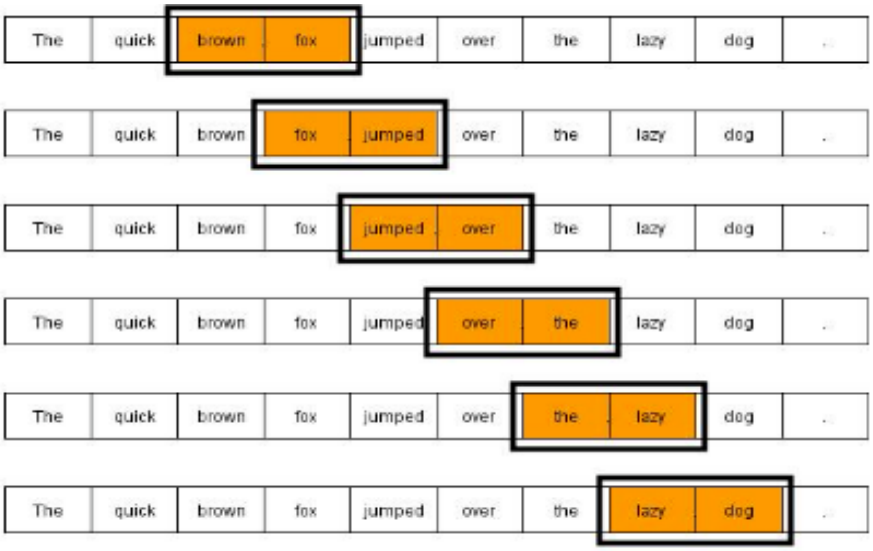


Рисунок 2 - Метод N-грам

Машинне навчання: Методи машинного навчання, такі як нейронні мережі, науково-статистичні методи та класифікаційні алгоритми, можуть бути використані для автоматичного виявлення збігів у лабораторних роботах на основі великої кількості навчальних даних.

У таблиці 1 представлені дані експерименту із

застосування методу N-грам для оцінки подібності текстів між собою.

Таблиця 1

Порівняння застосування методу N-грам до текстів різної схожості

Текст№1	Текст№2	Збіжність тексту
Мовлення як вид людської діяльності завжди зорієнтоване на виконання певного комунікативного завдання. Висловлюючи думки і почуття, людина ставить конкретну мету – щось повідомити, про щось переконати тощо. Існує багато визначень тексту. Наведемо окремі з них.	Мовлення як вид людської діяльності завжди зорієнтоване на виконання певного комунікативного завдання. Висловлюючи думки і почуття, людина ставить конкретну мету – щось повідомити, про щось переконати тощо. Існує багато визначень тексту. Наведемо окремі з них.	98.9%
«Текст – це витвір мовленнєвого процесу, що відзначається завершеністю, об'єктивований у вигляді письмового документа, літературно опрацьований відповідно до типу документа.	«Текст – певна, з функціонально-смислового погляду упорядкована, група речень або їх аналогів, які являють собою, завдяки семантичним і функціональним взаємовідношенням елементів, завершену смислову єдність»	2.44%
Мовлення як вид людської діяльності завжди призначена на виконання якось комунікативного завдання. Висловлюючи думки і почуття, людина ставить певну мету – щось повідомити, про щось переконати . Існує багато визначень тексту. Наведемо окремі з них.	Мовлення як вид людської діяльності завжди зорієнтоване на виконання певного комунікативного завдання. Висловлюючи думки і почуття, людина ставить конкретну мету – щось повідомити, про щось переконати тощо. Існує багато визначень тексту. Наведемо окремі з них.	73.2%

Висновок

У роботі було проведено дослідження існуючих методів оцінювання унікальності текстів. Метод TF-IDF враховує важливість слова в документі та базі, але не враховує порядок слів; алгоритм Левенштейна ефективний для визначення відстані між текстовими рядками, однак не враховує семантичні зв'язки; метод N-грам добре виявляє схожість певним чином впорядкованих слів. Проведений експеримент показав адекватність методу N-грам для застосування при перевірці практичних робіт студентів на унікальність змісту.

Список використаних джерел

1. ntchosting https://www.ntchosting.com/encyclopedia/frameworks/symfony/ (дата звернення: 10.04.2024)
2. Turnitin https://www.turnitin.com/infographics/the-plagiarism-spectrum (дата звернення: 15.04.2024)
3. Herrera F. Multilabel classification: problem analysis, metrics and techniques / F. Herrera, F. Charte, A. J. Rivera, M. J. del Jesus // Springer International Publishing Switzerland, 2016. — P. 65-78.
4. Taylor J. An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements / J. Taylor. — University Science Books, 1999. — P. 128-129.

УДК 004.896.4

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ВИДОБУВАННЯ СТРУКТУРОВАНОЇ ІНФОРМАЦІЇ ПРО ТОВАР З ТЕКСТІВ КОРИСТУВАЧІВ

Хаєцький М.В.[1)], **Микитюк В.А.**[2)], **Ділай П.Р.**[3)], **Костик Б.П.**[4)], **Поворозник Б.С.**[5)]
Західноукраїнський національний університет
[1)]магістрант;[2]аспірант;[3)]магістрант;[4)]аспірант;[5)]магістрант

І. Постановка проблеми

Виявлення ставлення користувачів до товарів і послуг, що купуються, є важливою частиною роботи маркетингового відділу будь-якого бізнесу. Спираючись на отриману інформацію, компанії можуть керувати асортиментом продукції, враховувати переваги окремих соціальних груп при формуванні рекламних повідомлень, діагностувати проблеми, що виникають при експлуатації, і вчасно реагувати на них, використовувати негативний зворотний зв'язок для формування бачення майбутніх продуктових ліній. Доступним джерелом великої кількості вихідних даних для подібного аналізу є численні сайти, де споживачі можуть обговорювати товари, висловлюючи свою думку про різні аспекти товару[1,2].

У цій роботі пропонується витягувати з текстів відгуків окремі оціночні висловлювання покупців, які містили б у собі основний зміст повідомлення, у формі, корисній при вирішенні виникаючих на етапі супроводу продукту завдань.

ІІ. Мета роботи

Метою дослідження є розробка математичного та програмного забезпечення для видобування структурованої інформації про товар з текстів користувачів.

ІІІ. Математичне забезпечення для вилучення та аналізу думок користувачів про споживчі властивості товарів

Дослідження текстів відгуків про товари показало, що найчастіше у своїх відгуках користувачі згадують три джерела проблем: логістичний фактор (швидкість доставки, якість упаковки, стан товару), якість товарів, досвід спілкування з продавцем. Часто спостережувану поведінку можна пояснити тим, що споживачі з обережністю ставляться до покупок у даному магазині і частіше вибирають дешеві товари без високих вимог до якості, турбуючись переважно про терміни та якість доставки. Опис якості, як правило, досить небагатослівний і обмежується загальною оцінкою товару. Це дозволяє зробити висновок про те, що аналіз буде корисним маркетологам для покращення якості до ставки та пошуку шляхів підвищення довіри у користувачів. Можна сказати, що у майбутньому ситуація почне змінюватися, оскільки інтернет-магазини активно покращують логістичний аспект свого бізнесу. У міру того як проблеми із доставкою стануть менше, користувачі почнуть більше уваги приділяти безпосередньо якості товару [3,4].

Для об'єднання множини типів думок користувачів у групи з метою виявлення найбільш важливих напрямів розвитку продукту та супровідних сервісів, в які потрібно вкласти додаткові ресурси, запропоновано оригінальний класифікатор, заснований на елементах універсальної моделі діяльності: суб'єкт діяльності, об'єкт на який спрямована діяльність, засоби використовувані у процесі діяльності, взаємозв'язку між елементами діяльності. Стосовно особливостей розв'язуваного завдання суб'єктом діяльності є продавець, об'єктом діяльності – продукт, що підлягає продажу.

До основних засобів можна віднести сайт інтернет-магазину, логістичні служби, відділ маркетингу. Виходячи із запропонованих елементів декомпозиції, пропонується асоціювати з кожним її елементом типи оціночних висловлювань, які користувачі можуть використовувати для висловлювання своєї думки під час написання текстів відгуків. Вибір типів думок здійснювався у відповідності з критерієм інформативності/значимості висловлювання для вирішення основних завдань, що виникають на етапах експлуатації та супроводу продуктів [3,4].

Споживчі властивості товару: думки про товар в цілому та його експлуатаційні характеристики, такі як якість виконання, виконання заявлених функцій, зовнішній вигляд товару.

Споживчі властивості товарів конкурентів: думки цілком аналогічні класу "Споживчі властивості товару", однак вони стосуються характеристик товарів конкурентів, висловлених споживачами у відгуках.

Якість обслуговування: думка споживачів про те, як продавець враховує їхні побажання, зокрема щодо конкретних товарів. У цю категорію також можна віднести згадки про готовність продавця робити спеціальні пропозиції, такі як знижки, безкоштовна доставка, подарунки.

Здатність вирішувати конфліктні ситуації: оцінка споживачами того, як продавець поводиться у разі конфлікту інтересів, такого як виявлення дефектів, проблеми з доставкою товару тощо.

Гарантійне обслуговування: оцінка справедливості гарантійних термінів та умов обслуговування, якість та оперативність ремонту товарів.

Оплата: форма та якість процедури оплати товарів та/або послуг.

Доставка: швидкість відправлення товару, якість упаковки товару, цілісність товару після прибуття. До цієї групи включаються лише ті аспекти доставки, на які може вплинути продавець.

На основі проведеного аналізу текстів відгуків про продукти користувачів інтернет-магазину для організації розмітки даних та проведення експериментального дослідження було обрано типи думок і об'єднані в наступні класи:

- товар (споживчі якості товару);
- продавець (загальна якість обслуговування, здатність вирішувати конфліктні ситуації);
- доставка (швидкість та якість доставки).

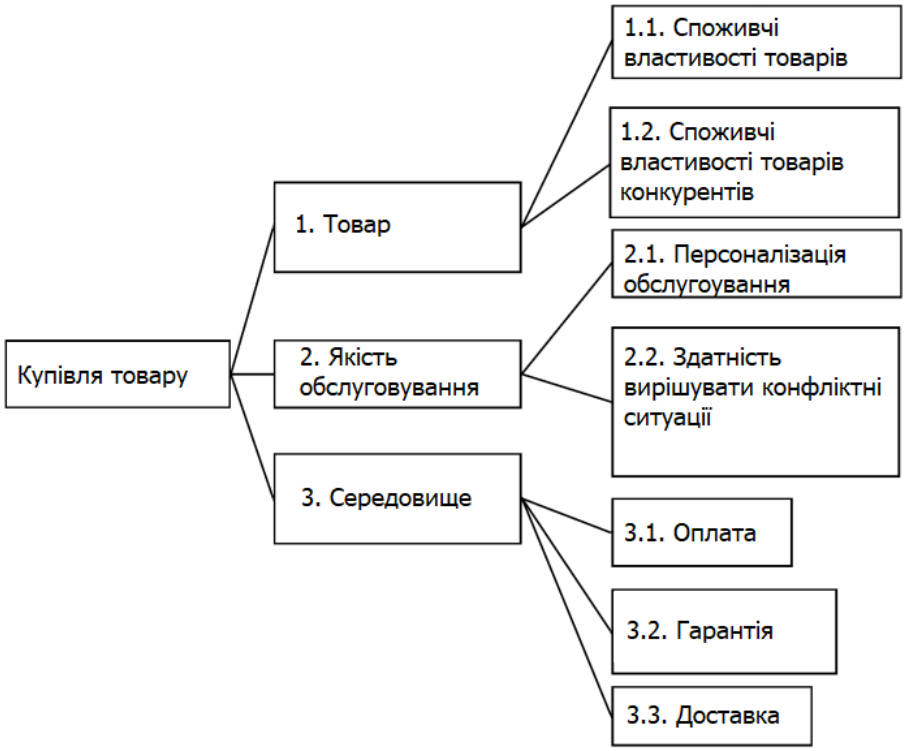


Рисунок 1 –Класифікатор типів думок користувачів

З урахуванням вищезазначеного завдання вилучення та аналізу думок користувачів про товар може бути подане в наступному вигляді. Нехай задано множину текстів відгуків покупців T, що містять думки з множини O про набір продуктів P. Думкою називатимемо четвірку (Аспект, Опис, Тональність, Тип). Аспектом називатимемо послідовність слів, що позначає важливу характеристику або згадку про продукт. Описом будемо називати послідовність слів, що містить висловлену користувачем думку про певний аспект.

Тональність буде визначатися категоріальною шкалою з трьох рівнів: позитивна (+), нейтральна (0), негативна (-). Множина типів думок буде подана відповідно до введеного ієрархічного класифікатора, проте для формування навчального набору даних та практичної апробації будемо використовувати узагальнені класи: {Товар, Продавець, Доставка}.

Маючи вихідні множини T, P, O і задану структуру класифікатора думок, необхідно витягти з нових текстів відгуків T' множини думок користувачів O' про новий набір товарів P'. Для вирішення цього завдання потрібно адаптувати нейромережу у модель шляхом визначення складу множин $Lbl, A, V(a)$ згідно смислового наповнення, введеного вище поняття думки.

$$\begin{aligned} &Lbl = \{Аспект, Опис\}, \\ &A = \{Тональність, Мета\}, \\ &V(Тональність) = \{+, 0, -\}, \\ &V(Мета) = \{Товар, Продавець, Доставка\}. \end{aligned} \tag{1}$$

У функцію $A(Ct)$ потрібно додати нову умову, що дозволяє будувати зв'язок тільки в тих випадках, коли вона поєднує аспект та опис. Виходячи з даної формалізації, множину переходів визначено таким чином:

$$
\begin{aligned}
Y = \{&Shift, Start(Аспект), Start(Опис),\\
&Add(Аспект), Add(Опис), Link(n_1, n_2), End,\\
&Attr(Тональність, +), Attr(Тональність, 0),\\
&Attr(Тональність, -), Attr(Мета, Товар),\\
&Attr(Мета, Продавець), Attr(Мета, Доставка)\}.
\end{aligned}
\tag{2}
$$

Загальна архітектура нейронної мережі для вилучення думок з текстів з урахуванням заданих $Lbl, A, V(a)$ та $A(Ct)$, отримана із загальної моделі, наведена на рисунку 2.

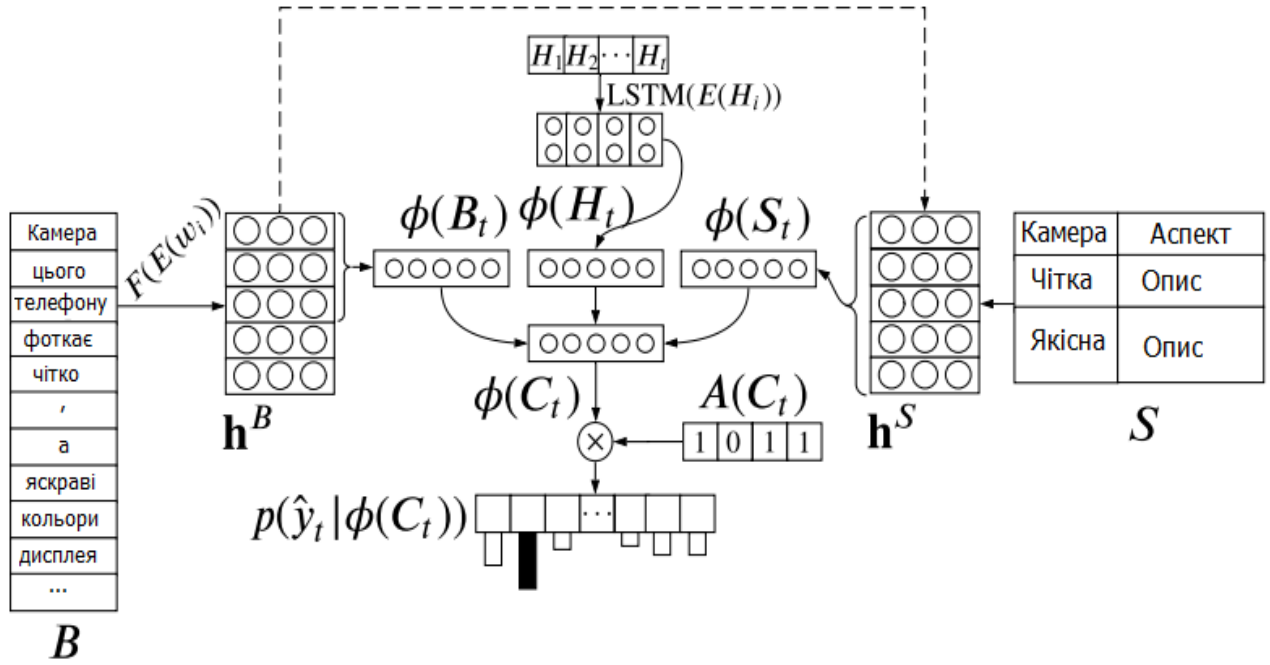


Рисунок 2–Архітектура нейронної мережі для отримання думок

Запропонована загальна нейромережева модель на основі системи переходів для вилучення складових об'єктів та їх атрибутів дозволяє адаптувати її для вилучення об'єктів у різних предметних областях шляхом конкретизації складу множин типів фрагментів та атрибутів.

Висновок

В рамках цього дослідження запропонована модельна основі системи переходів для вилучення складових об'єктів та їх атрибутів. Вона має більш високу порівняно з альтернативами якість визначення значень атрибутів при збільшенні довжини речення, що свідчить про здатність краще враховувати контекстну інформацію при здійсненні передбачень. Використана методика оцінки моделей та отримані результати дозволяють зробити висновок про практичну придатність моделі для вилучення думок про товари з категорій, не включених до навчальної вибірки без додаткового навчання.

Список використаних джерел

1. Зайченко Ю. П. Основи проектування інтелектуальних систем / Ю.П. Зайченко. - Київ: Видавничий дім «Слово», 2004. - 352 с.
2. Терейковський І.А., Заріцький О.В., Терейковська Л.О., Погорелов В.В. Метод розробки архітектури глибокої нейронної мережі, призначеної для розпізнавання комп'ютерних вірусів. Захист інформації. 2018. Т. 20, № 3, С. 188−199.
3. -Zaragoza J., Toselli A. H., Vidal E. Hand written Music Recognition for Mensural Notation: Formulation, Data and Baseline Results. 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, 2017. Pp. 1081–1086.
4. Tapia E., Rojas R. Recognition of on-line handwritten mathematical formulas in the E-chalk system. Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings, 2003. Pp. 980–984.

УДК 004.7

МАТЕМАТИЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СИСТЕМИ СПІЛЬНИХ ПОКУПОК

Порплиця Н. П.[1)], **Цьовка Ю.В.**[2)], **Юшко А.В.**[3)]
Західноукраїнський національний університет
[1)]к.т.н., доцент; [2)] магістрант; [3)]аспірант

І. Постановка проблеми

Системи спільних покупок у сучасному світі відображають постійну еволюцію у сферах технологій та психології людської свідомості. Завдяки стрімкому розвитку цифрових технологій та широкому доступу до Інтернету, способи, якими люди обирають та купують товари та послуги, зазнали значних змін. Тепер споживачі не просто шукають продукти в магазинах або онлайн-платформах, але й активно об'єднуються в групи для спільних покупок.

ІІ. Мета роботи

Метою роботи є розробки ефективного, надійного та зручного користувацького цифрового рішення, що спрощує процес спільного закупівлі товарів для групи людей. Основні аспекти цієї мети включають:

1. Поліпшення доступності та зниження вартості товарів: Завдяки об'єднанню замовлень, система повинна забезпечувати учасникам можливість купувати товари за більш вигідними цінами, використовуючи переваги оптових покупок.
2. Автоматизація процесів: Розробка системи з функціоналом, що автоматизує ключові аспекти спільних покупок, такі як збір замовлень, розрахунок вартості, управління платежами та координація доставки.
3. Забезпечення прозорості та довіри: Створення механізмів, які забезпечують прозорість угод, чесне ціноутворення та розподіл витрат, щоб підвищити довіру між учасниками.
4. Інтуїтивний користувацький інтерфейс: Розробка зручного та інтуїтивно зрозумілого інтерфейсу, який би спрощував процес взаємодії користувачів з системою.
5. Адаптивність та масштабованість: Створення гнучкої системи, яка може адаптуватися до різних розмірів груп та різноманітності товарів, а також масштабуватися для великої кількості користувачів.
6. Інтеграція з іншими сервісами: Інтеграція з платіжними системами, логістичними операторами та іншими зовнішніми сервісами для забезпечення безшовного користувацького досвіду.
7. Забезпечення безпеки даних: Розробка надійних механізмів захисту особистих та фінансових даних користувачів.

ІІІ. Основна частина

У цій статті ми досліджуємо проблему, пов'язану зі спільними покупками товарів користувачами, та пропонуємо впровадження мікросервісної архітектури як ефективного рішення (див.рис. 1).

Основна мета нашої роботи полягає у створенні гнучкої та масштабованої системи, яка забезпечить легкість у координації замовлень, управлінні платежами та доставкою товарів. Ми впроваджуємо архітектуру, що включає різноманітні сервіси: від управління обліковими записами користувачів, каталогізації товарів, до обробки замовлень та платежів[1-2].

Окрім основних аспектів розробки мікросервісної архітектури, важливо звернути увагу на питання оптимізації продуктивності системи. Для забезпечення високої швидкості обробки запитів та мінімізації затримок рекомендується застосовувати технології кешування на різних рівнях, зокрема кешування результатів запитів до бази даних та використання розподілених кешуючих систем[3]. Також доцільно впровадити механізми балансування навантаження, що дозволить рівномірно розподіляти запити між сервісами та забезпечити стабільну роботу системи навіть при значних пікових навантаженнях. Такі підходи сприятимуть підвищенню загальної ефективності та надійності запропонованої платформи для спільних покупок.

Запропонована архітектура включає різноманітні сервіси: від управління обліковими записами користувачів, каталогізації товарів, до обробки замовлень та платежів. Особлива увага приділяється

сервісу спільних покупок, який управляє логікою групових замовлень та розподілом витрат. Інтеграція сервісу відгуків та оцінок, а також аналітичного сервісу, забезпечує зворотний зв'язок від користувачів та вдосконалення системи.

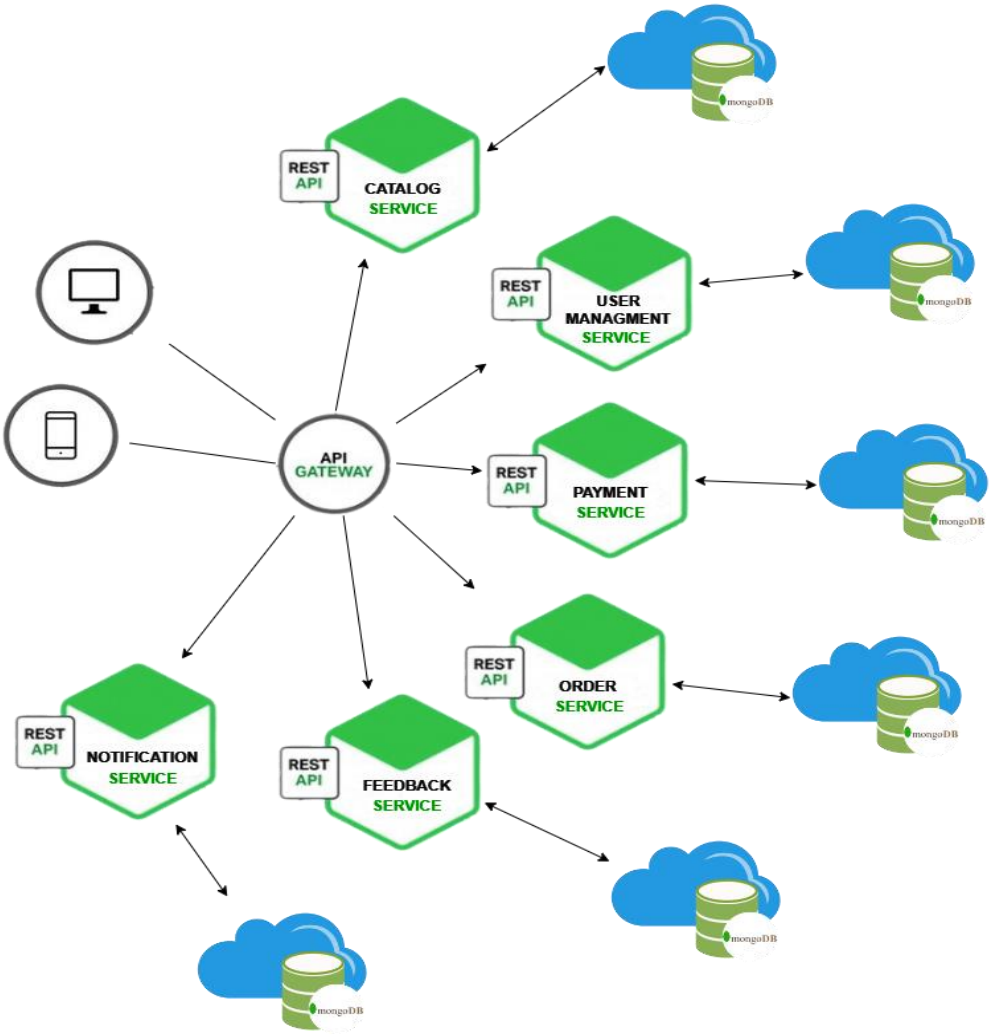


Рисунок 1 – Мікросервісна архітектура додатку сервісу для спільних покупок

На завершення, сервіс повідомлень підтримує активне спілкування з користувачами. Така архітектура дозволяє досягти високої стійкості та ефективності системи, водночас забезпечуючи легкість управління та масштабування.

Висновок

В даній роботі запропоновано розробку програмного рішення для системи групових покупок, яке відповідає сучасним змінам у споживацькій поведінці. Система спрощує процес спільних покупок, підвищуючи доступність та ефективність.

Також було запропоновано мікросервісну архітектура для керування акаунтами, каталогом продуктів, обробки замовлень та платежів, з особливою увагою до логіки групових замовлень і розподілу витрат.

Список використаних джерел

1. A. Pukas, A. Melnyk, I. Voytyuk, A. Yushko, M. Romanyuk and L. Honchar, "Transactional Business Application Based on Microservice Architecture," 2021 11th International Conference on Advanced Computer Information Technologies (ACIT), Deggendorf, Germany, 2021, pp. 564-567, doi: 10.1109/ACIT52158.2021.9548385.
2. Zhelev, Svetoslav&Rozeva, Anna. (2019). Using microservices and event driven architecture for big data stream processing. AIP Conference Proceedings. 2172. 090010. 10.1063/1.5133587.
3. Chopade, R. and Pachghare, V. (2020). Mongodb indexing for performance improvement., 529-539. https://doi.org/10.1007/978-981-15-0936-0_56

УДК 004.4

АНАЛІЗ МЕТОДІВ ІДЕНТИФІКАЦІЇ ІНТЕРВАЛЬНИХ МОДЕЛЕЙ СТАТИЧНИХ СИСТЕМ НА ОСНОВІ РОЙОВОГО ІНТЕЛЕКТУ

Дзиґа Ю.В.
Західноукраїнський національний університет
аспірант

I. Постановка проблеми

У статті розглянуто підходи до ідентифікації інтервальних моделей статичних систем з використанням методів ройового інтелекту [1]. Наведено огляд основних методів, що використовуються для побудови інтервальних моделей, та проаналізовано їхні переваги і недоліки. Запропоновано перспективні напрямки розвитку підходів на основі ройового інтелекту, які дозволяють підвищити точність та надійність ідентифікації.

Ідентифікація моделей систем є однією з ключових задач у сфері моделювання та керування складними системами [2-4]. Особливий інтерес представляють інтервальні моделі [5], які дозволяють враховувати невизначеність та варіабельність параметрів системи. У сучасних дослідженнях активно використовуються методи ройового інтелекту (swarm intelligence), що імітують колективну поведінку соціальних організмів (наприклад, мурах, бджіл, рибних косяків) для вирішення складних оптимізаційних задач [6].

II. Мета роботи

Метою дослідження є аналіз існуючих методів ройового інтелекту (swarm intelligence), які доцільно використовувати для ідентифікації математичних моделей складних систем.

III. Методи ройового інтелекту в ідентифікації інтервальних моделей

Ройовий інтелект включає в себе декілька методів, серед яких найпоширенішими є алгоритми оптимізації на базі мурашиної колонії (ACO), оптимізації рою частинок (PSO), та штучних бджолиних колоній (ABC) [7]. Кожен із цих методів має свої особливості та області застосування.

Оптимізація рою частинок (PSO). Цей метод є одним із найбільш популярних у сфері оптимізації. Основою методу є симуляція руху частинок у просторі параметрів, де кожна частинка намагається знайти оптимальне значення параметра за допомогою власного досвіду та досвіду сусідів. PSO дозволяє ефективно шукати глобальні екстремуми у великих просторах параметрів, що робить його привабливим для ідентифікації інтервальних моделей.

Оптимізація на базі мурашиної колонії (ACO) [8] імітує поведінку мурах під час пошуку їжі, що дозволяє вирішувати задачі комівояжера та інші оптимізаційні задачі, де важливо знайти оптимальний шлях або послідовність дій. Використання ACO для ідентифікації інтервальних моделей дозволяє обчислювати оптимальні інтервали параметрів за допомогою ефективного обстеження простору рішень.

Штучна бджолина колонія (ABC) [9] моделює поведінку бджіл під час пошуку нектару, де кожна бджола відповідає за обстеження певної області простору рішень. Цей метод дозволяє ефективно знаходити локальні екстремуми та застосовується для уточнення інтервалів параметрів у процесі ідентифікації моделей.

IV. Порівняльний аналіз методів

Для оцінки ефективності методів ройового інтелекту в задачах ідентифікації інтервальних моделей статичних систем важливо враховувати кілька критеріїв, таких як точність ідентифікації, обчислювальна складність, швидкість збіжності, та стійкість до локальних екстремумів.

Точність ідентифікації. PSO забезпечує високу точність завдяки здатності швидко знаходити глобальний екстремум. Водночас ACO та ABC мають схильність до швидкої збіжності, але можуть бути менш ефективними у випадку багатовимірних задач з великою кількістю локальних екстремумів.

Різні методи ройового інтелекту мають різний рівень обчислювальної складності, що впливає на їхню придатність для конкретних задач. Розглянемо детальніше аналіз складності наведених вище алгоритмів.

Оптимізація рою частинок (PSO). Обчислювальна складність PSO залежить від кількості частинок у рої та кількості ітерацій, необхідних для досягнення збіжності. Загальну складність можна

оцінити як $O(N \cdot T \cdot D)$, де N — кількість частинок, T — кількість ітерацій, а D — розмірність простору пошуку (кількість параметрів). PSO є порівняно ефективним методом, оскільки кожна частинка вимагає лише обчислення своєї нової позиції на основі власного і глобального досвіду, що зменшує обчислювальні витрати порівняно з іншими методами.

Оптимізація на базі мурашиної колонії (ACO). Обчислювальна складність ACO є значно вищою через необхідність симуляції поведінки великої кількості мурах, які шукають оптимальний шлях, обчислюючи феромонні сліди та виконуючи локальні пошуки. Складність ACO можна оцінити як $O(M \cdot T \cdot C)$, де M — кількість мурах, T — кількість ітерацій, а C — складність обчислення критерію якості на кожному кроці (залежить від задачі). ACO зазвичай вимагає значних ресурсів для моделювання феромонних карт та обчислення маршрутів кожної мурашки, що збільшує час виконання, особливо при великих розмірах задачі.

Штучна бджолина колонія (ABC). Метод ABC також має високу обчислювальну складність, яка залежить від кількості бджіл та ітерацій. Загальна складність ABC можна оцінити як $O(B \cdot T \cdot E)$, де B — кількість бджіл, T — кількість ітерацій, а E — кількість обчислень для оцінки якості розв'язку. Оскільки кожна бджола виконує окреме обстеження простору рішень, метод ABC може вимагати більше часу для збіжності, особливо в задачах з високою розмірністю або складною структурою простору пошуку.

Загалом, PSO є менш ресурсозатратним методом порівняно з ACO та ABC завдяки простішому механізму обчислень та меншій кількості необхідних операцій на кожному етапі. Однак, ACO та ABC можуть забезпечувати кращу якість рішень у складних задачах завдяки більш інтенсивному обстеженню простору рішень, хоча це й вимагає більшого обсягу обчислень. Швидкість збіжності. PSO характеризується швидкою збіжністю до оптимального рішення, однак може застрягнути в локальних екстремумах. ACO, завдяки своїй колективній природі, може краще досліджувати простір рішень, але має повільнішу збіжність у випадку складних задач.

Висновок

У роботі досліджено методи ройового інтелекту, такі як PSO, ACO та ABC, які є потужними інструментами для ідентифікації інтервальних моделей статичних систем. Кожен із розглянутих методів має свої переваги та недоліки, що визначають області їх застосування. PSO є найбільш ефективним для задач, де важлива швидкість та точність ідентифікації, тоді як ACO та ABC можуть бути корисними в задачах з високою невизначеністю та складною топологією простору рішень. Подальші дослідження можуть бути спрямовані на комбінування цих методів для підвищення надійності та точності ідентифікації.

Список використаних джерел

1. Смірнов О. О., Бондаренко О. А. Застосування методів ройового інтелекту для ідентифікації інтервальних моделей // Математичне моделювання та обчислювальні методи. — 2018. — Т. 14, № 2. — С. 94-102.
2. Dyvak M., Pukas A., Manzhula V., Kasatkina N., Komar M., Zabchuk V. The Task of Parametric Identification the Interval Models with Nonlinear Parameters. - 2022 12th International Conference on Advanced Computer Information Technologies (ACIT).- 2022. - p. 106-111.
3. Крепич С.Я. Моделювання та забезпечення функціональної придатності статичних систем методами аналізу інтервальних даних/ дисертація на здобуття ступ. к.т.н. – Тернопіль, 2016. 166с.
4. Крепич С.Я., Співак І.Я. Оцінювання часової складності застосування методу Монте-Карло та інтервального аналізу даних для встановлення функціональної придатності РЕК. Сучасні комп'ютерні інформаційні технології: Матеріали III Всеукраїнської школи-семінару молодих вчених і студентів ACIT'2013. – Тернопіль: Економічна думка, 2013. – С.36-37
5. Dyvak M., Porplytsya N., Spivak I., Tymchyshyn V., Krepych S., Taraj M.. The Method of Modeling the Characteristics of a Swarm of UAVs as an Object with Distributed Parameters Based on the Analysis of Interval Data. 2023 13th International Conference on Advanced Computer Information Technologies (ACIT). – 2023. – p. 165-169
6. Eberhart R., Kennedy J. A new optimizer using particle swarm theory // Proceedings of the Sixth International Symposium on Micro Machine and Human Science. — 1995. — P. 39-43.
7. Karaboga D., Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm // Journal of Global Optimization. — 2007. — Vol. 39, No. 3. — P. 459-471.
8. Dorigo M., Caro G. Di, Gambardella L.M. Ant algorithms for discrete optimization // Artificial Life. — 1999. — Vol. 5, No.2. — P. 137-172.
9. Voytyuk I., Porplytsya N., Pukas A., Dyvak T.. Identification the interval difference operators based on artificial bee colony algorithm in task of modeling the air pollution from vehicular traffic. 2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM). – 2017. – p.58-62.

УДК 330.341.1

ВИЗНАЧЕННЯ ПОНЯТТЯ ІННОВАЦІЇ У СУЧАСНИХ УМОВАХ

Юзьвак В.М
Akademia WSB
магістрант

І. Постановка проблеми

У сучасному світі, що швидко змінюється, інновації відіграють ключову роль у розвитку економіки, технологій та суспільства в цілому. Однак, незважаючи на широке використання терміну "інновація", його визначення часто варіюється залежно від контексту та галузі застосування. Це створює певні труднощі у розумінні та оцінці інноваційних процесів, що відбуваються у різних сферах людської діяльності.

ІІ. Мета роботи

Метою даної роботи є комплексний аналіз та уточнення поняття "інновація" в контексті сучасних умов. Основні аспекти цієї мети включають:

1. Систематизація існуючих підходів до визначення інновацій у різних галузях науки та практики.
2. Виявлення ключових характеристик інновацій, що є спільними для різних сфер їх застосування.
3. Розробка уніфікованого визначення інновації, яке враховує сучасні тенденції та може бути застосоване у різних контекстах.
4. Визначення критеріїв оцінки інноваційності продуктів, процесів та ідей.

ІІІ. Основна частина

У цій роботі ми досліджуємо еволюцію поняття "інновація" та його сучасне розуміння в різних контекстах. Ми пропонуємо комплексний підхід до визначення інновацій, який враховує їх багатогранну природу.

Основна мета нашого дослідження полягає у створенні уніфікованого визначення інновації, яке буде відображати її сутність у сучасному світі. Ми аналізуємо різні аспекти інновацій: від технологічних проривів до соціальних інновацій, від продуктових інновацій до інноваційних бізнес-моделей.

Історично поняття інновації з наукової точки зору трактувалося як результат (J. Schumpeter, P. Drucker, P. Kotler) або процес (M. Porter, R. Griffin) [1-3]. Вважаємо, що у сучасному світі для визначення поняття інновації потрібно підходити до інновації як комплексного явища.

Наведемо приклад. Підприємство займається дистрибуцією свіжих фруктів та овочів у великому місті. Традиційно маршрути доставки плануються логістами вручну, що часто призводить до неоптимальних рішень, затримок та додаткових витрат. Компанія впроваджує систему планування маршрутів на основі нейронних мереж. Ця система враховує множину факторів: 1) поточний трафік на дорогах, 2) прогноз погоди, 3) часові вікна доставки для кожного клієнта, 4) вантажопідйомність та технічні характеристики транспортних засобів, 5) термін придатності продукції, 6) історичні дані про швидкість розвантаження у різних точках. Нейронна мережа аналізує ці дані та генерує оптимальні маршрути доставки в режимі реального часу. У результаті витрати були скорочені (зменшився загальний пробіг транспортних засобів, скоротився час доставки, зменшилася кількість повернень через затримки доставки), а задоволеність клієнтів зросла. З точки зору кінцевого споживача продукт (доставлені свіжі фрукти) залишилися тими самими, деякі з користувачів могли і не помітити жодних змін, і якщо і помітили покращення, то не знали його причини. Чи можна говорити про інновацію у даному випадку?

Для відповіді на дане питання дамо визначення поняття інновації і його кваліфікаційним характеристикам.

Інновація – це комплексне, багатовимірне явище, яке передбачає впровадження нових або суттєво вдосконалених продуктів, послуг, процесів або бізнес-моделей, що створюють додаткову цінність для окремих суб'єктів, організацій або суспільства в цілому.

При визначенні кваліфікаційних характеристик важливо не тільки навести їх перелік, але прояснити умови їх застосування. У випадку кваліфікації наявності чи відсутності інновації у конкретній ситуації слід насамперед взяти до уваги точку дотику інновації до бізнес-моделі, процесу,

продукту чи послуги. Наприклад, у запитанні, яке наведене вище, сумнів виникає через порівняння змін у зовнішніх відносинах між підприємством та споживачами, в той час як інноваційні зміни мали місце в удосконаленні внутрішнього процесу.

Рисунок 1 – Основні кваліфікаційні ознаки інновації

Розглянемо нашу ситуацію на відповідність вищенаведеним ознакам.

Новизна – використання нейронних мереж для оптимізації маршрутів є новим підходом для компанії.

Практична реалізація – система впроваджена та активно використовується.

Економічний ефект – значне зниження витрат та підвищення ефективності.

Системність: рішення враховує множину взаємопов'язаних факторів.

Адаптивність: система адаптується до змін у реальному часі.

Сталість: зменшення викидів CO2 через оптимізацію маршрутів сприяє екологічній стійкості.

Отже, розглянута вище ситуація повністю відповідає нашому визначенню інновації

Висновок

У даній роботі запропоновано комплексний підхід до визначення поняття інновації в сучасних умовах. Ми визначили ключові характеристики інновацій, які залишаються актуальними незалежно від сфери їх застосування, та запропонували уніфіковане визначення, яке враховує багатогранну природу інновацій.

Також було розглянуто вплив сучасних технологічних та соціальних факторів на розуміння інновацій. Наведений приклад використання нейронних мереж для оптимізації маршрутів доставки демонструє, що інновації можуть відбуватися на рівні внутрішніх процесів підприємства, не змінюючи безпосередньо продукт чи послугу для кінцевого споживача. Це підкреслює комплексність та багатовимірність поняття інновації в сучасних умовах, де технологічні рішення, зокрема штучний інтелект, відіграють все більшу роль у підвищенні ефективності бізнес-процесів та створенні конкурентних переваг.

Список використаних джерел

1. Білоусько Р.С. Інновації в забезпеченні конкурентоспроможності аграрних підприємств. Механізми забезпечення сталого розвитку економіки: проблеми, перспективи, міжнародний досвід: матер. III Міжнар. наук.-практ. інтернет-конф., м. Харків. 10 листопада 2022 р. Харків, ДБТУ. 2022. С.45-47.

2. Шовкун-Заблоцька Л. В., Шовкун З. М. Класифікація інновацій та їх специфіка. Розвиток бізнесу в контексті європейської інтеграції: глобальні виклики, стратегічні пріоритети, реалії та перспективи: матеріали Міжнар. науково-практичної конф., 07 червня 2024 р. Держ. біотехнологічний ун-т. Харків, 2024. С. 451-452

3. Ганас Л. М., Дорош І. М., Петрова Я. Ю. Характеристика та типологія інновацій як економічної категорії. Економіка та держава. 2020. № 4. С. 201–205. DOI: 10.32702/2306-6806.2020.4.201

Наукове видання

Комп’ютерні інформаційні технології

Матеріали
школи-семінару молодих вчених і студентів
СІТ’2024

Відповідальний за випуск:
Пукас А.В., д.т.н., професор,
завідувач кафедри комп’ютерних наук
Західноукраїнського національного університету

Підписано до друку 01.05.2024р.
Формат 60х84/16. Папір офсетний.
Друк офсетний. Зам. № 9-365
Тираж 25 прим.

Віддруковано ФО-П Шпак В. Б.
Свідоцтво про державну реєстрацію В02 № 924434 від 11.12.2006 р.
Свідоцтво платника податку: Серія Е № 897220
м. Тернопіль, вул. Просвіти, 6.
тел. 8 097 299 38 99
E-mail: tooums@ukr.net