

Міністерство освіти і науки України
Чорноморський національний університет
імені Петра Могили



«Інформаційні технології та інженерія»

*Всеукраїнська науково-практична конференція
молодих вчених, аспірантів і студентів*

ТЕЗИ

9–11 лютого 2022 року

Програмний комітет: Бойко А. П., Кондратенко Ю. П., Гожий О. П., Журавська І. М., Хомченко А. Н., Давиденко Є. О., Кірей К. О., Крайник Я. М., Беліков О. Є.

Організаційний комітет: Горбань Г. В., Сіденко Є. В., Кулаковська І. В., Антіпова К. О., Таранов М. О., Савічева Т. Ю.

Секретар конференції: Горбань Г. В.

Інформаційні технології та інженерія : Всеукраїнська науково-практична конференція молодих вчених, аспірантів і студентів : тези доп., 9–11 лютого 2022 р. / ЧНУ імені Петра Могили. Миколаїв, 2022. 182 с.

Напрями роботи конференції: «Інформаційні системи та їх інтелектуалізація», «Машинне навчання та штучний інтелект», «Системний аналіз, моделі і засоби підтримки прийняття рішень», «Автоматизація та інтегровані комп'ютерні технології», «Методи і засоби комп'ютерної інженерії», «Методи і засоби програмної інженерії», «Вебтехнології та вебдизайн», «Інформаційні технології у навчальному процесі».

© ЧНУ імені Петра Могили, 2022

ЗМІСТ

Інформаційні системи та їх інтелектуалізація

<i>Агафонов А. С., Сіденко Є. В.</i> Моніторинг та прогнозування параметрів системи очищення води.....	8
<i>Березоруцька О. В., Рудніченко М. Д.</i> Модель розумного саду на базі мікроконтролера ESP8266, використовуючи Blynk	9
<i>Бубнова Д. В., Шевченко О. В., Кондратенко Ю. П.</i> «Порівняльний аналіз та особливості застосування AND-операторів в «м'яких» обчисленнях	13
<i>Д'яченко Т. Ю., Болюбаш Н. М.</i> Управління логістичною діяльністю на основі технології блокчейн	16
<i>Зіменіна Ю. М., Рудніченко М. Д.</i> Аналіз існуючих програмних продуктів для управління інвестиціями фізичної особи	19
<i>Кутняк В. Є., Кондратенко Г. В.</i> Аналіз існуючих програмних продуктів для управління інвестиціями фізичної особи	22
<i>Пушняк І. А., Калініна І. О.</i> Інтелектуальна система обробки інформації по якості фільтрації води	24
<i>Рокитенко В. М., Рудніченко М. Д.</i> Штучний інтелект та інтернет речей	26
<i>Ткаченко М. П., Кучеренко Є. А., Журавська І. М.</i> Автоматизована система виявлення бот-атак на основі аналізу користувацьких профілів у соціальних мережах	28
<i>Шаутін О. В., Кулаковська І. В.</i> Адміністрування інформаційної системи з реалізацією алгоритму розпізнавання облич	31
<i>Volkov A., Blazhko O.</i> Database of human physical exercises in virtual reality	34

Машинне навчання та штучний інтелект

<i>Баршніков В. О., Мсцанінов О. П.</i> Послідовність розробки системи інтелектуального аналізу даних розумного будинку	38
---	----

Борисов М. В., Гожий О. П. Використання методів машинного навчання для прогнозування чистої вихідної електроенергії електростанції комбінованого циклу	40
Івченко І. О., Калініна І. О. «Застосування методів машинного навчання для вирішення задачі біологічної класифікації	43
Костиця М. А., Кондратенко Ю. П. Дослідження впливу архітектур згорткових нейронних мереж на ефективність сегментації об'єктів....	44
Малімон О. О., Сіденко Є. В. Інтелектуальний аналіз та класифікація текстів з використанням технологій штучного інтелекту	47
Нечахін В. В. Застосування нейромережевої архітектури LSTM в системі керування сонячною електростанцією	50
Петрович В. І., Кондратенко Г. В. Дослідження методів машинного зору для автоматизованого діагностування хвороб шкіри.....	52
Под'ячев А. Д., Гожий О. П. Використання нечіткої логіки в ігровому штучному інтелекті	55
Попель О. О., Гожий О. П. Хмарні обчислення задач машинного навчання та штучного інтелекту з використанням інструментів Infrastructure as Code.....	57
Савчук О. А., Кондратенко Ю. П. Діагностування COVID-19 з використанням методів штучного інтелекту.....	60
Скакун Є. І., Гожий О. П. Інтелектуальна система для визначення локацій методом машинного навчання.....	63
Скубак М. Д., Калініна І. О. Інтелектуальна система аналізу контенту музичного вебсерверу для Андроїд-застосунку.....	65
Скубак О. Д., Петроченко О. О., Калініна І. О. Аналіз часових рядів за допомогою машинного навчання.....	67
Фінажсин М. Ф., Калініна І. О. Інтелектуальна система прогнозування на основі індексів криптовалюти.....	69

Системний аналіз, моделі і засоби підтримки прийняття рішень

Біряк Д. В., Рудніченко М. Д. Перспективи аналізу даних захворювань COVID-19 на базі використання машинного навчання.....	72
--	----

<i>Гапішко Д. О., Сіденко Є. В.</i> Система оцінювання інноваційних проєктів на основі модифікованого Fuzzy TOPSIS	74
<i>Гончарова Н. В.</i> Принципи побудови діаграм впливів для вирішення задач сценарного аналізу ризикоутворюючих факторів в закладах вищої освіти	76
<i>Грохольська А. І., Сіденко Є. В.</i> Дослідження методу багатокритеріального прийняття рішень FUZZY MARCOS при формуванні рейтингу університетів.....	79
<i>Лубковський В. О., Кондратенко Ю. П.</i> СППР для оцінювання проєктів електростанцій на основі методу Fuzzy Best-Worst	83
<i>Паленко Р. О., Сіденко Є. В.</i> Комбінований FUZZY TOPSIS-DEMATEL для багатокритерійного прийняття рішень в медицині.....	85
<i>Стовманенко В. О., Давиденко Є. О.</i> Огляд реалізацій систем підтримки прийняття рішень	87
<i>Таранчук Д. О., Кондратенко Г. В.</i> Рекомендаційна система для вибору мобільних пристроїв на основі методів прийняття рішень.....	89
<i>Франчук О. В., Кондратенко Ю. П.</i> Планування транспортних маршрутів для задач CVRP в умовах невизначеності	93
<i>Юр'єва А. О., Кондратенко Г. В.</i> Дослідження впливу виробництва харчових продуктів на довкілля з використанням методів багатокритеріального прийняття рішень	95

Автоматизація та інтегровані комп'ютерні технології

<i>Войтасик А. М.</i> Розробка лабораторного стенду по дослідженню принципу роботи електроакустичного перетворювача	99
<i>Головченко Д. С., Кубов В. І.</i> Передача даних з сенсорів дистанційних об'єктів	102
<i>Мальцев Є. Є., Кудінов Д. В., Сідєлев М. І.</i> SCADA-системи в системах автоматизації водяних насосних станцій	104
<i>Нерега М. В., Бєліков О. Є.</i> Автоматичний дезінфектор для рук	106
<i>Паламарчук А. О., Димитров Ю. Ю., Кубов В. І.</i> Дистанційне управління обладнанням систем автоматизації об'єктів водних ресурсів.....	108

<i>Розганяєв Д. О., Прищепов О. Ф.</i> Система Electronic Stability Program (ESP) для вітчизняних автомобілів	110
<i>Терещенко Г. Ю., Беліков О. Є.</i> Автоматизована система видачі лікарських засобів за приписом лікаря з функцією віддаленого сповіщення	111
<i>Тіхоміров К. А., Прищепов О. Ф.</i> Автоматизована система керування безпілотним літаючим апаратом	113

Методи і засоби комп'ютерної інженерії

<i>Важоха В. Б., Крайник Я. М.</i> Комплексна система управління сервісами з використанням засобів контейнеризації.....	117
<i>Горбуров Л. М., Крайник Я. М.</i> Особливості проектування інтелектуальних систем для дослідження нейронних мереж	118
<i>Дяденко А. С., Бойко А. П.</i> Багатокористувацький серверний застосунок на базі рушія UNREAL ENGINE	120

Методи і засоби програмної інженерії

<i>Голуб Р. Р., Дворецький М. Л.</i> Використання архітектурних-патернів MVVM та MVP при розробці мобільних застосунків мовою Kotlin .	123
<i>Гусєва-Божаткіна В. А.</i> Визначення метрик та довірчого інтервалу для побудови регресійного рівняння для оцінювання тривалості розробки програмних застосунків, що розробляються мовою COBOL для платформи MAINFRAME	125
<i>Кандиба І. О., Ткаченко М. П.</i> Аналіз складових телекомунікаційної мережі закладу вищої освіти засобами Python.....	129
<i>Кожушан М. Г., Кунгурцев О. Б.</i> «Фільтрація тлумачень термінів в електронному словнику	132
<i>Кошовий Р. В., Кірей К. О.</i> Застосунок для оцінки збіжностей файлів із кодом С-подібних мов програмування	134
<i>Тіщенко С. Є., Рудніченко М. Д.</i> Мобільне застосування для обліку особистих коштів	137
<i>Фіник В. Ю., Журавська І. М.</i> Цифрові технології обробки наборів даних для оцінювання реабілітації пост-інсультних пацієнтів	140
<i>Червіцька К. В., Комлева Н. О.</i> Особливості архітектури вебсайту для кафедри системного програмного забезпечення	142

Червіцька К. В., Комлева Н. О., Попова М. О. Використання дошки Kanban для проектування вебсайту кафедри та оцінка його показників зручності використання 145

Вебтехнології та вебдизайн

Бабін О. С., Болюбаш Н. М. Вебзастосунок забезпечення комунікаційної діяльності в IT- компаніях 148

Климчук А. М., Дворецький М. Л. Аналітична система класифікації кандидатів на вакансію на основі аналізу характеристик 151

Кондратюк І. В., Воробйова А. І. Конверсія електронної торгівлі в Shopify 154

Лукіна А. В., Обухова К. О., Ткаченко М. П. Розробка вебсайту з SEO-підтримкою та маркетинговим аналізом 156

Медяник Є. І., Рудніченко М. Д. «Використання фреймворку Laravel при розробці вебзастосунків на основі MVC 158

Павлющенко М. М., Обухова К. О., Ткаченко М. П. Розробка інформаційної системи компанії-реалізатора автомобілів з пробігом 161

Полосмак Н. В., Донченко М. В. Розробка інтернет-магазину із застосуванням методів аналізу даних 164

Сапонько Д. С., Кулаковська І. В. Переваги використання мікросервісної архітектури над іншими типами архітектур 165

Сокрут А. А., Воробйова А. І. Розробка інтернет-магазину роздрібного продажу товарів сучасної моди 168

Третяк В. О., Лисенков Е. А. Управління проектами за допомогою Scrum 172

Інформаційні технології у навчальному процесі

Дворецька М. М., Воробйова А. І. Визначення оптимального розподілу значень коефіцієнтів складових конкурсного балу абітурієнта 174

Плотніков М. С. «Інтерактивна система навчання для зручної візуалізації об'єктів з використанням доповненої реальності 178

Інформаційні системи та їх інтелектуалізація

УДК 004.02

Агафонов А. С., Сіденко Є. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

МОНІТОРИНГ ТА ПРОГНОЗУВАННЯ ПАРАМЕТРІВ СИСТЕМИ ОЧИЩЕННЯ ВОДИ

Важко переоцінити значення води в забезпеченні існування життя на Землі, адже на планеті немає жодного живого організму, який би не містив воду. Для людини дуже важливо пити чисту воду – це одна з основних складових здоров'я, як і чисте повітря, здорова їжа і фізична активність. Але зараз стає очевидним, що ми споживаємо дуже багато води, більше, ніж Планета може відновити і очистити. Нажаль, багато країн вже досягли граничних можливостей водокористування: споживання прісної води за останні пів століття потроїлося. Понад мільярд людей на планеті не мають доступу до чистої питної води і ще 2.6 мільярда відчувають нестачу води для потреб гігієни.

Саме тому на сьогоднішній день однією із важливіших систем у побуті справедливо вважають систему очищення води, чи як вона називається в магазинах – фільтр для води. Найефективнішим вважається такий підвид фільтру, як осмос. Він працює за принципом повного очищення води, але так як така вода не є корисною для людини, далі ця вода мінералізується, тобто до неї штучно додаються різні мінерали та корисні речовини.

В рамках дослідження розроблена система виконує функцію моніторингу, тобто вона дозволяє у режимі реального часу спостерігати за показниками води: температура, тиск, вхідний та вихідний показник відносної чистоти (TDS), який вимірюється у ppm, потік води(л/хв), кількість витраченої води та залишок ресурсу картриджа (через скільки днів або літрів його потрібно буде міняти). Система працює на апаратно-обчислювальній платформі Arduino, що натомість відправляє дані до серверу, який буде зберігати їх та візуалізувати користувачу. Для функціонування системі потрібна платформа, що має принаймні шість аналогових входів/виходів та встановлену систему зв'язку (Wi-Fi). Через це, у системі використовується Arduino моделі esp32, так як це майже єдина система, що відповідає заданим критеріям. Теоретично, ця система є масштабованою, так як до неї можна під'єднати ще 1 аналоговий та до 15 цифрових датчиків, та відстежувати їх показники.

Так як програмне забезпечення є оптимізованим з метою зменшення споживання напруги, за розрахунками, від двох акумуляторів типу 18650 вона зможе працювати до двох років (з урахуванням саморозряду). Систему можна також жити від блоку живлення – потрібно, щоб він відповідав показникам – 5В та від 500мА.

Використовуючи модель експоненційного згладжування з трьох місяців вихідних даних було здійснено прогноз, що ресурс картриджів для фільтру становитиме приблизно 7700 літрів. Після здобуття даної інформації, вона була занесена до параметрів системи, і тепер при моніторингу можна бачити, скільки літрів і споживача залишилось до зміни на новий картридж, щоб не погіршувалась якість води.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Rob Hyndman, Anne B. Koehler, J. Keith Ord. Forecasting with Exponential Smoothing: The State Space Approach. 2008; 20:355 URL: <https://robjhyndman.com/expsmooth/>
2. Ahmad Farid Osman. A New Approach to Forecasting Based on Exponential Smoothing with Independent Regressors. 2012. 291 p. URL: <https://www.monash.edu/business/econometrics-and-business-statistics/research/publications/ebs/wp02-15.pdf>
3. Sever Spanulescu. Esp32 Programming for the Internet of Things. 2018. 197 p. URL: <https://play.google.com/store/books/details?id=y5uBDwAAQBAJ&rdid=book-y5uBDwAAQBAJ>

УДК 004

Березоручька О. В., Рудніченко М. Д.

*Національний університет «Одеська політехніка»,
м. Одеса, Україна*

МОДЕЛЬ РОЗУМНОГО САДУ НА БАЗІ МІКРОКОНТРОЛЕРА ESP8266, ВИКОРИСТОВУЮЧИ BLYNK

Інтернет речей (Internet of Things, IoT) - це технологія, яка об'єднує багато фізичних об'єктів, підключених до інтернету, в мережу та дозволяє їм збирати, аналізувати, обробляти та передавати дані іншим об'єктам. Інвестиційна компанія Goldman Sachs прогнозує, що продуктивність світового агробізнесу до 2050 року зросте на 70% [1]. Так як більшість земель, придатних для вирощування фруктів та овочів, вже задіяні, то майбутнє розвитку сфери рослинництва - за IoT.

Вирощування рослини - дуже відповідальний процес, тому що він залежить від багатьох факторів: полив, освітлення, вологість

навколишнього середовища, поживні речовини тощо. Без використання датчиків неможливо зрозуміти напевно, які показники оточують рослину. Для підтримки оптимального росту важливо керувати цими факторами. IoT-пристрої дають таку можливість, значно полегшуючи догляд за рослинами [2]. Використовуючи статистику з датчиків та аналізуючи отриману інформацію, можна приймати управлінські рішення щодо оптимального зростання культур та підвищення врожайності.

Метою даної роботи є розробка IoT-рішення розумного саду, яке буде надавати користувачеві можливість дистанційного догляду за рослинами: стежити за їх зростанням, використовуючи датчики температури, вологості, освітленості, штучне освітлення (стрічка світлодіодів) та полив (використовуючи сервомотор). Запропоноване програмне забезпечення дозволить автоматизувати цей процес.

Основною відмінністю існуючих на ринку IoT-систем для догляду за кімнатними рослинами від представленої в даній роботі є те, що в готових реалізаціях вирощування рослин відбувається повністю автоматично (за показниками датчиків) без можливості зміни запрограмованого процесу. Виростити на цих пристроях можна лише певні рослини, до яких готові програми вирощування. Якщо потрібно виростити іншу рослину, це не вдасться. Представлена модель дає можливість дистанційного керування освітленням та поливом через мобільний застосунок. Дані з датчиків аналізує людина і сама вирішує, чи включити інше освітлення або полити рослину, тобто користувачеві потрібні навички садівництва. Також ціна реалізації проекту буде значно нижчою від тих, що є на ринку.

Після встановлення система зможе передавати в режимі реального часу на мобільний пристрій такі дані з датчиків: температуру навколишнього середовища, вологість повітря, рівень освітленості. Також користувач зможе дистанційно відкривати клапан подачі води та включати додаткове штучне освітлення у темну пору доби (для рослин найкраще підходить світло червоного та синього спектру, оскільки воно сприяє покращенню фотосинтезу, а також прискореному зростанню розсади та коріння). На базі цієї моделі можна організувати розширене управління вирощуванням рослин: додати датчики кислотності та вологості ґрунту, вмісту поживних речовин у ній, автоматично регулювати освітленість залежно від сезону, замінити стрічку світлодіодів (штучне освітлення) на фітолампку (випромінює ультрафіолетове світіння, забезпечуючи рослини світлом, схожим зі світлом сонця), вентилувати приміщення та очищати повітря, підтримуючи його якість.

З метою реалізації проекту використовуватимемо плату ESP-12E на базі мікроконтролера ESP8266. Для вимірювання температури та вологості скористаємося датчиком DHT11. Для відкриття клапана для поливу рослин – сервомотор, для штучного освітлення - світлодіодну RGB-стрічку WS2812 на 8 світлодіодів. У якості датчика освітленості використовуємо MAX44009.

Для управління зі смартфона та безпечного підключення до хмари використовуємо Blynk та середовище розробки Arduino IDE. Blynk App дозволяє створювати інтерфейси програм iOS та Android для проектів без використання коду шляхом перетягування готових віджетів. Blynk-сервер несе відповідальність за зв'язок між смартфоном та апаратним забезпеченням. Також бібліотека Blynk дозволяє встановлювати зв'язок із сервером та обробляти всі вхідні та вихідні команди. Є можливість прямої маніпуляції підключеними пристроями без написання коду, можна легко додати нові функціональні можливості, використовуючи віртуальні піни.

Функціональну схему проекту і проект моделі розумного саду у мобільному застосунку у Blynk наведено на рис. 1-2. Програмна реалізація – <https://drive.google.com/file/d/1YyHiSfvavXb7e-zpynaubgccENIbUPh/view?usp=sharing>.

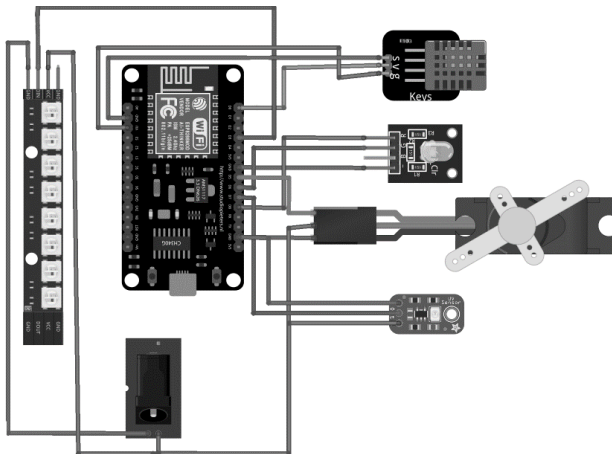


Рисунок 1 – Функціональна схема проекту

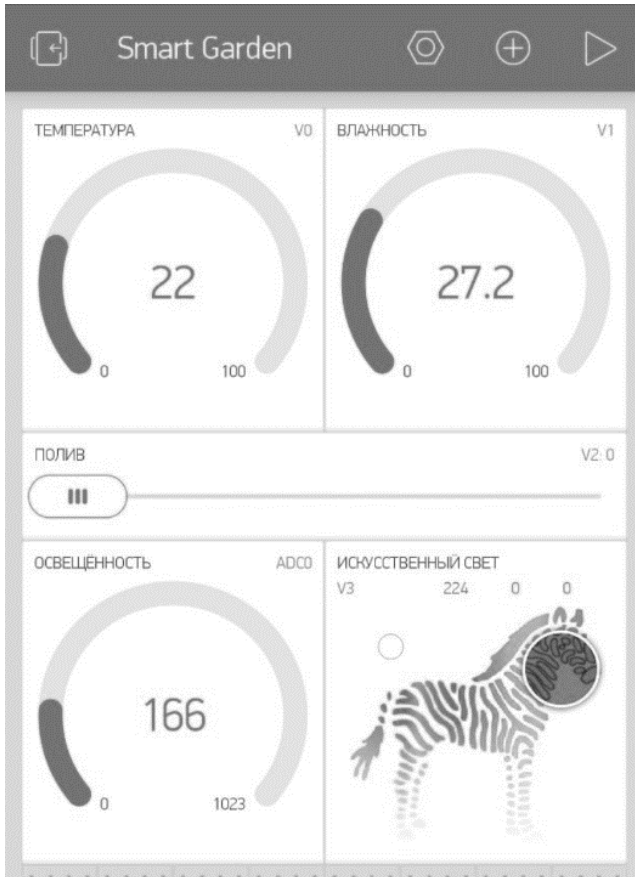


Рисунок 2 – Проект моделі розумного саду у Blynk

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ІНТЕРНЕТ РЕЧЕЙ У СІЛЬСЬКОМУ ГОСПОДАРСТВІ: 8 ПОРАД: Вебсайт. URL: <https://hub.kyivstar.ua/news/internet-rechej-u-silisikomu-gospodarstvi-8-porad/>
2. Сычев И. «Интернет растений» для фермерства в городе / Иван Сычев. 2014. URL: <https://habr.com/ru/post/364367/>

ПОРІВНЯЛЬНИЙ АНАЛІЗ ТА ОСОБЛИВОСТІ ЗАСТОСУВАННЯ AND-ОПЕРАТОРІВ В «М'ЯКИХ» ОБЧИСЛЕННЯХ

Метою даного дослідження є порівняльний аналіз обговорення особливостей застосування “AND”-операторів, що використовуються в “м’яких” обчисленнях при розв’язанні задач управління та прийняття рішень на основі нечіткої логіки. Для досягнення поставленої мети авторами виділені наступні задачі: (а) характеристика основних параметричних та непараметричних AND-операторів; (б) аналіз властивостей AND-операторів t -норми; (в) аналіз AND-операторів на основі середнього (арифметичні, гармонічні, геометричні та ін.); (г) вибір оптимального AND-оператора при проектуванні конкретної нечіткої системи; (д) моделювання “м’яких” обчислень з використанням різнотипних AND-операторів; (е) ранжування AND-операторів за ступенем оптимізму; (є) формулювання висновків і рекомендацій на основі результатів проведених досліджень.

AND-оператори. Оператор t -норми [1] є функцією T , що моделює операцію AND-перетину двох нечітких множин A і B , з наступними властивостями для всіх $x \in X$:

$$\text{Області відображення: } T : [0, 1] \times [0, 1] \rightarrow [0, 1] \quad (1)$$

$$\text{Властивість обнулення: } T(0, 0) = 0 \quad (2)$$

$$\text{Випадок при } \mu(x)=1: T(\mu_A(x), 1) = \mu_A(x), T(\mu_B(x), 1) = \mu_B(x) \quad (3)$$

$$\text{Комутативність: } T(\mu_A(x), \mu_B(x)) = T(\mu_B(x), \mu_A(x)) \quad (4)$$

$$\begin{aligned} \text{Асоціативність: } T(\mu_A(x), T(\mu_B(x), \mu_C(x))) = \\ = T(T(\mu_A(x), \mu_B(x)), \mu_C(x)) \end{aligned} \quad (5)$$

$$\begin{aligned} \text{Монотонність: } \mu_A(x) \leq \mu_C(x), \mu_B(x) \leq \mu_D(x) \Rightarrow \\ \Rightarrow T(\mu_A(x), \mu_B(x)) \leq T(\mu_C(x), \mu_D(x)) \end{aligned} \quad (6)$$

У виразах (1)-(6) використовуються позначення $\mu_A(x)$, $\mu_B(x)$, $\mu_C(x)$ та $\mu_D(x)$, що відповідають результатам фаззифікації параметра x на нечітких множинах A , B , C та D . Якщо значення $\mu_A(x)$, $\mu_B(x)$, $\mu_C(x)$ та $\mu_D(x)$ відповідають антецедентам нечітких баз правил, то їх агрегацію доцільно здійснювати з використанням відповідних AND-операторів. Слід зазначити: (а) властивість комутативності (4) означає, що порядок обчислення операндів не впливає на кінцевий результат; (б) властивість асоціативності (5) визначає можливість покрокового обчислення логічного добутку кількох множин шляхом добутку їхніх пар.

Розрізняють параметричні та непараметричні AND-оператори t -норми. Результати обчислень з використанням непараметричних AND-операторів t -норми є постійними для фіксованих значень $\mu_A(x)$, $\mu_B(x)$, $\mu_C(x)$ та $\mu_D(x)$, тоді як для параметричних t -норм операторів результати залежать від вибору значення показника параметризації, що є ступенем свободи відповідного AND-оператора. До непараметричних операторів t -норми [1] відносять наступні оператори: MIN-оператор, PROD, добуток Гамахера, добуток Ейнштейна, посилений добуток, обмежена різниця. До параметричних операторів t -норми [1] відносять AND-оператори: Дюбуа, Гамахера, Ягера.

AND-оператори на основі середнього. В даній доповіді також детально аналізуються як самі AND-оператори на основі середнього [1], так і їх властивості. Зокрема, це стосується AND-операторів на основі (а) середнього арифметичного, (б) середнього гармонійного та (в) середнього геометричного.

Особливу увагу приділено дослідженню узагальненого оператора середнього

$$\mu_{A_1 \cap A_2 \cap \dots \cap A_n}(x) = \frac{(\mu_{A_1}^\alpha(x_1) + \mu_{A_2}^\alpha(x_2) + \dots + \mu_{A_n}^\alpha(x_n))^{1/\alpha}}{n}, \forall x \in X. \quad (7)$$

Цей AND-оператор перетину є параметричним. При цьому параметром налаштування виступає показник α . Авторами досліджено вплив показника параметризації α на результати «м'яких» обчислень.

Підтверджено, що узагальнений AND-оператор (7) трансформується до різних тирів AND-операторів, зокрема, при:

- 1) $\alpha \rightarrow -\infty$, узагальнений оператор середнього зводиться до оператора MIN;
- 2) $\alpha = -1$, маємо оператор середнього гармонійного;
- 3) $\alpha = 0$, маємо оператор середнього геометричного;
- 4) $\alpha = 1$, маємо оператор середнього арифметичного;
- 5) $\alpha \rightarrow +\infty$ маємо оператор MAX.

Всі оператори, що формуються на основі оператора узагальненого середнього (7), відносяться до типу super-MIN та є більш оптимістичними, ніж оператор MIN. Ступінь оптимізму зростає зі зростанням коефіцієнта α у формулі (7).

Вибір оптимального AND-оператора. При виборі AND-оператора для конкретної нечіткої системи прагнуть отримати максимально ефективний результат при заданих вхідних параметрах, насамперед з погляду поставленої мети управління або прийняття рішення. Яким саме має бути спосіб вибору найкращого оператора однозначної відповіді немає, в зв'язку з чим на практиці AND-оператор (\cap) обирається: (а) інтуїтивно, виходячи з досвіду, (б) на основі яких-

небудь гіпотез, (в) методом проб або (г) застосуванням відповідних алгоритмів оптимізації (градієнтних, еволюційних, мультиагентних та ін.). Вибір оптимального AND-оператора - це задача структурної оптимізації конкретної нечіткої системи [2,3].

Ранжування AND-операторів. В доповіді обговорюються результати досліджень різнотипних AND-операторів для заданих результатів фаззифікації: $\mu_A(x) = 0,3$, $\mu_B(x) = 0,5$ та $\mu_C(x) = 0,8$. Отримано наступні результуючі значення $\mu_{A \cap B \cap C}(x)$ при застосуванні непараметричних AND-операторів: оператор мінімум - $\mu_{A \cap B \cap C}(x) = 0,3$; добуток Гамахера - $\mu_{A \cap B \cap C}(x) = 0,217$; алгебраїчний добуток - $\mu_{A \cap B \cap C}(x) = 0,12$; добуток Ейнштейна - $\mu_{A \cap B \cap C}(x) = 0,074$.

Відповідно до ступеня оптимізму, непараметричні AND-оператори t -норми можуть бути впорядковані наступним чином [1]: 1) мінімум, 2) добуток Гамахера, 3) алгебраїчний добуток, 4) добуток Ейнштейна, 5) обмежена різниця, 6) посилений добуток. Для наведеного прикладу також проведено дослідження результатів агрегації $\mu_{A \cap B \cap C}(x)$ при застосуванні параметричних операторів та AND-операторів на основі середнього. Сформовано узагальнений рейтинговий ряд всіх розглянутих AND-операторів за ступенем оптимізму.

Висновки. В процесі наукового дослідження проведено порівняльний аналіз AND-операторів в “м’яких” обчисленнях та визначено особливості їх застосування. На основі асоціативної властивості (4) розроблена програмна реалізація та блок-схема алгоритму агрегації для довільного числа N компонентів: $\mu_A(x)$, $\mu_B(x)$, $\mu_C(x)$, $\mu_D(x)$, ..., $\mu_N(x)$. В результаті проведення аналізу властивостей, відповідних розрахунків та ранжування AND-операторів за ступенем оптимізму підтверджена доцільність використання розглянутої гамми AND-операторів для структурної оптимізації конкретних нечітких систем [2,3] в процесі їх проектування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Piegat, A. Fuzzy Modeling and Control. Springer, Heidelberg (2001).
2. Kondratenko, Y.P.; Simon, D. Structural and parametric optimization of fuzzy controdecision-making systems. In Recent Developments and the New Direction in Soft-Computing Foundations and Applications. Studies in Fuzziness and Soft Computing; Zadeh, L., Yager, R., Shahbazova, S., Reformat, M., Kreinovich, V., Eds.; Springer: Cham, Switzerland, 2018; Volume 361, pp. 273–289.

3. Kondratenko, Y.P.; Klymenko, L.P.; Al Zu'bi, E.Y.M. Structural Optimization of Fuzzy Systems' Rules Base and Aggregation Models. *Kybernetes* 2013, 42, pp. 831–843.

УДК 004.9

Д'яченко Т. Ю., Болюбащ Н. М.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

УПРАВЛІННЯ ЛОГІСТИЧНОЮ ДІЯЛЬНІСТЮ НА ОСНОВІ ТЕХНОЛОГІЇ БЛОКЧЕЙН

В умовах глобалізації сучасного суспільства ланцюги поставок продукції, пов'язані з її виробництвом та доставкою до кінцевого споживача, містять велику кількість об'єктів, які можуть бути розміщені у різних країнах світу. Відстеження та контролювання поставки продукції потребує прозорого моніторингу великої кількості різних операцій усередині мережі роздрібних торговців, дистриб'юторів, транспортників, складських приміщень та інших постачальників. Застосування технології блокчейн у логістиці дозволяє підвищити ефективність товароруку між усіма учасниками ланцюга поставок.

Оптимізація та підвищення ефективності управління логістичними каналами є актуальною задачею, яка розв'язується з використанням сучасних інформаційних технологій. Однак технологія блокчейн у науковій літературі представлена недостатньо широко. Серед програмних рішень, що реалізують технологію блокчейн, існує два ключових напрямки застосування блокчейн – фінансові технології й Інтернет речей. Незважаючи на великий потенціал, кількість компаній, які впровадили й адаптували блокчейн під свої потреби в сфері логістики є невеликою. Метою роботи є дослідження переваг та виявлення основних підходів використання технології блокчейн в управлінні логістичною діяльністю, які підвищують ефективність забезпечення прозорості та безпеки товароруку у наскрізному моніторингу фінансових та інформаційних потоків.

Логістичний ланцюг у технології блокчейн є розподіленою між усіма учасниками поставок базою даних, у якій відсутнє централізоване управління [1]. Дані зберігаються на комп'ютерах різних користувачів і представлені у вигляді ланцюжка послідовних взаємопов'язаних блоків – інформаційних пакетів, кожен з яких містить у собі усі попередні відомості та частину нових. Усі користувачі ланцюга є рівноправними

й утворюють мережу комп'ютерів, на кожному з яких зберігається копія даних блокчейн.

Ключовими в технології блокчейн є наступні операції: взаємодія учасників, формування блоків, формування ключів, шифровка записів. У логістиці дані, що утворюють ланцюг блоків, можуть містити таку інформацію: відомості про угоди, людей, об'єкти, транзакції, серійні номери, видані кредити та інші дані, які стосуються товароруху між учасниками поставок продукції. Ланцюжки блоків дозволяють відправляти дані в будь-яку точку світу, де буде доступний файл блокчейна за умови наявності закритого ключа, створеного за криптографічним алгоритмом, щоб дозволити доступ тільки до тих блоків, якими володіє конкретна особа. Надаючи закритий ключ, можна передати сторонній особі певну інформацію, яка зберігається у відповідному розділі ланцюжка блоків.

Використання складної системи шифрування забезпечує наявність у кожного блока свого унікального ключа, даючи можливість кожному користувачеві змінювати тільки ті блоки ланцюга, до яких у них є доступ. Ця особливість технології блокчейн дозволяє реалізувати довірчі відносини між користувачами й унеможливає підробку інформації у розподіленій базі даних. Що гарантує збереження і точність відомостей, які зберігаються в мережі, забезпечуючи їх цілісність і достовірність. Формування ключів відбувається шляхом хешування – перетворення інформації у рядок букв та цифр фіксованої довжини. Стандарти хешування можуть мати різну швидкість, розмір хешу та механізм кодування. До найбільш швидкісних та популярних відносять стандарти CRC, MD4, MD5 та SHA.

Перевірку операцій та підтвердження правильності їх запису здійснює вбудований механізм – алгоритм консенсусу. Алгоритм консенсусу гарантує дотримання правил протоколу й вірогідність усіх транзакцій. Найбільш затребувані алгоритми консенсусу серед кращих блокчейн-проектів: Proof-of-Work, Proof-of-Stake, Delegated Proof-of-Stake, Proof-of-Authority. Алгоритм Proof-of-Work (PoW) вимагає для підтвердження транзакцій виконання досить складної обчислювальної роботи, наступний блок створює той вузол, який першим закінчив розрахунки. Алгоритм є дуже енергозатратним, має низьку швидкість та погану масштабованість і не підходить для масового повсякденного використання. Proof-of-Stake (PoS) – популярний алгоритм консенсусу, у якому творцем наступного блоку в ланцюжку вибирається вузол, що має більший баланс – кількість ресурсів. Цей алгоритм частково виправляє недоліки алгоритму PoW – низьку швидкість та погану масштабованість, однак потребує витрат та технічних знань. Проведений аналіз дозволив виявити, що при плануванні ланцюгів

поставок високу ефективність у логістиці показав алгоритм Proof-of-Authority (PoA), основою якого є система репутації, коли кожен користувач блокчейну має ідентифікатор, який відображає його репутацію. За роботу мережі відповідає фіксоване число валідаторів – ті, хто підтвердив свою особистість.

На ринку існує значне число блокчейн-платформ, які розрізняються застосовуваними протоколами шифрування, алгоритмами консенсусу, функціональними можливостями [2]. Серед блокчейн-платформ, які використовують у сфері логістики, було виявлено Ethereum, Ripple, Hyperledger Sawtooth, Hyperledger Fabric, IBM Blockchain Platform, Microsoft Azure Blockchain Platform, Quorum. Для розробки програмного застосунку підтримки логістичної діяльності доцільно обрати платформи Microsoft Azure Blockchain Platform та Quorum, оскільки вони підтримують алгоритм консенсусу PoA.

Аналіз досвіду використання технології блокчейн у логістичній сфері дозволив виявити такі основні переваги, які можна отримати від її впровадження [3]:

- забезпечення відповідності та прозорості, яка виключає шахрайство й можливість доступу до конфіденційної інформації стороннім особам;

- забезпечення конфіденційності та безпеки транзакцій завдяки використанню ключа шифрування, який видно тільки її учасникам;

- забезпечення доступності до інформації та швидкого її знаходження й синхронізації, скорочення часу взаємодії та зменшення затримки в операціях;

- збільшення довіри клієнта завдяки наданню достовірну інформацію щодо продукту на всіх етапах його життєвого циклу, забезпечення можливості отримати відгук від споживачів у реальному часі й вчасного виявити та виправити помилки;

- забезпечення можливості програмування й автоматизації бізнес-процесів у обраній сфері, перенесення усієї інформації про процеси доставки у цифровий формат.

Застосування блокчейн технології надає принципово новий підхід до управління логістичною діяльністю, використання складної системи шифрування підвищує надійність і безпеку роботи усіх учасників ланцюга поставок, забезпечуючи наскрізний надійний моніторинг фінансових та інформаційних потоків між ними.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Савельев И. Е. Технология Blockchain и ее применение. Прикладная информатика. *Journal of Applied Informatics*. 2016. Т. 11. № 6(66). С.19-24.

2. Морозова Ю. А. Программные решения блокчейн в логистике и управлении цепями поставок. *Информационное общество*, 2019. № 6. С 49-58.

3. Кудирко О. В. Інновації в логістиці: перспективи використання технології блокчейн у ланцюгах поставок. Науковий вісник Ужгородського національного університету, 2017. Вип. 15, ч. 1. С. 158–162.

УДК 004

Зіменіна Ю. М., Рудніченко М. Д.

*Національний університет «Одеська політехніка»,
м. Одеса, Україна*

АНАЛІЗ ІСНУЮЧИХ ПРОГРАМНИХ ПРОДУКТІВ ДЛЯ УПРАВЛІННЯ ІНВЕСТИЦІЯМИ ФІЗИЧНОЇ ОСОБИ

Питання ефективного використання накопичених коштів дедалі частіше виникає у світі. У наш час перед звичайними користувачами стоїть не лише питання як заробити фінансові кошти, але й як ефективно реалізувати їх надлишок, що залишився після задоволення первинних потреб. І тут на допомогу приходять давно відомий інструмент – інвестування.

Інвестування – це вкладення коштів у різні матеріальні і нематеріальні активи заради їх примноження.

Розвиток інформаційних технологій дозволив зробити інвестування більш доступним для пересічних людей. Це було зумовлено розробкою різних фінансових застосунків управління грошовими коштами, зокрема й управління інвестиціями.

Фінансовий застосунок є програмним забезпеченням та/або хмарним інтернет-сервісом, що дозволяє рахувати, розподіляти, планувати та контролювати власні кошти, аналізувати стан власних фінансів та активів. Дані програми розробляються як для великих фінансових корпорацій, так і для приватних користувачів. Як правило, програми, розроблені для приватних користувачів, пропонують безкоштовний набір стандартних функцій обліку та розподілу коштів користувача, але також найчастіше передбачається платний доступ до ширшого спектру інструментів програми.

Програми управління інвестиціями є зручним інструментів управління грошовими коштами, їх правильного інвестування і примноження.

Це програмне забезпечення зазвичай розробляється на замовлення компаній фінансового та банківського секторів для спрощення доступу своїх клієнтів до фінансового ринку.

Більшість програм управління мають стандартний набір функцій таких як:

- відстеження доходів та витрат;
- котирування активів;
- відкриття брокерського рахунку;
- аналітичні графіки;
- онлайн консультант;
- пошук за заданими критеріями.

Далі розглянемо найбільш популярні фінансові програми, що часто використовуються на практиці.

1. Bloomberg: Market & Financial News. Особливістю програми Bloomberg: Market & Financial News є можливість створення особистого портфеля акцій та моніторинг свого персонального портфоліо. Також у застосунку реалізовано функцію отримання персональних оповіщень про позиції на біржі та загальну фінансову, економічну та корпоративну інформацію, актуальну для фінансових фахівців та відповідну потребам та запитам глобального бізнесу.

2. Yahoo Finance. Додаток призначений для моніторингу фінансового сектора та ринку економіки. Воно надає користувачеві актуальні курси валют, котирування акцій відомих світових компаній, вартість облігацій, ф'ючерси, світові індекси, вартість сировинних товарів тощо. Особливістю даної програми є можливість персоналізації новин, індивідуальне настроювання та редактура особистого вебпортфоліо, порівняння котирувань за допомогою інтерактивних графіків та діаграм.

3. Investing. com. Додаток, розроблений фахівцями однойменного порталу, є потужним багатофункціональним інструментом. Функціонал даної утиліти включає безліч глобальних та локальних фінансових інструментів та відмінно підходить для інвесторів, трейдерів та бізнесменів. З особливостей програми слід виділити його фокусування на ринку цифрової валюти та операціях з криптовалютами.

4. Intelinvest. Додаток є інтелектуальною заміною стандартних брокерських звітів і таблиць формату Excel. У цьому застосунку реалізовані функції з обліку угод щодо акцій, облігацій, інвестицій (ф'ючерси, ППФи, банківські депозити). Особливістю програми є імпорт

брокерських звітів 19 брокерів (Тінккофф, Відкриття, Альфа-Дірект тощо).

5. *MSN Money - Stock Quotes & News*. Інформаційна програма від корпорації Microsoft надає користувачеві актуальні новини світового фінансового ринку. Функціонал програми дозволяє не тільки отримувати актуальну інформацію про курси валют, вартість сировинних товарів, індексів, даних фондових бірж, проводити різні фінансові розрахунки (у тому числі і за іпотеками), а й розвивати фінансову грамотність користувача.

У програмі реалізована функція персоналізованого налаштування списків відстеження. Дані списки дозволяють здійснювати моніторинг користувачів акцій, котирувань, інвестиційних фондів, отримувати діаграми, тенденції зростання і падіння, а також отримувати корисні рекомендації та докладну інформацію про компанії, що цікавлять.

Більшість застосунків, призначених для управління інвестиціями фізичної особи, є інформаційними порталами з можливістю відстеження певних видів валют, акцій, отримання котирувань і світових індексів, а також створення одного і в окремих випадках кількох інвестиційних портфельів. Слід зазначити, що робота подібних застосунків часто територіально обмежена і не призначена для міжнародного використання. А також відсутня можливість підключення рахунків, відкритих у різних банках.

Більшість застосунків не мають графічного представлення інформації, що дозволяє відслідковувати динаміку зростання та падіння валют, котирувань, вартості акцій та ін. А також не мають інтуїтивно зрозумілого для непрофесіонала аналітичного матеріалу за наявними активами та пасивами.

Варто відзначити, що більшість програмних продуктів у сфері управління інвестиціями орієнтовані на представників бізнесу, брокерів та трейдерів, і меншою мірою орієнтовані на використання приватним інвестором.

В результаті проведеного аналізу існуючих програмних продуктів для управління інвестиціями фізичної особи були виявлені основні недоліки та переваги, які можуть бути враховані за подальшої розробки програмного продукту для управління інвестиціями фізичної особи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Буценко Е.В. Анализ программных средств управления инвестиционными проектами. Управленец. 2017. №65. С. 72-77.

2. Лукасевич И.Я. Инвестиции. М.: Вузовский учебник, Инфра-М, 2019. 412 с.

3. Теплова Т.В. Инвестиции: теория и практика. М.: Юрайт, 2019. 782 с

УДК 004.42

Кутняк В. Є., Кондратенко Г. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ОПТИМІЗАЦІЯ UNITY UI ІНТЕРФЕЙСІВ

В оптимізації користувацького інтерфейсу немає універсальних правил, що працюють у будь-якій ситуації. Все зводиться до пошуку балансу між складовими рушія Unity, а саме: вартістю батчингу та кількістю викликів для відтворення.

По-перше необхідно виділити чотири основні проблеми, що спонукають нас звернутися до проблеми оптимізації інтерфейсу:

- занадто велике завантаження GPU (надмірне навантаження на відтворення зображення);
- занадто багато навантаження на процесор при зміні/оновленні полотна;
- занадто багато змінних елементів, що ведуть до перебудови полотна;
- занадто велике навантаження на процесор при генерації мешів (зазвичай пов'язане з текстом).

Основним елементом інтерфейсу користувача Unity є полотно (Canvas). Він несе відповідальність за генерацію, сортування та відтворення сітки дочірніх елементів інтерфейсу. Всі елементи інтерфейсу повинні бути дочірніми для будь-якого полотна, інакше вони не будуть відображатися на екрані. Відтворення відбувається від далекого до найбільш близького об'єкта від камери (back-to-front) у черзі прозорості (Transparent queue) з альфа-змішуванням. Отже, коректне налаштування полотна є першим етапом оптимізації інтерфейсу.

Наступним етапом є підготовка графічних матеріалів проекту задля подальшого заощадження значної кількості ресурсів девайсу. Саме для таких цілей у Unity існує функціонал Sprite Atlas (атлас спрайтів), що поєднує в собі кілька текстур. Атласи дозволяють зменшити кількість відтворень (draw call) та підвищення продуктивності.

Сам функціонал є простим але чи найнеобхіднішим для

кожного проєкту, що наповнюється графікою. Слід мати на увазі, що навіть якщо на сцені використовуються один або декілька спрайтів із атласу, то атлас все одно буде повністю завантажено. Тому не має сенсу намагатися ввести всі зображення в один масивний атлас, який на пристроях з невеликою операційною пам'яттю може зайняти велику частину. Краще розділити на деякі менші частини.

Задля демонстрації ефективності цього інструменту нижче наведено порівняльну таблицю. Було проведено тест ефективності за допомогою Unity Profiler, що надав наступні результати.

Таблиця 1 – Порівняння основних показників продуктивності до і після об'єднання зображень у атлас

	До	Після
Затримка	8	4
Мінімальні кількість кадрів за секунду (FPS)	60	250

Не менш важливим етапом оптимізації є робота з текстом. Текст в інтерфейсі Unity складається з сіток, в яких кожен символ створює свій квад (quad). Сітка реструктується кожного разу, коли змінюється значення тексту. Крім того, реструктуризація відбувається, якщо текстовий компонент або його parent-об'єкт був виключений і повторно включений.

За замовчуванням шрифти в Unity додаються як динамічні. Для кожного динамічного шрифту, який використовується в текстовому компоненті на сцені, створюється його атлас. Цей атлас включає лише використані символи. Для кожного символу, що відрізняється за розміром або стилем, його образ буде створено в Атласі.

Використання резервних шрифтів (Fallback Fonts), перерахованих у полі імен шрифтів у параметрах шрифту, призводить до збільшення використовуваної пам'яті. Це особливо помітно на піктограмних шрифтах.

Також у текстовому компоненті є інструментарій, що зможе допомогти зекономити час але витратити ресурси системи. Наприклад Best Fit. Цей варіант призводить до швидкого переповнення атласу і викликає його реструктуризацію. Best Fit ігнорує параметри розміру шрифту та намагається відповідати тексту до прямокутника текстового компонента.

Велика кількість шрифтів з різними локалізаціями або великим шрифтовим атласом може займати багато пам'яті. Тому краще використовувати попереднє завантаження, необхідне для певної локалізації шрифтів.

Тема дійсно дуже велика – способи розглянуті вище не вимагають втручання з коду. Оптимізація завжди про вибір, про пошук балансу між різними, а іноді навіть і суперечливими один одному рішеннями. Немає універсальної інструкції на «всі випадки життя», не можна брати кожен із зазначених методів без попереднього аналізу та впевненості, що ваші дії гарантовано призведуть до позитивного результату.

Не треба негайно сплющувати усю графіку і робити дуже мінімалістичний але оптимізований сірий квадрат на сірому фоні, відмовившись від приємної візуальної складової інтерфейсу. Тим не менш, кожен раз, поглиблюючись, з'являється можливість уникнути збільшення кількості проблем навіть на початку. Ситуації варіюються від проєкту до проєкту, від мети до мети, але загальні принципи завжди залишаються однаковими.

Розробники Unity, звичайно, також не сидять. Офіційні гайди на Unity Learn поповнюються хорошими статтями щодо оптимізації інтерфейсу, а нові версії рушія повільно, але виправляють проблеми старих. Судячи з останніх анонсів, Unity вже готує абсолютно новий уніфікований інструмент для редагування користувацьких інтерфейсів - на перший погляд, дуже нагадує CSS для веброзробки. Нова система обіцяє повністю переробляти інтерфейс і значно підвищити продуктивність.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Кандибін М. А., Кінчак О. В., Оптимізація Unity UI без коду 2019. № 8 (115). С. 305–310.
2. Joe Hocking, Unity in Action: Multiplatform Game Development in C# with Unity 5 2015. 43(1):89-96.
3. Sue Blackman, Unity for Absolute Beginners 2014. 40(12):2872-8.

УДК 004.042

Пушняк І. А., Калініна І. О.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ОБРОБКИ ІНФОРМАЦІЇ ПО ЯКОСТІ ФІЛЬТРАЦІЇ ВОДИ

Інтелектуальна система — це технічна або програмна система, здатна вирішувати завдання, що традиційно вважаються творчими, що належать конкретній предметній області, знання про яку зберігаються в

пам'яті такої системи. Структура інтелектуальної системи включає три основні блоки: базу знань, механізм виведення рішень та інтерфейс.

Розроблена ІС відноситься до інтелектуальної інформаційної системи і виконує рекламно-інформаційні функції (інтерактивні банери). ІС (інтелектуальна інформаційна система) призначена для збору інформації клієнтів для подальшого її аналізу.

База даних яка поступає від клієнтів складається з:

1. Температура води;
2. Вхідний TDS;
3. Вихідний TDS;
4. Ресурс картриджу для води у літрах та днях;
5. Тиск;
6. Інформація про протік води з фільтра.

Розроблена система (інтерфейс) це вебзастосунок, тому не потребує встановлення зайвих програм. Для отримання доступу потрібно лише придбати продукцію та зареєструвати код виробу. Вся необхідна інформація буде доступна на окремій сторінці.

ІС має наступний функціонал:

1. Збір даних від клієнтів по якості фільтрації води;
2. Створення звітів для подальшого аналізу;
3. Сповіщення клієнтів про неполадки фільтру;
4. Актуальне зображення інформації показників фільтру.

В силу того, що питна вода це харчовий продукт, вироблений системою водопостачання для щоденного споживання і до неї ставлять достатньо високі вимоги відносно її безпеки і якості для здоров'я людини, її якість необхідно контролювати та слідкувати за тим, щоб вода відповідала всім вимогам. Так як все більше і більше виробів, галузей переходять на автоматизований вид керування, дана інтелектуальна інформаційна система є досить актуальною, так як допомагає стежити за якістю своєї питної води не виходячи з дому. Вся інформація структурована та візуальна зображена на одній сторінці, це полегшує її сприйняття та розуміння.

Дана система є корисною не лише для користувачів, а й для виробників. Вона допомагає слідкувати за своїми виробами та оперативно реагувати на різні види задач.

Також система є корисною зі сторони маркетингу, так як дозволяє відстежувати ресурси картриджів та своєчасно пропонувати клієнтам їх заміну.

Система є масштабованою, так як дозволяє додавати безліч інших даних клієнтів, дозволяє розширювати функціонал, додавати різні методи обробки інформації та її аналізу.

ШТУЧНИЙ ІНТЕЛЕКТ ТА ІНТЕРНЕТ РЕЧЕЙ

Інтернет речей та штучний інтелект разом з машинним навчанням є прогресивними та мультизастосованими інформаційними технологіями, що набувають популярності в застосування в різних сферах діяльності людини. Тому для ще більшого вдосконалення буде ефективно поєднати ці технології.

Інтернет речей (IoT) – це технологія, яка допомагає нам переосмислити повсякденне життя, але штучний інтелект (AI) є справжньою рушійною силою повного потенціалу IoT.

За прогнозами, до 2025 року в усьому світі буде 42 мільярди пристроїв, підключених до Інтернету речей. Цілком природно, що в міру зростання кількості цих пристроїв будуть рости і масиви даних. Ось де втручається AI — надає свої можливості навчання підключенню IoT.

Інтернет речей уповноважений трьома ключовими новими технологіями:

1. Штучний інтелект (ШІ)

Програмовані функції та системи, які дозволяють пристроям навчатися, міркувати й обробляти інформацію, як люди.

2. Мережі 5G

Мобільні мережі п'ятого покоління з високошвидкісною, майже нульовою затримкою для обробки даних у реальному часі.

3. Великі дані

Величезні обсяги даних обробляються з численних джерел, підключених до Інтернету.

Разом ці взаємопов'язані пристрої змінюють наш спосіб взаємодії з нашими пристроями вдома та на роботі, створюючи при цьому AIoT («штучний інтелект речей»).

Даний симбіоз технологій може застосовуватися не тільки в промисловості, але й у такій складній та точній сфері як медицина. Вже сьогодні є пристрої для людей хворих на діабет, що допомагають визначати рівень цукру в крові та відображати його через застосунок на мобільному пристрої.

Схожа розроблена система для створення аналізу електрокардіограми (ЕКГ) та виявлення серцевих захворювань Система

включає в себе апаратне забезпечення на основі IoT, користувальницький інтерфейс програми для смарт-пристроїв (APP), хмарну базу даних і платформу AI для виявлення серцевих захворювань. Апаратне забезпечення на основі IoT, носимий патч ЕКГ, що включає аналоговий інтерфейсний контур і модуль Bluetooth, може виявляти сигнали ЕКГ. Додаток на розумних пристроях може не тільки відображати ЕКГ-сигнали користувачів у режимі реального часу, але й миттєво позначати незвичайні сигнали та виявляти захворювання в режимі реального часу.

Також штучний інтелект допоможе вирішити таку важливу та складну проблему як безпечність використання. На передньому краї кібербезпеки знаходиться штучний інтелект (AI), який використовується для розробки складних алгоритмів для захисту мереж і систем, у тому числі системи IoT.

За статистикою було виявлено, що AI та IoT є найкращими технологіями, у які компанії найбільше інвестують, щоб підвищити ефективність та забезпечити конкурентну перевагу. На графіку нижче показано деталі:

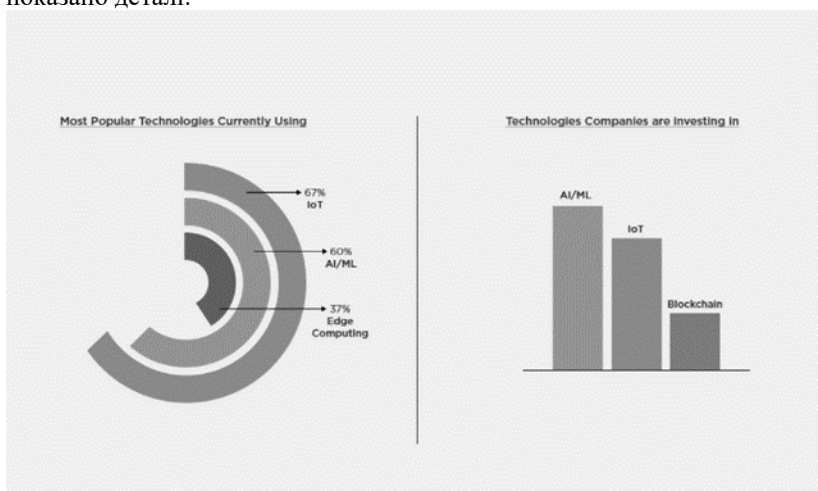


Рисунок 1 – Діаграма популярності інвестицій

Розглянувши переваги синтезу 3 технологій, ми отримуємо потужний популярний та вигідний інструмент для вдосконалення вже існуючих продуктів, розширення сфер застосування та вирішення цілої низки проблем. Представленні основні положення та ключові аспекти роботи та їх сильні сторони. Також отримання підвищення рівня

безпеки робить можливим використання у вразливих від кібератак речах.

УДК 004.725: 004.056

*Ткаченко М. П., Кучеренко Є. А., Журавська І. М.
Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АВТОМАТИЗОВАНА СИСТЕМА ВИЯВЛЕННЯ БОТ-АТАК НА ОСНОВІ АНАЛІЗУ КОРИСТУВАЦЬКИХ ПРОФІЛІВ У СОЦІАЛЬНИХ МЕРЕЖАХ

Починаючи з 2004 року – з появи та розвитку соціальних інтернет-мереж, таких як: «Facebook» (2004 р.), «Twitter» (2006 р.), «Instagram» (2010 р.) та інших – виникла проблема соціально-технічного характеру: неможливість забезпечити технічний захист від деяких видів атак, що відносяться до соціального аспекту функціонування мережі, нп., flame- та flood-атак [1, 2].

Атаки flame.&flood – це категорія кібератак, що ставлять на меті виконати соціальний та психологічний тиск на обрану ціль. Основою засобом проведення flood-атаки є багаторазове та масове відправлення повідомлень, які можуть мати безглуздий, агресивний або ще будь-який характер, що призводить до потрібної поведінки жертви та у решті-решт до мети, яку обрав зловмисник. Це може бути, наприклад, доведення цілі до стану психозу або іншого психічного розладу у легкій формі з метою шантажу або спричинення шкоди ще й фізичному здоров'ю жертви – наприклад, доведення до самогубства.

Flame-атаки, на відміну від flood-атак, можуть та найчастіше є більш короткочасними, бо мають інший механізм соціального впливу на обрану ціль. Головна ціль flame-атаки – виявлення «гострих» політичних та/або соціальних тем, а також соціальної групи (друзів, родичів, знайомих, колег тощо), до якої входить жертва даного виду атаки. Крім того, метою flame-атаки є формування двох або більше суперечливих думок або складної – з соціально-політичного боку – тези чи висловлювання, що призводить до бурхливого обговорення обраної теми серед учасників соціальної групи. В результаті це може призвести до розпаду такої групи або погіршенню відносин серед її учасників. Варто зазначити, що на відміну від flood-атаки, основною метою flame-

атаки є не формування певної думки за допомогою її масового повторення від «різних» джерел. Її головне призначення – це саме першочергове підняття «гострої» теми з її подальшим обговоренням самими учасниками обраної соціальної групи.

Найчастіше описані вище атаки виконуються за допомогою соціальних інтернет-мереж та системи бот-програм – спеціального виду автоматизованого програмного забезпечення, яке імітує соціальну активність реальних користувачів з першочерговою метою викликати довіру та емпатію інших учасників. Такий підхід призводить до більш зацікавленого обговорення теми з їх боку та формування певної думки за допомогою її циклічного повторення від різних екземплярів бот-кластеру.

Одним із способів протидії flame.&flood атакам є автоматизований аналіз профілів соціальних інтернет-мереж за заданим набором параметрів, Таким чином можливо виявлення потенційних учасників бот-системи та попередження інших учасників про виявлену загрозу маніпулювання їх соціальною поведінкою [3].

Для забезпечення можливості автоматизованого аналізу профілів користувачів необхідно отримати інформацію заданого набору профілів для їх подальшого об'єднання у кластер та розбиття на підкластери на підставі заданих параметрів.

У розроблюваній автоматизованій системі виявлення бот-атак для отримання даних користувацьких профілів запропоновано використовувати «парсинг» – автоматизований аналіз сирцевого коду вебсторінки. За такою технологією можливо відокремлення певних елементів вебсторінки та отримання їх значень. Наприклад, це може бути пошук HTML-тегу <title> та отримання назви вебсторінки або будь-якого іншого тегу та елемента вебсторінки і його значення. Також передбачено використання інструментів імітації та автоматизації користувацьких дій, таких як: натискання на віртуальні кнопки та перехід за посиланням на інші сторінки.

За допомогою парсингу та автоматизованого обходу заданого дерева сторінок – з урахуванням глибини та зв'язності дерева (кількості нащадків для певного вузла) можливо здійснити аналіз та прогнозування того, наскільки інформація з профіля, що є вузлом побудованого дерева, є достовірною та не відноситься до елементів бот-системи.

Для виконання поставленої мети використовуються нижчезазначені технології та методи.

Технології:

- а) мова програмування Python;
- б) додаткові бібліотеки мови програмування Python:
 - 1) *Selenium* – бібліотека для автоматизації імітування взаємодії з графічним користувацьким інтерфейсом (GUI) та його елементами;
 - 2) *BeautifulSoup4* – бібліотека для парсингу вмісту HTML та XML-файлів;
 - 3) *Requests* – бібліотека для виконання HTTP- та HTTPS-запитів;
 - 4) *NetworkX* та *Matplotlib* – бібліотеки для візуалізації графів, дерев та довільних підготовлених масивів даних.

Методи:

- а) аналіз часових рядів;
- б) кластеризація та кластерний аналіз;
- в) засоби та алгоритми Data Mining – алгоритми збору даних у часовому полі, їх аналіз та пошук шаблонів (патернів) та взаємозв'язків між окремими частинами даних. Спроба використання знайдених патернів на інших наборах даних для пошуку прихованих взаємозв'язків;
- г) засоби та алгоритми Data Analysis – алгоритми роботи з даними, які на відміну від Data Mining, не призначені для пошуку скритих патернів та можуть використовуватись на неструктурованих або погано/недостатньо структурованих наборах вхідних даних. Але основною задачею цих алгоритмів як раз і є процес впорядкування та організації «сирих» даних для отримання корисної інформації у більш зручному вигляді.

Таким чином, аналіз користувацьких профілів у соціальних мережах може бути ефективно використаний для виявлення бот-атак за допомогою розробленої автоматизованої системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Mayer E. Spammers show they give a fig about Instagram, launch fruit-based attack. Publ. June 29, 2013. URL: <https://www.cnet.com/tech/services-and-software/spammers-show-they-give-a-fig-about-instagram-launch-fruit-based-attack/> (Last accessed: 29.12.2021).

2. Литвиненко Ю. Крупнейшая хакерская атака на Twitter. Опубл. 16 июля 2020 г. URL: <https://www.vedomosti.ru/technology/articles/2020/07/16/834712-vzлом-milliardrov-v-twitter> (дата обращения: 29.12.2021).

3. Green J. S. Cyber security: An introduction for non-technical managers. New York, NY, United States : Routledge (Taylor & Francis Group), 2016. 264 p.

УДК 004.89, 004.93

Шаутін О. В., Кулаковська І. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АДМІНІСТРУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З РЕАЛІЗАЦІЮ АЛГОРИТМУ РОЗПІЗНАВАННЯ ОБЛИЧ

Нейронна мережа – це ланцюг із нейронів, які з'єднані між собою синапсами. Структура штучної нейронної мережі була натхнена біологією. В наслідок цього, машина може аналізувати, запам'ятовувати та навіть відтворювати різну інформацію із своєї пам'яті. Іншими словами, штучна нейронна мережа, це відтворення міркувань людського мозку, який містить мільйони-мільярди нейронів та ще більше синапсів, які передають інформацію у вигляді електричних імпульсів.

Нейронні мережі використовуються для вирішення задач, які потребують аналізу даних подібних до міркувань, які відбуваються у людському мозку, такі сфери застосувань представлені в [1]. Найпоширенішими задачами, де вони використовуються, є:

- класифікація – сортування даних по параметрам. Наприклад, на вході завантажуються набір даних про людей, і треба вирішити, кому з них можна надати кредит. Цю задачу може виконати нейронна мережа, аналізуючи таку інформація як: вік, платоспроможність, кредитна історія і таке інше;

- прогнозування – можливість передбачати наступний крок чи подію. Наприклад, зріст чи падіння акцій, базуючись на аналізі стану фондового ринку;

- розпізнавання – найпоширеніше використання нейронних мереж. Наприклад, в камерах телефонів, коли розпізнається та виділяється власник, положення обличчя та інше.

Використання спеціально навченої нейронної мережі може автоматизувати більшість рутинних процесів, наприклад, можна під'єднати нейронну мережу навчену розпізнавати людей по обличчю до відеокamera на вокзалі, та ідентифікувати людей, яких розшукують правові служби [2]. Або спостерігати, чи всі працівники дотримуються противірусних норм.

В статті [3] розглядаються особливості розпізнавання облич, аналізується розуміння алгоритму LBPН. Існують різні типи алгоритмів розпізнавання обличчя, наприклад: власні обличчя (1991); гістограми локальних двійкових шаблонів (LBPH) (1996); масштабно інваріантне перетворення ознак (SIFT) (1999); прискорення надійних функцій (SURF) (2006).

В посиланні [2] інформація KanKan, це дочірня компанія Remark Holding, яка оголосила про укладання семизначного контракту з одним із найбільших державних підприємств Китаю, щоб надати муніципальному агентству охорони здоров'я Шанхаю можливість розпізнавання осіб та об'єктів. Їхня технологія штучного інтелекту в даний час використовується в 200 ресторанах, але скоро буде розширена до 2000 об'єктів. Камери на кухні або у закладі громадського харчування стежать за тим, щоб люди одягали маски або засоби захисту волосся, коли цього вимагають правила техніки безпеки. Порушення можуть бути виявлені та виправлені практично в реальному часі.

KanKan використовує розпізнавання об'єктів штучним інтелектом для багатьох програм: від боротьби з конфіденційністю IP до автоматичної перевірки номерних знаків.

Було створено застосунок, який розпізнає обличчя опираючись на існуючу базу даних. У своїй реалізації система відносно дешева, тому що її працездатність залежить від наявності відеокамери, та потужності пристрою, на якому штучна нейронна мережа буде проводити обчислення.

У [4] представлено детальний опис методів LBP та їх варіантів, а також надано огляд того, як текстурні методи можна використовувати для вирішення різних видів проблем комп'ютерного зору (див. рис. 1).

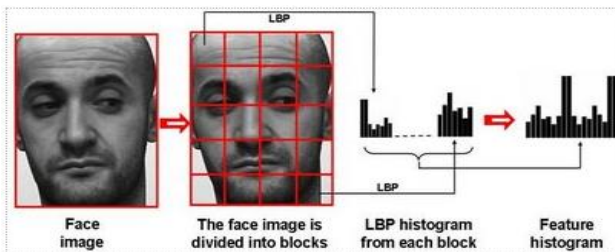
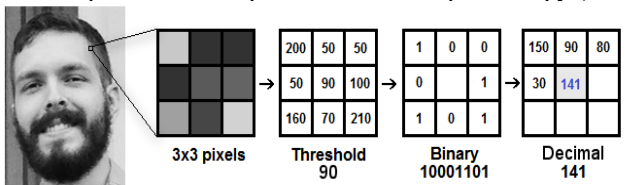


Рисунок 1 – Приклад роботи використаного алгоритму LBPН

У власній програмній реалізації було використано алгоритм LBPН, який позначає пікселі зображення, обмежуючи околиці кожного пікселя, і розглядає результат як двійкове число (див. рис. 2). Результат роботи програми наведено на рисунку 2.



Рисунок 2 – Результат роботи розробленої системи

Система розпізнала обличчя з точністю 72%. На точність розпізнавання впливає роздільна здатність камери, яка знімає обличчя.

Таким чином, розглянуто дослідження сучасного стану задачі із застосування штучного інтелекту для розпізнавань облич; проведено аналіз сфери застосувань та обрано технологію та алгоритм для вирішення задачі розробки інформаційної системи з реалізацією алгоритму розпізнавання облич; виконана програмна реалізація дослідженої інформаційної системи.

Практична цінність дослідження - розроблена інформаційна система дає можливість контролювати доступ до підприємства з міркувань безпеки за допомогою неінвазійного способу ідентифікації особистості.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. How face perception unfolds over time | *Nature Communications* : вебсайт. URL: <https://www.nature.com/articles/s41467-019-09239-1> (дата звернення 14.01.2022).

2. AI in China : police cars scanned faces in search of Criminals - *SEO China Agency* : вебсайт. URL: <https://seoagencychina.com/ai-in-china-police-cars-scanned-faces-in-search-of-criminals/> (дата звернення 14.01.2022).

3. Face Recognition: Understanding LBPH Algorithm | by Kelvin Salton do Prado | *Towards Data Science* : вебсайт. URL: <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b> дата звернення 14.01.2022).

4. Pietikäinen, M., Hadid, A., Zhao, G. and Ahonen, T. (2011), *Computer Vision Using Local Binary Patterns*, Springer. <http://www.springer.com/mathematics/book/978-0-85729-747-1>.

UDC 004.582

A. Volkov, O. Blazhko

National university «Odessa Polytechnic», Odesa, Ukraine

DATABASE OF HUMAN PHYSICAL EXERCISES IN VIRTUAL REALITY

Nowadays computer games began to be actively used as simulators of human physical activity for therapeutic purposes. Hardware and software components are used for recognizing patient movements to physically and mentally improve the patient's condition [1]. Unfortunately, these games are still part of the scientific programs of major medical centers and are not available to ordinary therapists. Therefore, authors of this work have previously created several similar games to control human movement with an infrared sensor MS Kinect and Unity3D [2]. The first game "Collector" motivates a player to quickly collect coins besides him by moving his arms in different directions. The second game "Flappy Bird" motivates a player to perform monotonous exercises so the bird flaps and does not crash inside the maze. But the description of these exercises was programmed in the program code of the game, not allowing them to be edited by an expert-analyst without presence of a programmer. Therefore, the goal of this work is to develop a relational database of human exercises by using a software model of human skeleton joints in the infrared sensor MS Kinect.

The MS Kinect sensor version 2.0 uses skeleton with 26 point-joints. But the algorithms of created games used only 6 point-joints as was presented in figure 1. Figure 1 presents stages of data transformation during a human body recognition, skeleton`s joint recognition of the MS Kinect controller and controlling of signals for bird flapping in computer game, where bird must fly through the maze.

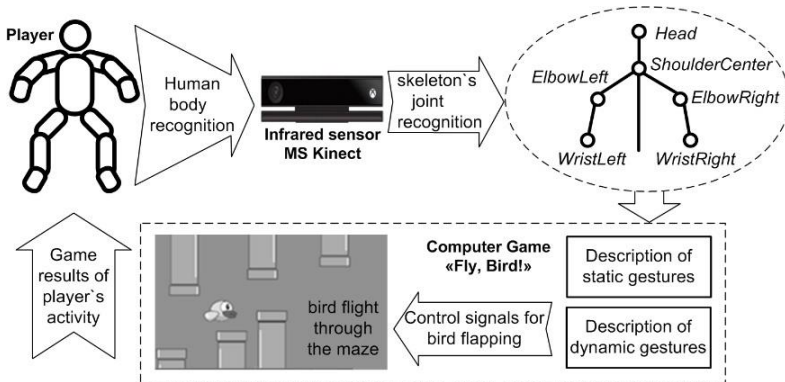


Figure 1 - Fragment of many point-joints in human skeleton of the MS Kinect controller

It is proposed to present a physical exercise in the form of an ordered set of dynamic gestures as pairs of static gestures. Static Gesture is determined by the condition of the ratio of the spatial location of skeleton's joints. UML-diagram of classes was designed to store the description of the physical exercises and was presented in figure 2:

- class «Training Program» use attributes: *Name* – name of training program, *Difficulty level* – difficulty of an exercise, for example with values as 1 – easy, 2 – medium, 3 – difficult;
- class «Exercise» use attributes: *Name* – name of exercise, *Velocity* - velocity of object in game as function with parameter «difficulty level»;
- class «Dynamic Gesture» use attributes: *Name* – name of gesture, *Dependency* – gesture link with other gestures that are performed in parallel to each other;
- class «Static Gesture» use attributes: *Order* – order number in sequence of gestures, *Skeleton1* – first point-joint of the two-joint static gesture; *Skeleton2* – second point-joint of two-joint static gesture; *Condition* – condition string with many comparisons of spatial coordinate values, for example, {=,>, <}; *Coordinate* – one of spatial coordinates { X, Y, Z}.

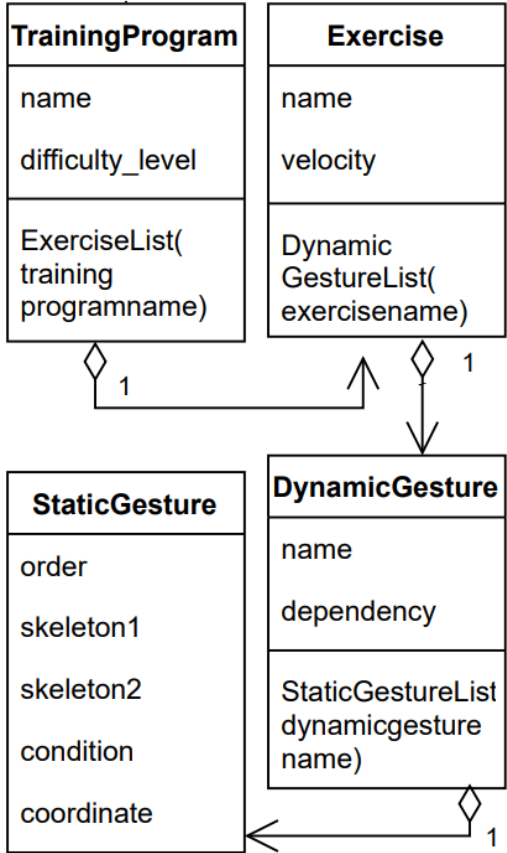


Figure 2 – UML-diagram of classes

Table «Exercise»

Id	Name	Velocity
1	Brass	1

Table «DynamicGesture»

Id	Name	Dependency	Exercise_id
1	Right dynamic gesture	2	1
2	Left dynamic gesture	1	1

Table «StaticGesture»

Dg_id	Order	Skeleton1	Skeleton2	Condition	Coordinate
1	1	ElbowRight	Head	=	Y
1	2	WristRight	ElbowRight	=	X
1	3	WristRight	ElbowRight	>	X
1	4	ElbowRight	SpineShoulder	<	Y
2	1	ElbowLeft	Head	=	Y
2	2	WristLeft	ElbowLeft	=	X
2	3	WristLeft	ElbowLeft	>	X
2	4	ElbowLeft	SpineShoulder	<	Y

Figure 3 – Fragment of content of relational database tables for swimming exercise in style of "Brass"

The several styles of swimming were analyzed in the form of animations presented on the site [3] to test the proposed gesture data storage structure. The fragment of content of relational database tables for swimming exercise in style of "Brass" was presented in figure 2(b). The database is managed by cloud DBMS MySQL. There were created several REST API-functions as methods of the classes by using PHP for communication between the game and the database.

REFERENCES

1. Paula Amorim, Beatriz Sousa Santos, Paulo Dias, Samuel Silva and Henrique Martins. "Serious Games for Stroke Telerehabilitation of Upper Limb - A Review for Future Research". *Int J Telerehabil*, 2020. doi: 10.5195/ijt.2020.6326
2. Demonstration of the gesture recognition program on the example of the game "Flappy Bird". URL : <https://bit.ly/3sXRzKX> (Active link – 29.12.2021)
3. Iskusstvo plavat. Blog, posvyashchenny plavaniyu. Sportivnyye stili plavaniya: ot krolya do batterflyaya. URL : <https://iplav.com/stili> (Active link – 29.12.2021)

Машинне навчання та штучний інтелект

УДК 004.42

Баришніков В. О., Мещанінов О. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПОСЛІДОВНІСТЬ РОЗРОБКИ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ РОЗУМНОГО БУДИНКУ

Вибір засобів розробки є важливим етапом при розробці програмної системи, так як від них залежить якість результуючого продукту. Розробка програмного рішення велася у середовищі розробки PyCharm, використовуючи Python 3.6. Python – об'єктно-орієнтована, інтерпретована мова програмування з динамічною типізацією. На сьогоднішній день дана мова використовується в багатьох галузях: автоматизація, наукові розрахунки, веброзробка, аналітика і машинне навчання, тощо [1]. Її вибір обумовлений тим, що на сьогоднішній день існує велика екосистема з бібліотек саме для машинного навчання що і є необхідним в рамках даної роботи. Для здійснення маніпуляцій з даними було використано бібліотеку Pandas. Це бібліотека з відкритим кодом яка пропонує ряд можливостей щодо більш зручної роботи з даними. Основною структурою даних є DataFrame яка підтримує індексацію. Певним чином дана структура є схожою на sql-таблиці. На рисунку 1 зображено приклад даної структури.

Відкрита бібліотека Scikit-learn містить реалізацію більшої кількості існуючих класичних методів машинного навчання регресії, кластеризації, класифікації, регресії, ансамблевих моделей, а також зменшення простору [2]. В даній роботі використана реалізація моделі випадкового лісу даної бібліотеки і методу опорних векторів. Слід зазначити що в бібліотеці цей метод реалізований на базі LibSVM – однією з найбільш ефективних та оптимальних реалізацій даного методу.

Для побудови нейронних мереж була використана бібліотека Keras. Keras – відкрита бібліотека для зручної та ефективної розробки моделей глибокого навчання створена на мові Python [3]. Для використання моделі Латентного Розміщення Діріхле була використана відкрита бібліотека з відповідною назвою – lda. Основна частина даної бібліотеки реалізована на C – тому є досить ефективною. Дана бібліотека надає подібній до Scikit-learn інтерфейс, що робить її досить зручною у використанні.

	Mountain	Height (m)	Range	Coordinates	Parent mountains	First ascent	Ascents bef. 2004	Failed attempts bef. 2004
0	Mount Everest / Sagarmatha / Chomolungma	8848	Mahalangur Himalaya	27°59'17"N 86°55'31"E	NaV	1953	>>145	121.0
1	K2 / Gogri / Godwin-Austen	8611	Baforo Karakoram	35°52'33"N 76°30'48"E	Mount Everest	1954	45	44.0
2	Kangchenjunga	8586	Kangchenjunga Himalaya	27°42'12"N 86°08'51"E	Mount Everest	1955	38	24.0
3	Lhotse	8516	Mahalangur Himalaya	27°57'42"N 86°55'59"E	Mount Everest	1956	26	26.0
4	Makalu	8485	Mahalangur Himalaya	27°53'22"N 87°05'20"E	Mount Everest	1955	45	52.0
5	Cho Oyu	8188	Mahalangur Himalaya	28°05'38"N 86°39'39"E	Mount Everest	1964	79	28.0
6	Dhaulagiri I	8167	Dhaulagiri Himalaya	28°11'48"N 83°29'33"E	K2	1960	51	39.0
7	Manaslu	8163	Manaslu Himalaya	28°33'00"N 84°33'35"E	Cho Oyu	1956	49	45.0
8	Nanga Parbat	8126	Nanga Parbat Himalaya	35°14'14"N 74°39'21"E	Dhaulagiri	1953	52	67.0
9	Annapurna I	8091	Annapurna Himalaya	28°35'44"N 83°49'13"E	Cho Oyu	1960	36	47.0

Рисунок 1 – Приклад датафрейму з бібліотеки DataFrame

Для створення користувацького інтерфейсу була використана бібліотека Dash – відкрита Python бібліотека для створення веб інтерфейсів. Як правило використовується для створення графічного інтерфейсу для застосунків у сфері аналізу даних та моделювання. При написанні застосунків не потрібно писати Javascript кодом або HTML так як дана бібліотека надає Python інтерфейсу до великої кількості вебкомпонентів.

Входом системи є дані з потоку сенсорів необхідного формату зібрані протягом певного періоду часу. Ці вхідні дані можна поділити на дані тренування і тестування моделі. В даному випадку передбачається що даними тренування є анотовані дані зібрані під час життя мешканця будинку, а даними тестування є ті, що будуть поступати у реальному часі. Однак дані тестування можуть мати теж анотації для можливості подальшого аналізу результатів класифікації.

Існуючі пакети в розробленій програмній системі приведені на рисунку 2.

В програмній системі існує чотири пакети: gui для користувацького інтерфейсу, controller для зв'язування моделі і користувацького інтерфейсу, модуль modeling містить реалізації моделей класифікації, а processing – реалізації сегментації та формування вектору ознак.

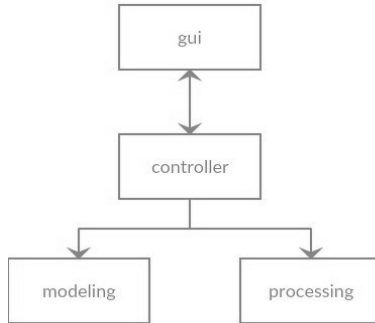


Рисунок 2 – Пакети програмної системи

Отже, у роботі описані обрані засоби розробки, проаналізовані їх особливості, які стали умовою вибору. Так були обрані саме ті засоби які забезпечують найбільшу ефективність і найбільш зручні умови використання результуючої програмної системи. Наведені пакети програмного забезпечення системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Beckmann M. A KNN Undersampling Approach for Data Balancing / M. Beckmann, N. Ebecken. *Journal of Intelligent Learning Systems and Applications* / M. Beckmann, N. Ebecken., 2016. С. 104–116.
2. Chen D. Pandas for Everyone: Python Data Analysis. Daniel Chen., 2018. 406 с.
3. Chih-Chung C. LIBSVM: A Library for Support Vector Machines / C. Chih-Chung, Chih-Jen., 2002. 39 с.

УДК 004.06

Борисов М. В., Гожий О. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПРОГНОЗУВАННЯ ЧИСТОЇ ВИХІДНОЇ ЕЛЕКТРОЕНЕРГІЇ ЕЛЕКТРОСТАНЦІЇ КОМБІНОВАНОГО ЦИКЛУ

На сьогоднішній день одною з найбільш актуальних проблем людства є проблема добування електроенергії, найпоширеніший спосіб якого, є використання природних джерел пального, що тягне за собою

дві інші величезні проблеми, це забруднення навколишнього середовища, та дефіцит природного пального такого як газ, нафта та вугілля. Оскільки людство може опинитися на порозі світової екологічної катастрофи, вже сьогодні необхідно приймати рішення щодо переходу до використання екологічно чистих та відновлювальних джерел енергії таких як гідроелектростанції, вітрові електростанції та інші.

Не всі країни можуть відмовитись від класичних видів добування електроенергії, деякі не мають виходу до водоїм, тому не можуть встановити гідроелектростанції. Деяким країнам клімат не дозволяє встановлювати сонячні електростанції. В Україні ця проблема навіть більш актуальна, тому що через непросту зовнішню політику, наша країна купує газ та вугілля за високою ціною. Близько 60% від усіх електростанцій у нашій країні становлять ТЕС, 12,4% - гідроелектростанції, а частка вітрових та сонячних електростанцій становить лише 0,5% від усієї електроенергії. Тому необхідно розглядати інші варіанти добування електроенергії.

Одним з таким варіантів є електростанція комбінованого циклу з використанням газотурбінної енергетичної системи. Дану станцію також називають станцією комбінованого циклу. В основі даного виробництва лежить класична газотурбінна установка, тобто при згорянні газової суміші пара крутить лопаті турбіни, яка виробляє електроенергію. ККД даної станції приблизно 33%, інші 67% це відходи.

Інше рішення це газокomboваний цикл в якому, пропонується додавати до газової турбіни пару. Після відпрацювання у газовій турбіні вихлопні гази залишаються ще досі гарячими (140°C), тому ці відходи можна використовувати для нагрівання води для парової турбіни (Рис. 1).

Використання електростанції комбінованого циклу підвищує ККД вдвічі, тобто має 68% електроенергії в порівнянні із 33% у звичайної електростанції. Оскільки використання цієї технології є дуже перспективним, було вирішено використати методи машинного навчання для прогнозування вихідної електроенергії реальної електростанції комбінованого циклу.

Методи машинного навчання розповсюджуються в сучасному світі та знаходять використання у все більшій кількості різних сфер життя, таких як:

- комп'ютерне бачення;
- розпізнавання мови;
- комп'ютерна лінгвістика;
- медична діагностика;

- біоінформатика;
- технічна діагностика;
- інформаційний пошук;
- фінансові застосунки.

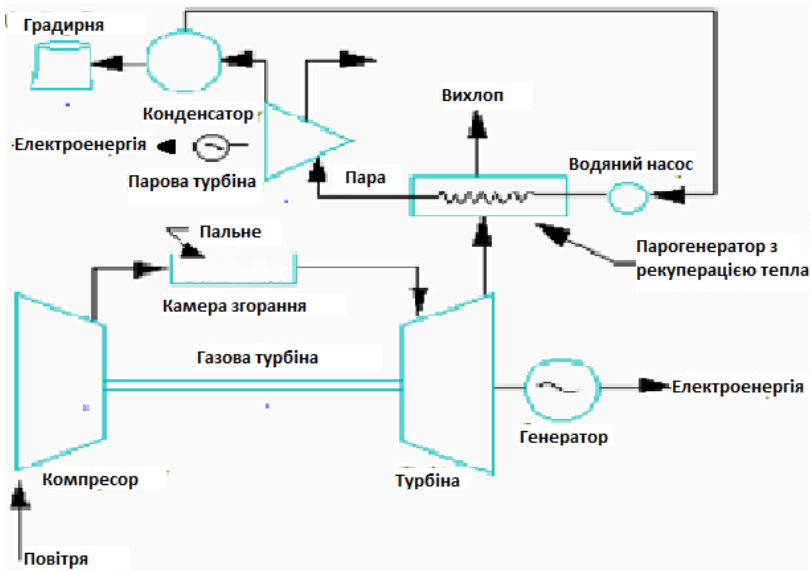


Рисунок 1 - Схема роботи газотурбінного циклу

Тому їх використання є дуже актуальним та потрібним. Раніше такі методи були складними та майже неможливими завданнями для комп'ютерів. Але зі збільшенням обчислювальної потужності та використанням методів машинного навчання з'явилися нові методи вирішення багатьох проблем, які раніше було складно або неможливо розв'язати.

Головна складність методів машинного навчання полягає в величезній кількості параметрів під час налаштуванні під конкретну задачу. Але висока адаптивність та швидкість роботи роблять методи машинного навчання дуже зручним інструментом.

Використовуючи відкриті джерела можна отримати датасети для створення суспільно-корисних та актуальних тем. Саме відкриті джерела дозволяють людям досліджувати реальні проблеми та запропонувати різні підходи для вирішення різних проблем. Прикладом такого набору даних є дослідження роботи електростанції комбінованого циклу протягом 2006-2011 років. Оскільки це реальні

дані, то використання класифікатору можливе в інформаційних системах на реальних електростанціях.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Combined Cycle Power Plant Data Set. Machine learning repository. *UCI Center for Machine Learning and Intelligent Systems* : вебсайт. URL: <http://archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant> (дата звернення: 04.03.2021).
2. Нікольский Ю.В., Пасічник В.В., Щербина Ю.М. Системи машинного навчання. Львів 2010.
3. Огляд електростанції комбінованого циклу. Енергія та енергія для всіх: вебсайт. URL: <https://crushtymks.com/uk/energy-and-power/708-an-overview-of-combined-cycle-power-plant.html>

УДК 004.85

Івченко І. О., Калініна І. О.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ЗАСТОСУВАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ БІОЛОГІЧНОЇ КЛАСИФІКАЦІЇ

Методів машинного навчання у сучасному світі щороку здобувають все більше областей для застосування. Одною із проблем, що може вирішити машинне навчання – прогнозування невідомих атрибутів живих організмів на основі відомих предикторів.

Розглядається задача класифікації моллюсків (морського вушка) за віком. Товстий внутрішній шар їх мушлі складається з перламутру. Вік морського вушка визначають шляхом розрізання шкаралупи через конус, фарбування та підрахунку кількості кілець за допомогою мікроскопа – нудне і трудомістке завдання.

Можливо передбачити вік морського вушка за допомогою наступних методів, на наборі даних, що взяті з публічної бібліотеки наборів даних [1].

Перш за все було використано метод опорних векторів (SVM), що є машинним алгоритмом, котрий навчається на прикладах та використовується для класифікації об'єктів. SVM відрізняється від інших гіперплощинних методів класифікації тим, що він дозволяє обирати оптимальне розташування гіперплощини. Гіперплощина обирається таким чином, щоб бути розташованою на максимальній

відстані від елементів кожного з класів, тобто посередині деякої зони, що відділяє між собою ці елементи.

Також було використано метод k -найближчих сусідів. Класифікація кожної текстової ознаки залежить від класу, до якого належить окрема навчальна ознака, що може мати невірну мітку чи взагалі бути нетиповою. Метод k NN при $k > 1$ є стійким. В поставленій задачі, при $k=3$ метод є найбільш ефективним.

На останок, метод випадкових дерев, що використовують для класифікації, регресії та інших завдань, працює за допомогою побудови численних дерев прийняття рішень під час тренування моделі й продукує моду для класів (класифікацій) або усереднений прогноз (регресія) побудованих дерев.

Отже, можна зробити висновок, що для вирішення кожної конкретної задачі потрібно підбирати комбінацію методів. Хоча методів штучного інтелекту існує дуже багато, але для різних задач потрібні різні підходи, тоді методи надають надзвичайні можливості в сфері навчання на даних та передбачені результатів. Ця сфера є і буде актуальною ще багато років.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Abalone Data Set. Machine Learning Repository: вебсайт. URL: <https://archive.ics.uci.edu/ml/datasets/abalone>.

УДК 004.81

Костира М. А., Кондратенко Ю. П.

*Чорноморський національний університет імені Петра Могили,
Миколаїв, Україна*

ДОСЛІДЖЕННЯ ВПЛИВУ АРХІТЕКТУР ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ НА ЕФЕКТИВНІСТЬ СЕГМЕНТАЦІЇ ОБ'ЄКТІВ

Комп'ютерний зір – науковий розділ, що базується навколо надання комп'ютерам розуміння на основі цифрових зображень чи відео [1]. З перспективи науковця, комп'ютерний зір має автоматизувати виконання задач, на які здатна візуальна система людини. Семантична сегментація зображень є одним із ключових розділів комп'ютерного зору, процес поділу цифрового зображення на ряд сегментів (об'єктів зображення). Сегментацію можна описати як класифікацію зображень на піксельному рівні [2]. Основною метою сегментації є спрощення, або зміна представлення зображення для подальшого аналізу. В основному, сегментація зображень

використовується для відокремлення об'єктів та границь (ліній, кривих) на зображенні. Іншими словами, це процес присвоєння ярликів до кожного пікселю так, що пікселі під одним ярликом мають схожі характеристики.

На абстрактному рівні, задачу можна сформулювати наступним чином: “Маючи RGB кольорове (висота \times ширина \times 3), або чорно-біле (висота \times ширина \times 1) зображення, створити сегментаційну карту, що показує належність кожного пікселю до певного класу (наприклад, людина, дерево, автомобіль)”.

Для досягнення цієї задачі буде використана нейронна мережа. Існує багато типів нейронних мереж, кожен з яких, як правило, спрямований на вирішення конкретної задачі. Що стосується задачі семантичної сегментації, дуже добре себе зарекомендували згорткові мережі. Цей тип мереж зазвичай використовується для аналізу та обробки зображень у цілому.

Згорткові мережі дуже схожі на звичайні, вони також складаються з нейронів, ваг, та зміщення (bias). Мережа у цілому так само виражається оцінювальною функцією, на вхід приймається зображення, а на виході - ймовірності належності до класу. Значною різницею є те, що згорткова мережа (у випадку семантичною сегментації), заздалегідь передбачає використання зображення у якості вхідних даних. Це дозволяє надати архітектурі мережі певних особливостей, що значно покращують її загальну продуктивність. Нейрони такої згорткової мережі розташовані у трьох просторах (ширина, висота, глибина), кожен шар трансформує 3D вхідні дані у 3D вихідні нейронні активації.

Для побудови згорткової мережі, зазвичай, використовуються типи шарів: згортковий шар (Convolutional Layer), об'єднувальний шар (Pooling Layer), та повністю зв'язаний шар (Fully-Connected Layer). Шар RELU є явним записом активаційної функції. Найбільш розповсюдженою архітектурою є комбінація декількох CONV-RELU шарів, за якими слідує POOL, що повторюється до тих пір, доки розмірність презентації не стане надто малою. Іноді можна зустріти перехід до повністю з'єднаних шарів. Останній такий шар містить у собі вихідні дані мережі, наприклад, ймовірності класової приналежності.

Практика показує що власні архітектурні рішення стосовно комбінацій шарів не мають значного впливу на результат у 90% проблем. Замість створення власної архітектури для вирішення одиної проблеми, рекомендується знайти найбільш ефективну на даний момент, завантажити натреновану модель, та підігнати її під свої дані. Необхідність у створенні власної згорткової мережі з нуля, або її тренуванні, постає дуже рідко.

Створення власних моделей було дуже складним процесом у минулі часи. Проте, завдяки фреймворкам для машинного навчання, процеси отримання даних, тренування моделі та отримання результатів, стали значно простішими. Одним із таких фреймворків є TensorFlow від Google.

Написання та тренування глибоких нейронних мереж для класифікації написаних від руки чисел, розпізнавання зображень, обробки людської мови, та багатьох інших задач, дуже полегшено завдяки TensorFlow. Більше того, TensorFlow надає можливість створення графів для демонстрації потоку даних. Кожний вузол графу репрезентує математичну операцію, в той час як кожне з'єднання - тензор.

Датасети, що використовуються для тренування мереж для сегментації зображень, зазвичай, складаються з зображень, відповідних підписів, та піксельних масок. Кожний піксель відноситься до одного з попередньо визначених класів, наприклад, дорога, будівля, задній фон. Значну кількість корисних датасетів можна знайти на TensorFlow Datasets, що значно полегшує задачу завантаження, збереження та попередньої обробки даних.

Розглянемо варіант реалізації з використанням модифікованої UNet архітектури. Модель складається з двох шляхів. Перший шлях - це шлях скорочення (також його називають кодером), який використовується для отримання контексту зображення. Кодер представляє собою звичайну CNN. Другий шлях - це симетричний шлях розширення (також його називають декодером), який використовується для забезпечення локалізації за допомогою транспонованих конволюцій.

Для більш повного використання тренувальної вибірки буде проведено процес аугментації, для цього процесу також існує багато допоміжних бібліотек, у рамках даної реалізації використано `albumentations`.

Порівняння проведено на датасеті із зображеннями дорожніх сцен, виділяючи такі класи, як дорога, пішохід, небо, задній фон, та інші. Необхідну кількість епох важко передбачити заздалегідь, тому, зазвичай, це число обирається шляхом спроб та помилок, таким чином було обрано 40 епох.

Дослідження проведено із такими популярними моделями як UNet, FPN, та Linknet. Варто зазначити, що кожна з моделей має свої переваги, тому факт отримання найгіршого результату не свідчить про те, що для іншого варіанту використання, модель не показала б себе краще. В таблиці 1 нижче можна побачити результати навчання після 40 епох, працюючи з датасетом CamVid.

Таблиця 1 – Результати навчання після 40 епох

40 epoch	Loss	IOU	F	Val Loss	Val IOU	Val F
UNet	0.2006	0.6858	0.7820	0.1663	0.7272	0.8075
LinkNet	0.2263	0.6582	0.7606	0.2208	0.6803	0.7598
FPN	0.1960	0.6982	0.7936	0.1756	0.7153	0.7916

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Computer Vision. *What it is and why it matters* : вебсайт. URL: https://www.sas.com/en_us/insights/analytics/computer-vision.html (дата звернення: 08.01.2022).
2. T. Pham, "Semantic Road Segmentation using Deep Learning," *Applying New Technology in Green Buildings (ATiGB)*, 2021, pp. 45-48, doi: 10.1109/ATiGB50996.2021.9423307.

УДК 004.81

Малімон О. О., Сіденко Є. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ТА КЛАСИФІКАЦІЯ ТЕКСТІВ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ШТУЧНОГО ІНТЕЛЕКТУ

В сучасному світі більшість людей має власні профілі у соціальних мережах. Багато хто використовує свої сторінки висловлення власної думки. З всього цього може отримати користь бізнес, який за допомогою аналізу відгуків та коментарів в різних спільнотах зможе отримати об'єктивну та обґрунтовану думку згідно своїх товарів або послуг. Кожен бізнесмен хоче якомога краще розуміти клієнтів, їх настрої, потреби та думки – це дозволить покращити якість продукції та збільшити базу покупців.

Алгоритми аналіз тексту використовуються для оцінки думок людей або груп, наприклад, сегменту аудиторії якогось бренду. За допомогою механізму підрахунку балів алгоритми визначають ставлення, думки та емоції пов'язані з якоюсь подією, послугою чи темою, які відчуває людина. Аналіз настроїв іноді називають аналізом думок. Аналіз тексту є невід'ємним компонентом системи аналітики мовлення, який визначає думку або ставлення клієнту.

Згідно з актуальністю теми аналізу та класифікації тексту для дослідження обрано набір даних IMDb. Це набір, який складається із приблизно 15000 відгуків різних типів. Виділимо 2 різних типи повідомлень: позитивні та негативні. Дані в наборі розподілені рівномірно, це означає що позитивних й негативних відгуків майже порівну.

Для роботи з класифікацією тексту було використовувати сучасні методи та найбільш популярні бібліотеки, а саме ML.NET. Також для створення клієнтського застосунку використовувався сучасний фреймворк Angular. Вся серверна частина була написана за допомогою фреймворку ASP.NET Core.

В ході розробки було реалізовано плагіни з різними алгоритмами з метою подальшого їх порівняння, а саме: нейрона мережа та Naive Bayes. Для зручної роботи з реалізованими алгоритмами було створено один інтерфейс, який містить один публічний метод. В якості вхідних параметрів метод приймає речення, яке треба класифікувати, та додаткові налаштування для алгоритму. В якості результату повертається речення, його тип (позитивне або негативне), а також додаткові метрики, які були отримані під час виконання алгоритму та можуть бути цікаві користувачу.

В першому плагіні було реалізовано нейронну мережу за допомогою ML.NET. Точність розробленої моделі складає 95.7%. Спочатку вона навчається на тренувальному наборі, який складає 80% основної вибірки, й після чого перевіряє свою роботу на решті даних – 20% з базового набору.

Штучні нейрони мережі складаються з шарів, які в свою чергу – з вузлів. Нейронні мережі містять вхідний шар, один або кілька прихованих і один вихідний. Кожен вузол або штучний нейрон з'єднується з іншим і має відповідну вагу та поріг. Якщо вихід будь-якого вузла перевищує вказане порогове значення, цей вузол буде активований та надсилати дані на наступний рівень в мережі. В іншому випадку дані не передаються на наступний рівень мережі.

В наступному плагіні було реалізовано алгоритм Naive Bayes. С початку роботи створюється набір документів, які відносяться до категорій Good або Bad. Після цього створюється класифікатор та виконується аналіз вхідного речення від користувача. На виході алгоритм повертає дві ймовірності, а саме належність тексту до класу Good та Bad. Після чого робиться перевірка на те, яка з них більша. В разі якщо ймовірність належності до класу Good буде більше то речення буде класифіковано як Positive, інакше Negative.

Алгоритм класифікації Байєса являє собою алгоритм машинного навчання, який може бути використаний в широкому спектрі

задач класифікації. Частіш за все застосунки включаються в себе фільтрування спаму, класифікацію документу, аналіз тексту та інші. Алгоритм базується на теорії Томаса Байеса. Даний тип класифікатору є дуже популярним, адже його легко програмно реалізувати за допомогою будь-якої мови програмування. Також його можна використовувати в застосунках, які потребують обробки даних в реальному часі через те, що всі розрахунки є швидкими.

На рисунку 1 можна побачити сторінку з порівняльною характеристикою всіх алгоритмів, що реалізовані в системі.

Algorithms comparison

Here is a comparison table for all algorithms based on data that hasn't been used during training

#	Sentence	Expected	Naive Bayes	Neural Network
1	I really like this product over the Motorola because it is allot clearer on the ear piece and the mic.	Negative (1.00)	Positive (1.00)	Negative (0.82)
2	However, the keypads are so tiny that I sometimes reach the wrong buttons.	Positive (0.00)	Positive (0.88)	Positive (0.64)
3	I exchanged the sony ericson z500a for this and I'm pretty happy with that decision.	Negative (1.00)	Positive (0.99)	Positive (0.98)
4	I was very impressed with the price of the cases.	Negative (1.00)	Positive (0.93)	
5	Also makes it easier to hold on to.	Negative (1.00)	Positive (0.88)	Negative (0.57)

Рисунок 1 – Сторінка з порівняльною характеристикою алгоритмів

Для покращення точності роботи системи було видалено зайвий “шум”. Додатковим покращенням є видалення усіх слів, кількість яких є мінімальною, в порівнянні з іншими, адже вони не мають ніякого впливу при аналізі тексту. Видаливши такі слова, можна помітити, що результат класифікації став кращим, а продуктивність моделі збільшилась.

Слова, які є похідними від інших повинні бути сконвертовані до базової форми, або ж інфінітиву. Цей підхід називається Лематизація. Як приклад, уявімо, що в тексті є наступні слова: Playing, player, plays, play, played. Всі вони повинні бути нормалізовані до слова play.

В деяких випадках, особливості як комбінація слів можуть внести більший вклад для класифікації, ніж у випадках, коли всі слова беруться до уваги незалежними одне від одного. Комбінації з N слів разом називаються N-грамами. Відомо, що біграми є найкращим прикладом N-грам. Як приклад, слово “book” використовується як іменник, але може бути дієсловом. У прикладі іменника може бути

використано в реченні “book of N pages”, а як дієслово у “book a ticket”.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Savannah Krenzel Sentiment Analysis: A Definitive Guide : наукова стаття. URL: <https://monkeylearn.com/text-classification/> (дата звернення: 11.01.2022).
2. Alex Honchar Sentiment analysis: solutions and applications survey : наукова стаття. URL: <https://hp-analytics.medium.com/sentiment-analysis-solutions-and-applications-survey-9e52d3ea2ac7> (дата звернення: 11.01.2022).

УДК 004.6

Нечахін В. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ЗАСТОСУВАННЯ НЕЙРОМЕРЕЖЕВОЇ АРХІТЕКТУРИ LSTM В СИСТЕМІ КЕРУВАННЯ СОНЯЧНОЮ ЕЛЕКТРОСТАНЦІЄЮ

Зростання попиту на електроенергію, в поєднанні з можливістю скорочення запасів традиційних видів палива та зростаючою стурбованістю з приводу збереження навколишнього середовища, прискорили дослідження і розробки альтернативних джерел енергії, які є відновлюваними та мають незначний вплив на навколишнє середовище. Одним з найперспективніших засобів добування електроенергії є фотоелектрична генерація сонячними панелями, адже їх переваги включають відсутність потреби в регулярному обслуговуванні, відсутність шуму та використання сонячної енергії, яка є чистою і доступною. Вихідна потужність фотоелектричних модулів завжди змінюється в залежності від погодних умов, а саме від сонячного випромінювання і температури повітря. Отже, контроль за точкою максимальної потужності стає незамінним для підвищення ефективності фотоелектричних масивів. Цю функцію виконують контролери, які регулюють співвідношення між напругою та струмом з метою оптимізації потужності джерел електроенергії.

Одним із найпоширеніших алгоритмів знаходження точки максимальної потужності є метод висхідної провідності — він аналізує локальний максимум функції вольт-амперної характеристики та передбачає результат від зміни напруги. Цей метод має високі показники точності у порівнянні з іншими алгоритмами, але він схильний до виклику хаотичних коливань вихідної потужності джерела

електроенергії навіть при незмінних зовнішніх умовах. Також методу властиво працювати нестабільно у швидко мінливих атмосферних умовах, що призводить до втрати частини виробленої електроенергії. Для вирішення цих проблем пропонується використання рекурентних нейронних мереж для надання прогнозу щодо розташування точки максимальної потужності. Така система здатна передбачити зміни у продуктивності електростанції та, опираючись на минулі значення потужності та напруги, знаходити оптимальне відношення між ними. На відміну від алгоритму висхідної провідності, нейронна мережа не обмежується лише поданням сигналу на збільшення або зменшення напруги, а і вказує на ступінь необхідності зміни, що дає можливість швидше та точніше знаходити точку максимальної потужності не коливаючись навколо цього значення. Для реалізації цієї системи пропонується використовувати довгу короткочасну пам'ять LSTM — архітектуру рекурентних нейронних мереж, що використовується для роботи з часовими рядами та вирішення проблем прогнозування. Переваги LSTM над іншими типами рекурентних нейронних мереж включають в себе стійкість до проблем зникання та вибуху градієнту, а також нечутливість до прогалин у даних часового ряду. Ці проблеми вирішуються шляхом використання комірок пам'яті, що дозволяють вибірково запам'ятовувати важливу інформацію або забувати минулі дані. Нейронні мережі з архітектурою LSTM можуть ефективно виокремлювати довгострокові часові залежності, не зазнаючи перешкод при оптимізації. Також можливе використання двонаправленої LSTM, які приймають тренувальні дані у прямому та зворотному порядку, що розширює модель, надаючи їй повну інформацію про кожен пункт в заданій послідовності, а також минулі та майбутні дані відносно поточної точки. Обробка даних в обох напрямках допомагає двонаправленим LSTM отримати доступ не лише до минулого контексту, а і до майбутнього. Додатковий контекст прискорює процес тренування, а також покращує ефективність моделі при роботі з проблемою класифікації послідовностей.

У результаті було розроблено елементи системи керування для сонячних електростанцій на основі технологій штучного інтелекту. Були використані технології рекурентних нейронних мереж для знаходження точки максимальної потужності сонячних електростанцій. Для реалізації системи керування було створено багатозарову модель нейронної мережі з архітектурою LSTM, що використовує алгоритм ітеративного градієнтного спуску для тренування. При тестуванні моделі система знаходила точку максимальної потужності та показала приріст ефективності виробництва електроенергії у порівнянні із системою зі стандартним контролером.

ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО ЗОРУ ДЛЯ АВТОМАТИЗОВАНОГО ДІАГНОСТУВАННЯ ХВОРОБ ШКІРИ

Комп'ютерні технології використовуються в медицині майже в усіх сферах: діагностика, лікування та управління даними. Не дивлячись на важливість кожного процесу, діагностика завжди відіграла провідну роль. Оскільки без визначення конкретної хвороби неможливо почати ефективне лікування.

В сучасному світі нажаль все більше стали поширюватися злоякісні шкірні пухлини та з кожним роком кількість людей хворих такими захворюваннями тільки зростає. Основні причини появи – опіки шкіри, генетика, погане харчування. Саме через поширення таких хвороб з'явилася організація The International Skin Imaging Collaboration (ISIC) [1]. Головні цілі цієї організації – накопичення даних пов'язаних зі шкірними захворюваннями та проведення змагань на створення найкращої моделі для автоматизованого діагностування хвороб шкіри. Саме завдяки ISIC методи діагностування хвороб шкіри постійно розвиваються.

Згідно з актуальністю теми шкірних захворювань для дослідження обрано набір даних ISIC 2019. Це величезний набір із 25331 зображень різних шкірних захворювань. Усього 8 різних типів (класів) шкірних захворювань. Набір даних дуже нерівномірний, деякі класи мають не більше 300 зображень, а деякі більше 1000. Найбільший клас складається з 12875 зображень.

Для роботи з такою кількістю зображень потрібно використовувати найновітніші методи та оптимізовані засоби обробки зображень, усі вони зосереджені в бібліотеці OpenCV. Також для створення штучних нейронних мереж, які будуть класифікувати зображення, використовуються Tensorflow та Sklearn.

Основний та дуже важливий етап перед навчанням моделей – попередня обробка даних. Саме оптимальна обробка допомагає досягати найкращих результатів в навчанні. Усі зображення потрібно привести до стандартизованого відношення 1:1, щоб потім зменшувати до будь-яких популярних розмірів (112x112, 224x224). Однак першою перешкодою для масштабування зображень є їх розмірність. В наборі

даних знаходиться багато рисунків з різними пропорціями та з різною шириною та висотою. Тому проведено дослідження на кращий метод масштабування. Але усі вони мали свої недоліки, тому використано користувацький метод масштабування: видалення рамки в 10px, зменшення зображення по найменшій стороні зберігаючи пропорції, обрізання зображення до відношення 1:1. Саме такий підхід дозволяє обрізати якомога більше зайвого фону або здорової шкіри. Більшість зображень перевірені на правильність обрізання.

Через обмеженість апаратного забезпечення, набір даних повністю не може поміститися в оперативній пам'яті. Тому створено файл з розширенням HDF5, який може зберігати набори даних у постійній пам'яті та дозволяє працювати з ними, як у оперативній пам'яті. Саме на цьому етапі відбувається розподіл даних на тестові та для навчання у відсотках 20% та 80% відповідно.

Для покращення точності навчання використовується генерація додаткових даних. Генерація даних – це метод генерування нових навчальних даних без зміни міток класів шляхом застосування деяких випадкових спотворень і перешкод. Цей метод виконується на основі існуючих даних, для того щоб збільшити кількість даних в наборі для навчання та для того щоб покращити показник узагальнення мережі. Серед методів генерації даних використовуються: випадковий поворот, зміна ширини і довжини, приближення і віддалення, зсув, та дзеркальне відображення.

Після налаштування процесу обробки набору даних та запису їх в HDF5 файл можна створювати та налаштовувати моделі. Першою моделлю обрано MiniVGG, яка є VGG-подібною [2] моделлю. Цей тип моделей у своєму складі має дві підряд підключені згорткові шари з розміром фільтру 3x3 (на всіх рівнях), а лише після них йде зменшення розміру зображення.



Рисунок 1 – Графік навчання

В створеній моделі використовуються два шари по 32 фільтри та два шари по 64 фільтри. За ними йде нейронна мережа на 512 штучних нейронів. Для покращення точності використовуються шари нормалізації та випадкового вимкнення нейронів. Функцією активації обрано ReLU. Алгоритм оптимізації – зворотнє поширення помилки. Коефіцієнт навчання – 0.0001. Оскільки ця мережа не глибока, то розмір зображення обрано не 224x224, а 100x100.

Після 200 епох навчання вдалося отримати точність 63% на тестовій вибірці. Але навіть точність для навчання перестала зростати. Все це через недонавчання мережі. Причиною цього є дуже низький коефіцієнт навчання. Тому оптимальним буде створити таку саму мережу, але з меншим коефіцієнтом навчання, а саме 0.001. Нова модель змогла досягти 68.52% тестової точності. А точність навчання тренувальних даних продовжувала зростати.

Створена модель не може використовуватися в реальному світі через малу точність. Але в налаштованому середовищі розробки можна створювати та навчати інші, більш глибокі і складні моделі. Як приклад можна навчити більш глибоку VGG-подібну мережу. Або використати популярні варіанти мереж, як AlexNet та інші.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Welcome to ISIC. *The International Skin Imaging Collaboration* : вебсайт. URL: <https://www.isic-archive.com/> (дата звернення: 08.01.2022).
2. Karen Simonyan, Andrew Zisserman Very Deep Convolutional Networks for Large-Scale Image Recognition : наукова стаття. URL: <https://arxiv.org/abs/1409.1556> (дата звернення: 08.01.2022).

УДК 681.3

Под'ячев А. Д., Гожий О. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ВИКОРИСТАННЯ НЕЧІТКОЇ ЛОГІКИ В ІГРОВОМУ ШТУЧНОМУ ІНТЕЛЕКТІ

Ігровий штучний інтелект привертає увагу науковців та індустрії протягом останнього десятиліття. Сфера ігрового штучного інтелекту (ігровий ШІ) охоплює всі технології та методи введення інтелекту в відео ігри. До ігрового ШІ відноситься багато різних аспектів відеоігор: керування анімацією, керування, пошук шляху, планування, процедурна генерація, тактичне та стратегічне мислення та навчання [1]. Всі ці аспекти мають спільну основу, вони ставлять проблеми, ефективне вирішення яких вимагає алгоритмів ШІ. Мета ігрового ШІ – оживити неігрових персонажів (NPC), присутніх в іграх, наприклад, ворожі війська в стратегії або купців у мирному селищі, контрольованому штучним інтелектом. Ігровий ШІ, як і всі інші аспекти відеоігри, існує для досягнення вищої мети відеоігри: бути цікавою та розважальною, іншими словами, суспільство часто вважає девізом ігрового ШІ «Якщо гравець не бачить цього, навіщо це робити?» [1]. Проте контекст нині не такий простий, почасти завдяки успішним продуктам, які висунули передовий AI, частково завдяки поширені онлайн-ігор для кількох гравців, які змінили звички гравців, гравці тепер більш вимогливі до ШІ, і все більше і більше розробників ігор переходять від використання сценарних і передбачуваних агентів до використання переважно людиноподібних агентів, здатних до навчання і, отже, за класичним визначенням ШІ, розумні.

Академічні дослідження штучного інтелекту завжди відчували труднощі з визначенням самого штучного інтелекту, хоча зазвичай його можна визначити, як когнітивні процеси, схожі до когнітивних процесів людини або на рівні когнітивних процесів людини, і в здатність цих

процесів до навчання [3]. За словами Алана Тюрінга, якого вважають батьком штучного інтелекту, агент розумний, якщо його поведінку неможливо розрізнити від поведінки людини.

Головна особливість ігрового ШІ полягає в тому, що ігри, здебільшого, обмежені виконанням у режимі реального часу, тому залишається мало місця для повільних автономних методів. Цей факт робить ігровий ШІ подібним до ШІ, що використовується в робототехніці та системах керування [2], і це є причиною того, що можна використовувати, в тій чи іншій формі, ті ж самі прийоми, що використовуються у цих галузях.

Ще одна особливість це те, що сучасні ігри постійно удосконалюють та нарощують свої візуальні та аудіо складові, щоб залучити більше споживачів. Таким чином, більшість обчислювальних потужностей зайняті графічними обчисленнями, за якими часто слідувала обробка звуку і, звичайно, ігрова логіка. Це залишило ШІ дуже мало ресурсів для виконання своїх обчислень, що призвело до таких особливостей, як наприклад розповсюдження ворогів з низьким рівнем інтелекту у старих іграх. Порив до кращого, більш реалістичного та приємної графіки сповільнилася, деякі графічна складова деяких сьогоденішніх ігор практично не відрізняються від реальності. Зараз більшість із них підтримують виділені графічні процесори (GPU), які виконують графічні обчислення, залишаючи більше місця в ЦП для інших компонентів з потребуючих великих обсягів обчислень, таких як фізика та ШІ. У цьому контексті ігровий штучний інтелект заробив свою славу як засіб виділитися серед конкурентів. Нині більше часу потужності ЦП надаються ШІ в кожному ігровому кадрі, чого більше ніж достатньо для більшості програм, і деяких командах є програмісти, чиєю єдиною задачею є створенню складного, цікавого та (здавалося б) розумного.

Ігровий ШІ використовує багато методів із ширшої області ШІ, від простих кінцевих автоматів до найсучасніших еволюційних алгоритмів. Серед цих методів нечітка логіка є одним із інструментів повинен бути присутнім в арсеналі хорошого розробника ігрового ШІ через простоту її формулювання в поєднанні з її можливостями. Нечітка логіка приносить багато переваг для моделювання інтелектуальних ігрових агентів. Головна перевага – простота формулювання, що дозволяє розробникам з легкістю інтегрувати його в багато ігор, що є великою вигодою, якщо врахувати щільні графіки розробників ігор. Навіть у своїй простоті нечітка логіка може бути потужною технікою ШІ, особливо завдяки можливостям моделювати нелінійності, досягати складної поведінки за допомогою простих правил, бути інструментом для моделювання та відображення людських міркувань та емоцій.

Ігровий AI – це лише одна з галузей більш широкого поля штучного інтелекту. Ігровий ШІ додає ще один елемент до задачі визначення ШІ, оскільки вона не стосується реального існування інтелекту, але лише з ілюзією його. Щоб гра була успішною, нам насправді не потрібні високоінтелектуальні, схожі на людину, можливо, непереможні суперники, потрібен переконливий супротивник, який є агентом ШІ з яким приємно грати, та який симулює поведінку реальної людини чи істоти. При програмній реалізації поведінки ігрового персонажу нечітка логіка є одним з основних інструментів «інтелектуалізації» його поведінки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ian Millington. Artificial Intelligence for Games (The Morgan Kaufmann Series in Interactive 3D Technology). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.
2. Daniel Johnson and Janet Wiles. Computer games with intelligence. In In Procs. 10th IEEE Intl Conf. on Fuzzy Systems, pages 61–68. IEEE, 2001.
3. Hugo Pinto and Luis Otavio Alvares. Behavior-based robotic architectures for games. Game Programming Gems 6, 2006.

УДК 004.8

Попель О. О., Гожий О. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ХМАРНА ІНФРАСТРУКТУРА ДЛЯ ОБЧИСЛЕННЯ ЗАДАЧ МАШИННОГО НАВЧАННЯ ТА ШТУЧНОГО ІНТЕЛЕКТУ З ВИКОРИСТАННЯМ МЕТОДУ INFRASTRUCTURE AS CODE

Сфера машинного навчання та штучного інтелекту зараз розвивається швидше ніж будь-яка інша сфера комп'ютерних наук. Нейронні мережі уже здатні робити вражаючі речі — від розпізнавання тексту чи об'єктів на зображеннях та відео, до прийняття складних, багатокomпонентних рішень, таких як визначення медичних діагнозів.

Зі зростанням складності задач, які можуть виконувати нейронні мережі та зі зростанням об'ємів даних, які вони повинні оброблювати, зросли і потреби у обчислювальних потужностях, що необхідні для розробки та навчання таких нейромереж. Багато наукових закладів та R&D-відділів провідних компаній у сфері штучного інтелекту почали використовувати хмарну обчислювальну

інфраструктуру [1], замість обслуговування власної. Окрім зручності та усунення цілого класу проблем, пов'язаного з обслуговуванням власної інфраструктури, зараз хмарні обчислення інколи є безальтернативними, враховуючи кризову ситуацію з пропозицією на ринку потужних графічних адаптерів (відеокарт).

Основною проблемою хмарних рішень, особливо для невеликих наукових інститутів є їх вартість. З цієї проблеми випливає необхідність використання хмарних ресурсів лише за вимогою, тобто мінімізація часу простоювання інфраструктури. Для цього пропонується скористатись інструментами автоматизованого керування хмарною інфраструктурою, що реалізують підхід Infrastructure as Code (інфраструктура як код). Розглянемо приклад хмарної інфраструктури, що використовується для тренування нейронних мереж (рис. 1).

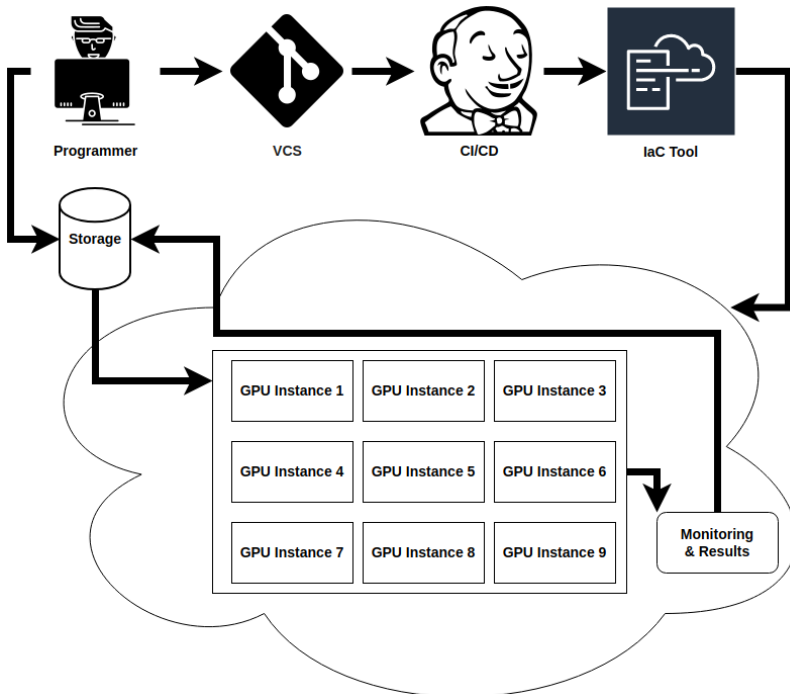


Рисунок 1 – Схема хмарної інфраструктури

На схемі зображено наступний процес роботи:

1. Програміст додає дані, які необхідно обробити до сховища (Storage).

2. Після цього, він додає новий код до системи контролю версій.
3. CI/CD-система бачить зміни у репозиторії і викликає IaC-інструмент.
3. IaC-інструмент створює усю необхідну хмарну інфраструктуру, в даному випадку — групу з 9 обчислювальних серверів та один сервер для моніторингу та перегляду результатів роботи.
4. Після запуску обчислювальні сервери отримують свою частку даних зі сховища та починають роботу.
5. Коли робота завершена, обчислювальні сервери вимикаються, а сервер моніторингу зберігає зібрані результати у постійному сховищі.
6. CI/CD-система знову запускає IaC-інструмент, тепер для знищення усєї тимчасової інфраструктури.

Можна виділити три переваги підходу Infrastructure as Code, на основі наведеної схеми — ідемпотентність, простота масштабування та аудит. Ідемпотентність дозволяє ефективно працювати в команді, адже будь-який її член може додавати нові дані та змінювати код, а стан інфраструктури буде залишатись тим самим при кожному запуску CI/CD. Простота масштабування проявляється у випадках, коли необхідно збільшити чи ускладнити існуючу інфраструктуру — IaC-інструменти дозволяють зробити це не перешкоджаючи роботі наявних хмарних ресурсів. До того ж, представлення інфраструктури у вигляді коду, по-перше, дозволяє легко клонувати її, для одночасного виконання декількох обчислювальних процесів, а по-друге дозволяє легко проводити аудит загального стану інфраструктури, хмарних ресурсів, які нею використовуються і індивідуальних налаштувань кожного з ресурсів. При цьому, підхід Infrastructure as Code є відносно молодим і недостатньо дослідженим [2], особливо його недоліки.

Для оцінки практичної цінності підходу і реальної оцінки раціональності його використання у тій чи іншій ситуації, проведено дослідження існуючих IaC-інструментів, проаналізовано як переваги і недоліки індивідуальних інструментів, так і підходу в цілому і розроблено представлення хмарної інфраструктури програмного проекту у вигляді коду Terraform — одного з найбільш широко використовуваних, гнучких та зручних IaC-інструментів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. D. Pop. Machine Learning and Cloud Computing: Survey of Distributed and SaaS Solutions. Institute e-Austria Timișoara, IEAT-TR-2012-1, 2012. 12 p.

2. A. Rahman, R. Mahdavi-Hezaveh, L. Williams. A systematic mapping study of infrastructure as code research. Information and Software Technology, Volume 108, 2019. P. 65-77.

УДК 004.81

Савчук О. А., Кондратенко Ю. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ДІАГНОСТУВАННЯ COVID-19 З ВИКОРИСТАННЯМ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ

Другий рік суспільство веде боротьбу із коронавірусом, який закрав на карантин більшу частину світу. Пандемія внесла свої корективи майже у всі сфери суспільства: в економіку, розваги, виробництво. Не залишилась осторонь і технологічна галузь. Велика кількість компаній перейшла на віддалену працю, але це не є рішенням для подолання проблеми.

Медпрацівникам, які стоять на варті боротьби к COVID-19, необхідно перевіряти та аналізувати сотні тисяч або навіть мільйонів медичних даних та діагнозів згідно власних знань, на кожний окремий випадок витрачається велика кількість часу за який пацієнта можна вже не врятувати. З стрімким розвитком алгоритмів машинного навчання, а саме нейронних мереж, проблему аналізу медичних даних, які є частиною медичної історії пацієнта стало можливо вирішити з високою точністю.

Задачею діагностування хвороб є аналіз симптомів пацієнта та його медичної історії за для вибору найкращого методу лікування хвороби. Розвитком даних алгоритмів є пошук найбільш ефективного лікування, коли алгоритм сам визначає не тільки симптоми пацієнта але й надає найбільш ефективний план лікування. Така система може виявляти хворобу на початку та попередити перехід в важку фазу.

Поняття штучного інтелекту було сформульовано у другій половині ХХ століття. Одне з перших визначень штучного інтелекту належить Джону МакКарті і було оприлюднене на конференції у Коледжі Дармуту (Нью Хемпшир) у 1956 р. як: «Спосіб примусити обчислювальну машину думати, як людина». Розуміння поняття інтелекту як придатності системи до навчання приводить науковців до ще одного з визначень штучного інтелекту: «здатність автоматизованих систем здобувати, адаптувати, модифікувати та поповнювати знання з метою пошуку рішень задач, формалізація яких ускладнена».

Сучасні методи медичної діагностики і біомедичних досліджень значною мірою ґрунтуються на аналізі зображень, одержуваних за допомогою технічних засобів (світлових і електронних мікроскопів, рентгено- та термографічних апаратів, томографів та ін.). Разом з тим, рішення діагностичних і наукових завдань при роботі з візуальною інформацією вимагає знання специфічних методів формування, реєстрації, цифрової обробки і аналізу зображень.

А. О. Білощицький і О. В. Діхтяренко розробили власний спосіб визначення ключових особливостей зображення. На відміну від визначення ключових точок, у цьому випадку головні риси зображення описували за допомогою векторів. Отримані набори векторів були основою для створення сигнатури зображення, для хешування використано локально чутливу функцію minHash. Метод названо min-Nash and tf-idfWeighting. Основне його завдання – це швидке виявлення схожих зображень у великих масивах даних. Метод знаходить схожі зображення, навіть якщо це різні зображення одного предмета, але помилкових спрацювань також багато.

Розробка системи дає зрозуміти, що у подібній технології є безліч практичних застосувань. Наприклад, система може стати помічником, який в реальному часі буде коментувати потенційний знімок. Система зможе проводити аналіз величезного числа зображень в короткий термін, і може грамотним способом оцінити зображення, пророкуючи прогресію захворювання.

В ході розробки системи використовувалася мова програмування Python з використанням бібліотек Tensorflow та Keras. Для навчання нейронної мережі використовувалося 50 зразків (25 зразків з позитивним COVID-19 та 25 зразків повністю здорових легень). Вхідні дані для навчання нейронної мережі представлені на рисунку 1. Результати навчання представлені на рисунку 2.

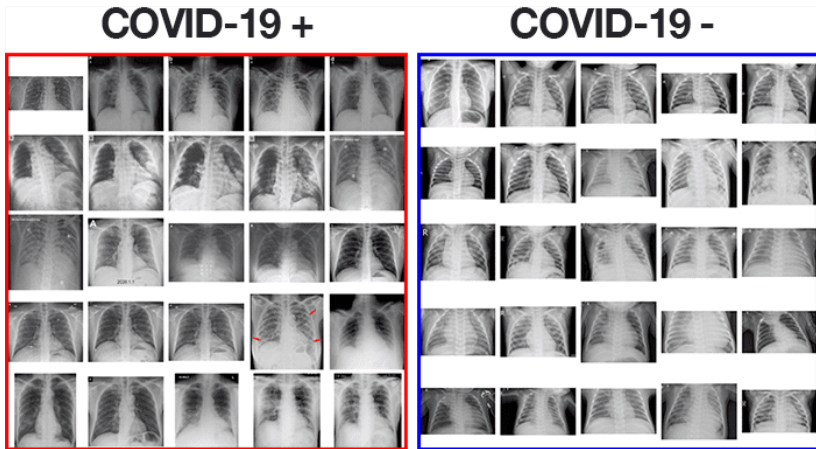


Рисунок 1 – Вхідні дані для навчання нейронної мережі

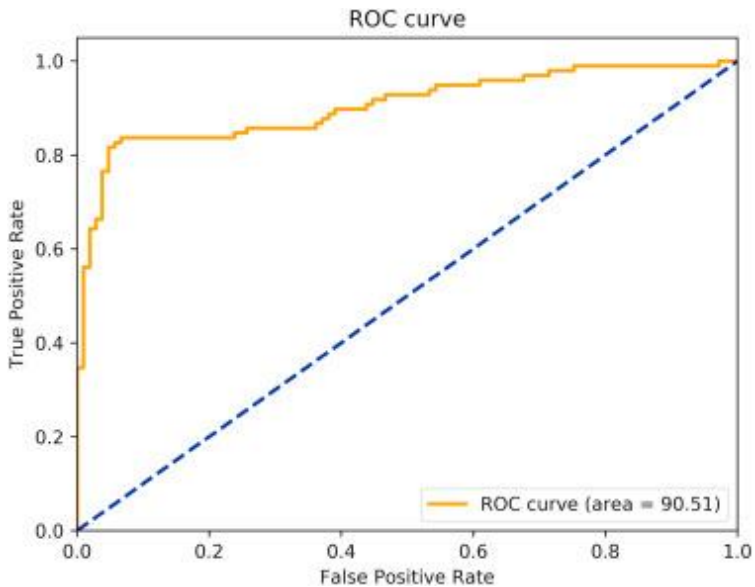


Рисунок 2 – ROC-крива результату навчання

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Z. Jianqiang, G. Xiaolin and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," in IEEE Access, vol. 6.

pp. 23253-23260, 2018. DOI: 10.1109/ACCESS.2017.2776930.

2. J.W. Crandall, et. al., "Cooperating with machines," in Nature Communications, vol. 9(1), 2018. DOI:10.1038/s41467-017-02597-8.

УДК 004.8

Скакун Є. І., Гожий О. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ДЛЯ ВИЗНАЧЕННЯ ЛОКАЦІЙ МЕТОДОМ МАШИННОГО НАВЧАННЯ

Пошук цікавих локацій головна проблема для мандрівників. Існує нескінченна кількість локацій, які можна відвідати, але щоб знайти саме ті локації, які можуть зацікавити користувачів, витрачається багато часу на пошук. Зазвичай цим займаються туристичні компанії, які надають сервісні послуги мандрівникам. Тому є нагальна потреба в інтелектуальних системах для підбору рекомендованих локацій за допомогою методів машинного навчання, що значно поліпшить процес пошуку користувачами цікавих локацій для відпочинку.

Метою роботи є створення швидкої та ефективної інтелектуальної системи для підбору рекомендованих локацій за допомогою методів машинного навчання на базі існуючого мобільного застосунку.

Основний алгоритм системи це алгоритм взаємодії підсистем розробленої системи наведений на рис. 1. В алгоритмі функціональні блоки наступні: *Блок «Розрахунок рейтингів»*: Формування рейтингів на основі відвіданих локацій або профайлу користувача. Даний підхід дозволяє формувати обмежену кількість попередньо розрахованих рейтингів для боротьби з такими проблемами, як «Розрідженість» [1] та «Холодний старт» [2].

Блок «БД рейтингів»: Центральне сховище збереження пар даних «користувач-локація-рейтинг».

Блок «Розрахунок рекомендацій»: Формування рекомендованих локацій за допомогою алгоритмів: MF[3], модифікацій KNN та інших.



Рисунок 1 – Схема взаємодії підсистем

Формування рейтингів на основі історії відвідування локацій базується на простому принципі виявлення ключових параметрів в історії відвідування за певний період часу, які вже впливають на формування інших рейтингів. Для боротьби з проблемами надлишкових даних за конкретними користувачами в обробку беруться тільки ті записи, які отримані пізніше за певний період часу. Всі отримані данні автоматично генеруються за рейтингом та нормалізуються в проміжку від 1 до 5, щоб не виникало колізії між автоматично розробленими рейтингами та реальними.

Для формування рекомендованих локацій застосовано 14 алгоритмів машинного навчання після оцінки яких, буде обраний лише один, який буде максимально підходити для обраного набору даних. Використання мови програмування Python та бібліотеки *Surprise* забезпечує можливість швидкої реалізації алгоритмів з гнучким налаштуванням різних параметрів. В розробленій системі є широкі можливості для вдосконалення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Martin Bauer, Nicolas Charon, Laurent Younes. Handbook of Numerical Analysis. USA 2019. P. 613–646.
 2. Saveski, Martin & Mantrach, Amin. Item cold-start recommendations. USA 2014 . P. 89-96.
- Koren, Yehuda & Bell, Robert & Volinsky, Chris. Matrix factorization techniques for recommender systems. USA 2009. P. 30-37.

ІНТЕЛЕКТУАЛЬНА СИСТЕМА АНАЛІЗУ КОНТЕНТУ МУЗИЧНОГО ВЕБСЕРВЕРУ ДЛЯ АНДРОЇД-ЗАСТОСУНКУ

Музичні застосунки використовуються людьми у всіх куточках планети. Музика є предметом організації музичних звуків, насамперед у часовій (ритмічній), звуковисотній та тембровій шкалі. Музичним може бути практично будь-який звук з певними акустичними характеристиками, які відповідають естетиці тієї чи іншої епохи, та може бути відтвореним. Джерелами такого звуку можуть бути: людський голос, музичні інструменти, електричні генератори тощо.

Музика діє вібраційно та створює незвичний вплив на організм людини. Зміни відбуваються на рівні клітин і органів, почуттів та думок. Музикотерапія є частиною оздоровлення та гармонізації особистості із середовищем і світом загалом. Тому музику і застосовують у вихованні, навчанні, лікуванні та навіть в управлінні державою.

Для покращення визначення оптимального набору (плейлист) музики для кожної людини слід розглянути можливість використання кластеризації даних (музики). Завдяки методам кластеризації музикальні твори можна групувати у кластери. Всередині кожної групи повинні виявитися «схожі» об'єкти, а об'єкти різних груп мають бути якомога відміннішими. Головна відмінність кластеризації від класифікації у тому, що перелік груп чітко не заданий й у процесі роботи алгоритму.

Серед основних алгоритмів кластеризації слід виділити такі алгоритми: k-середніх та пошарова кластеризація.

Метод k-середніх – метод кластерного аналізу, мета якого є поділ m об'єктів (з простору R^n) на k кластерів, при цьому кожен об'єкт відноситься до кластера, до центру (центроїду) якого воно найближче.

Як міру близькості використовується Евклідова відстань:

$$p(x, y) = \|x - y\| = \sqrt{\sum_{p=1}^n (x_p - y_p)^2}$$

де $x, y \in R^n$.

Отже, розглянемо низку об'єктів $(x^{(1)}, x^{(2)}, \dots, x^{(m)}), x^{(j)} \in R^n$.

Метод k-середніх поділяє m об'єктів на k груп (або кластерів) ($k \leq m$) $S = \{S_1, S_2, \dots, S_k\}$, щоб мінімізувати сумарне квадратичне відхилення точок кластерів від центроїдів цих кластерів:

$$\min \left[\sum_{i=1}^k \sum_{x^{(j)} \in S_i} \|x^{(j)} - \mu_i\|^2 \right],$$

де $x^{(j)} \in R^n, \mu_i \in R^n$,

μ_i - центроїд для кластера S_i .

При визначеній мірі близькості до центроїду розбиття об'єктів на кластери зводиться до визначення центроїдів цих кластерів. Число k кластерів задається дослідником заздалегідь.

Слід розглянути початковий набір k середніх μ_1, \dots, μ_k у кластерах S_1, \dots, S_k . На першому етапі центроїди кластерів вибираються випадково або за певним правилом (наприклад, вибрати центроїди, що максимізують початкові відстані між кластерами). За отриманими даними відносити спостереження тим кластерам, чие середнє (центроїд) до них найближче. Кожне спостереження належить лише до одного кластеру, навіть якщо його можна віднести до двох і більше кластерів. Потім центроїд кожного i -го кластеру перераховується за таким правилом:

$$\mu_i = \frac{1}{S_i} \sum_{x^{(j)} \in S_i} x^{(j)}$$

Таким чином, алгоритм k-середніх полягає у перерахуванні на кожному кроці центроїду для кожного кластера, отриманого на попередньому кроці. Алгоритм зупиняється, коли значення μ_i не змінюються: $\mu_i^t = \mu_i^{t+1}$.

Важливо: спочатку провести перевірку відповідного числа кластерів для набору даних. Неправильний вибір кількості k кластерів може призвести до некоректних результатів.

Пошарова кластеризація. Цей алгоритм заснований на виділенні зв'язкових компонентів графа на певному рівні відстаней між об'єктами (вершинами). Рівень відстані визначається порогом відстані c . Наприклад, якщо відстань між об'єктами $0 \leq p(x, x') \leq 1$, та $0 \leq c \leq 1$. Алгоритм пошарової кластеризації формує послідовність підграфів графа G , які відображають ієрархічні зв'язки між кластерами:

$$G^0 \subset G^1 \subset \dots \subset G^m,$$

де $G^t = (V, E^t)$ - Граф на рівні c^t ,
 $E^t = \{e_{ij} \in E : p_{ij} \leq c^t\}$,
 c^t – t -ий поріг відстані,
 m – кількість рівнів ієрархії,
 $G^0 = (V, o)$, o – порожня множина ребер графа, одержуване при $t = 0$,
 $G^m = G$, \Rightarrow граф об'єктів без обмежень на відстань (довжину ребер графа), оскільки $t_m = 1$.

За допомогою зміни порогів відстані $\{c_0, \dots, c_m\}$, де $0 = c_0 < c_1 < \dots < c_m = 1$, можна контролювати глибину ієрархії одержуваних кластерів. Таким чином, алгоритм пошарової кластеризації здатний створювати як плоске ієрархічне розбиття даних.

При використанні даних алгоритмів кластеризації розроблено андроїд застосунок, який за основними тегами пісень (рік, жанр, виконавець, альбом) створює рекомендації до певного суб'єкта за його вподобаннями та активністю, демо працює із музичними творами в пам'яті телефону.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Кластеризация K-средних и PCA для классификации музыки по схожим звуковым характеристикам. URL: <https://ichi.pro/ru/klasterizacia-k-srednih-i-pca-dla-klassifikacii-muzyki-po-shozim-zvukovym-harakteristikam-245233124019044>
2. Пічушкін К.А. Гибридная система классификации музыкальных треков. URL: https://dspace.spbu.ru/bitstream/11701/10624/1/Kostya_Pichushkin_-_diploma.pdf
3. Обзор алгоритмов кластеризации данных. Хабр: вебсайт. URL: <https://habr.com/ru/post/101338/>

УДК 004.8+330.115

Скубак О. Д., Петроченко О. О., Калініна І. О.

*Чорноморський національний університет імені Петра Могили,
 м. Миколаїв, Україна*

АНАЛІЗ ЧАСОВИХ РЯДІВ ЗА ДОПОМОГОЮ МАШИННОГО НАВЧАННЯ

Прогнозування – це найпоширеніша завдання, що виникає під час роботи з часовими рядами. Проте одержати надійні прогнози

непросто. Найчастіше для побудови прогнозу вдаються до ретельного аналізу часового ряду.

Найбільш популярними підходами до аналізу часових рядів є класичні: експонентне згладжування, модель Хольта-Вінтерса, AR, MA, ARMA ARIMA та інші статистичні моделі. Незважаючи на вагомні переваги вони мають ряд недоліків: відносна простота таких моделей не дозволяє мати високу точність при обробці великих наборів даних; статистичні моделі орієнтуються оцінку математичного очікування розподілу, а чи не самого розподілу; такі моделі не створені для опису динаміки нелінійних, нестационарних процесів і погано

У роботі реалізовані та досліджуються методи прогнозування засновані на застосуванні методології *Generalized Additive Models (GAM)* [1]. В основі цієї методології полягає процедура підгонки *адитивних регресійних моделей* наступного виду:

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t,$$

де $g(t)$ і $s(t)$ – функції, що апроксимують тренд ряду та сезонні коливання (наприклад, річні, тижневі тощо) відповідно, $h(t)$ – функція, що відображає ефекти свят та інших впливових подій, а ε_t – нормально розподілені випадкові збурювання. Для апроксимації функцій були використані такі методи:

- *тренд*: кусково-лінійна регресія;
- *річна сезонність*: часткові суми ряду Фур'є, кількість членів якого (порядок) визначає гладкість функції;
- *тижнева сезонність*: представлена у вигляді індикаторної змінної;
- *«свята»* (наприклад, офіційні святкові та вихідні дні – Новий рік, Різдво тощо, а також інші дні, під час яких властивості тимчасового ряду можуть істотно змінитись – спортивні чи культурні події, природні явища тощо) представлені також як індикаторні змінні.

Оцінювання параметрів моделі, що підганяється, виконувалося з використанням принципів байєсівської статистики: методом знаходження апостеріорного максимуму (MAP) або шляхом повного байєсівського виведення. Для цього застосовувалася платформа ймовірнісного програмування Stan, інтегроване середовище розробки RStudio.

Експериментальна частина виконана із застосуванням набору Cryptos, який містить зібрані із сайту [2] значення вартості 22 криптовалют на момент закриття торгів. Вартість криптовалют - це не найпростіші змінні для моделювання, що справедливо для

переважної більшості фінансових часових рядів. Часові ряди набору даних Стуртос мають складний тренд, дисперсія його значень зростає з часом (вони не стаціонарні), мають місце різкі зміни рівнів, ймовірно пов'язані з якимись особливими подіями. Проте це приклад реальних даних, з якими аналітик може зіткнутися на практиці.

Для перевірки якості підібраних альтернативних моделей прогнозування використано модифікацію методу перехресної перевірки, що враховує тимчасову спрямованість даних, а саме метод «імітованих історичних прогнозів» [3]. Як метрики для перевірки якості використано: середньоквадратична помилка (MSE), квадратний корінь із середньоквадратичної помилки (RMSE), середня абсолютна помилка (MAE), середня абсолютна питома помилка (MAPE), а також частка істинних значень змінної, що моделюється, які знаходяться в межах довірчих меж прогнозу.

Реалізована у роботі інформаційна система на основі методології GAM дає потужний інструмент для прогнозування часових рядів за допомогою машинного навчання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Taylor S. J., Letham B. Forecasting at Scale. The American Statistician, Vol. 72, 2018, Issue 1, pp. 37-45. <https://doi.org/10.1080/00031305.2017.1380080>.
2. CoinMarketCap. Today's Cryptocurrency Prices by Market Cap: вебсайт.URL: <https://coinmarketcap.com/>
3. Hyndman R. J., Athanasopoulos G. Forecasting: principles and practice. Publisher : OTexts, 2013. 293 p.

УДК 004.4

Фінажсин М. Ф., Калініна І. О.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ НА ОСНОВІ ІНДЕКСІВ КРИПТОВАЛЮТ

Інформація у сучасному прогресивному світі є одним з ключових елементів економічного розвитку, яка в деяких видах діяльності, може перевищувати вартість інших цінних матеріалів, таких як земля, робоча сила та інші. Найціннішою інформацією можна вважати лише найактуальніші дані. Інакше кажучи, за інших рівних

умов, чим свіжіша інформація, тим вона цінніша. Іншим важливим елементом цінної інформації є її точність та достовірність. Можливо сказати, що вся цінність ринку полягає у отриманні актуальної інформації. Саме тому системи прогнозування набули такої популярності. Окрім того, з появою криптовалют, ринок зазнав значних змін, так як вони є важливим ресурсом, який надає багато нових можливостей. Можливість спрогнозувати зміни криптовалют, надає власнику цієї інформації значний контроль над ситуацією на ринку, що може захистити від ризикових наслідків.

Розроблена інтелектуальна система прогнозування надає точні результати спрогнозованої вартості криптовалют. Подібна ефективність системи досягається завдяки виконанню поступових ітераційних етапів обробки даних. Основними етапами, які можливо виділити є наступні:

- Етап збірки та обробки даних. Даний етап слугує початковою ланкою роботи системи. Існує багато ресурсів-агрегаторів, які надають можливість отримати бази даних з колекціями крипто валютних індексів. Саме на цих даних, система буде навчатись та прогнозувати у подальшому. Окрім цього, на даному етапі, з набору криптовалютних індексів, вилучаються усі некоректні значення.

- Етап агрегації даних. Агрегування даних, потрібно для вилучення потрібних часових рядів з існуючого масиву даних. Завдяки розбиттю даних на необхідні часові ряди, система має можливість використання найбільш точної моделі прогнозування.

- Етап навчання моделей. Виконується розбиття використаних у моделі даних на тренувальну та перевірочну вибірки. В подальшому, кожна з моделей використовує власний набір особливостей прогнозування, що дає різну точність прогнозування.

- Етап прогнозування даних. Кінцевий етап роботи системи. Система робить вибір найбільш правдивої моделі прогнозування за допомогою виконання перехресної перевірки серед вже навчених моделей.

Кожний з етапів хоч і є поступовим у виконанні, має можливість ітераційного застосування. На виконанні кожного з етапів, система має можливість повернутися до будь-якого з попередніх, для уточнення або зміни вже існуючого набору даних. Це дає можливість гнучкого навчання системи, що веде до високо-точного прогнозування вартості криптовалют.

Слід зазначити, що при прогнозування вартості криптовалют з використанням криптовалютних індексів, часові ряди грають одну з основних ролей. Часові ряди зручно використовувати в різних системах оцінки, починаючи від прогнозування криптовалют і закінчуючи оцінкою активності сонячних циклів. Основні компоненти, через які

використання часових рядів є доцільним у задачах прогнозування криптовалют: тренд (Tt): характеризує довготривалу тенденцію даних (зниження чи зростання); циклічна компонента (Ct): довготривалі циклічні коливання, які зазвичай займають не менше 2 років; сезонна компонента (St): короточасні періодичні зміни, що мають фіксовану частоту (наприклад, добові зміни кількості сонячного світла, що падає на одиницю поверхні Землі); нерегулярна компонента (ϵ): ефекти випадкових факторів (“шум”).

Для прогнозування точних результатів, використовується розроблений спеціалістами компанії Facebook пакет prophet на базі мови програмування R. Головними перевагами прогнозування за допомогою подібної технології є не тільки можливість використання класичних і сучасних методів прогнозування (експонентне згладжування, модель Хольта-Вінтерса, ARIMA, моделі для згрупованих часових рядів, рядів з кількома сезонними компонентами і т.д.), так і використання сучасної методології адаптивних регресивних моделей, яку можливо описати наступним чином:

$$y(t)=g(t)+s(t)+h(t)+\epsilon t,$$

де $g(t)$ і $s(t)$ — функції, що апроксимують тренд ряду та сезонні коливання (наприклад, річні, тижневі тощо) відповідно, $h(t)$ — функція, що відображає ефекти свят та інших впливових подій, а ϵt - нормально розподілені випадкові обурення.

Визначення найбільш ефективної моделі системи, проводиться з використанням імітованих історичних прогнозів, так як саме цей метод, дозволяє зручно працювати з часовими рядами.

З використанням усіх пройдених ітераційних етапів для виявлення найбільш ефективної моделі прогнозування, які може навчити система, користувач отримує у свої руки високоточний інструмент, який дозволяє працювати з якісними прогнозами по індексам криптовалют.

Системний аналіз, моделі і засоби підтримки прийняття рішень

УДК 004.8

ПЕРСПЕКТИВИ АНАЛІЗУ ДАНИХ ЗАХВОРЮВАНЬ COVID-19 НА БАЗІ ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ

Пандемія коронавірусної хвороби 2019 року (COVID-19) торкнулася майже 271 мільйон чоловік і забрала 5,32 мільйона життів, самий останній епізод – варіант дельта коронавірусу – важкого гострого респіраторного синдрому. Пандемія COVID-19 просто доповнює список інфекційних захворювань, які були потенційними глобальними загрозами, такими як важкий гострий респіраторний синдром, Ебола і Зіка. Подібні інфекції підкреслюють необхідність розробки терапевтичних засобів для боротьби з виникаючими патогенами.

Процес розробки терапевтичних рішень для нових вірусів є виснажливим і непомірно довгим і займає від 10 до 15 років. Початковий етап визначення важливих молекул і терапевтичних мішеней для подальшого дослідження має вирішальне значення через величезні розміри хімічного простору, що запобігає вичерпний пошук з використанням дорогих експериментів і випробувань. Інструменти машинного навчання та високопродуктивних обчислень все частіше використовуються для визначення перспективних кандидатів на ліки. Хоча обчислювальні методи можуть частково знизити деякі пов'язані з цим експериментальні витрати, для навчання алгоритму машинного навчання зазвичай потрібна велика складова бібліотека з вимірюваними властивостями. Тому організація своєчасного реагування на виникаючу пандемію також створює проблему для обчислювальних методів, оскільки необхідно генерувати великі набори даних.

Для автоматизації процесу пошуку ліків необхідний алгоритм, який: перше – використовує існуючі великі бібліотеки, заповнені хімічними сполуками, без необхідності вимірювання хімічних властивостей; друге – прогнозує спорідненість до нових білкових мішеней з дуже обмеженими доступними експериментальними даними; третє – досліджує хімічний простір цільового патогена/інфекції для ефективної ідентифікації сполук для подальшого дослідження.

Щоб задовольнити ці три критерії, дослідники Національної Лабораторії Oak Ridge [1] використовували високопродуктивні обчислення для навчання узагальнених моделей машинного навчання, як для генерації кандидатів, так і для прогнозування подібності.

Їх експеримент був нещодавно опублікований на сервері попереднього друку bioRxiv і дав уявлення про використання алгоритму

на основі машинного навчання для аналізу та прогнозування терапевтичних цілей у виникаючих патогенів з цілим рядом мутацій.

Щоб скористатися перевагами великих існуючих бібліотек хімічних сполук, дослідники використовували текстове представлення даних про молекули, відоме як спрощена система введення молекулярного рядка. Використовуючи базу даних Epubine REAL як відправну точку, вони створили новий набір даних, що містить приблизно 9,6 мільярда унікальних молекул. Набір даних використовувався для попередньої підготовки моделі трансформатора, з використанням прогнозуючої маски, що зазвичай зустрічається в застосунках для обробки природної мови. Під час попереднього навчання підпослідовності даної молекули були замінені маскою, і модель була перевірена на здатність передбачати відповідну послідовність на основі контексту. Таким чином, модель отримала уявлення про хімічну структуру абсолютно неконтрольованим чином, що не вимагало додаткових вимірювань властивостей.

При попередньому запуску моделі глибокого навчання на приблизно 9,6 мільярдах молекул дослідники досягли максимальної продуктивності в 603 петафлопса зі змішаною точністю. Таким чином, цей експеримент дозволив успішно скоротити час попередньої підготовки з декількох днів до декількох годин у порівнянні з попередніми спробами з цією архітектурою. Цей процес також збільшив розмір набору даних [2].

Таким чином сучасні вчені почали вдаватися до використання новітніх технологій для підвищення ефективності боротьби з таким відомим і заразним вірусом як COVID-19. Завдяки даній розробці в найближчому майбутньому стане можливим знаходити вакцину новому вірусу в лічені хвилини, що допоможе людству знизити ризик смертності від раптово виниклого вірусного захворювання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. National Laboratory Oak Ridge. Wikipedia: вебсайт. URL: https://en.wikipedia.org/wiki/Oak_Ridge_National_Laboratory (дата звернення: 20.12.2021)
2. Identifying protein targets in SARS-CoV-2 via machine learning. News Medical: вебсайт. URL: <https://www.news-medical.net/news/20211217/Identifying-protein-targets-in-SARS-CoV-2-via-machine-learning.aspx> (дата звернення: 20.12.2021)

СИСТЕМА ОЦІНЮВАННЯ ІННОВАЦІЙНИХ ПРОЄКТІВ НА ОСНОВІ МОДИФІКОВАНОГО FUZZY TOPSIS

На сьогоднішній день інноваційні проекти стали досить популярними та широко розповсюдженими в будь-якій сфері діяльності, так як результатами інноваційних проєктів є поліпшення життя людей. На сьогоднішній день існує величезна кількість відповідних проєктів, тому було вирішено створити систему, що допоможе у виборі оптимального чи найкращого рішення на основі модифікованого Fuzzy TOPSIS, та дослідити вплив його параметрів на кінцевий результат.

Метою роботи є розробка програмного застосунку для оцінки інноваційних проєктів на основі модифікованого методу Fuzzy TOPSIS, який призначений допомогти користувачам в прийнятті оптимальних рішень, полегшити процес оцінювання інноваційних проєктів та визначити вплив параметрів зазначеного методу на результат.

Метод Fuzzy TOPSIS працює за наступним алгоритмом: оцінювання критеріїв та альтернатив експертами; перетворення лінгвістичних термів (ЛТ) в нечіткі числа; усереднення оцінок; нормалізація усереднених оцінок; формування зваженої нормованої матриці рішень; розрахунок відстані кожної альтернативи до нечіткого ідеального рішення (FPIS, англ. Fuzzy Positive Ideal Solution) та нечіткого негативного ідеального рішення (FNIS, англ. Fuzzy Negative Ideal Solution) за кожним критерієм; розрахунок коефіцієнту близькості альтернатив; здійснено ранжування альтернатив.

В ході виконання поставленого завдання розроблено програмний застосунок зі зручним інтерфейсом (рис. 1). Успішно реалізовано всі необхідні модифікації та розрахунки (рис. 2).

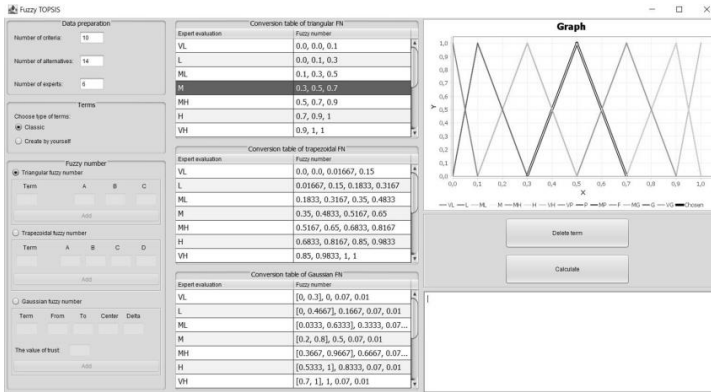


Рисунок 1 – Розроблений програмний застосунок для оцінювання інноваційних проєктів

Closeness coefficient	Rank	Closeness coefficient	Rank		
CC1 MPBoard	0,4553	11	CC1 MPBoard	0,4553	11
CC2 Pixium	0,4492	12	CC2 Pixium	0,4492	12
CC3 GUPY Services	0,5204	3	CC3 GUPY Serv	0,5204	3
CC4 Snager	0,4377	13	CC4 Snager	0,4377	13
CC5 SIFmeter	0,4923	5	CC5 SIFmeter	0,4923	5
CC6 Holo Media System	0,5211	2	CC6 Holo Medi	0,5211	2
CC7 Interview.top	0,4709	8	CC7 Interview,i	0,4709	8
CC8 InstaAdver	0,5090	4	CC8 InstaAdve	0,5090	4
CC9 Pillars of Light	0,4635	9	CC9 Pillars of L	0,4635	9
CC10 E-Cup	0,3661	14	CC10 E-Cup	0,3661	14
CC11 Elxy	0,4832	7	CC11 Elxy	0,4832	7
CC12 Electro teacher	0,4628	10	CC12 Electro te	0,4628	10
CC13 Cyberstick	0,5901	1	CC13 Cyberstic	0,5901	1
CC14 Econd	0,4854	6	CC14 Econd	0,4854	6
Closeness coefficient	Rank	Closeness coefficient	Rank		
CC1 MPBoard	0,4420	11	CC1 MPBoard	0,5053	1
CC2 Pixium	0,4335	12	CC2 Pixium	0,5034	6
CC3 GUPY Services	0,5271	3	CC3 GUPY Serv	0,5053	2
CC4 Snager	0,4184	13	CC4 Snager	0,5044	5
CC5 SIFmeter	0,4895	5	CC5 SIFmeter	0,5049	4
CC6 Holo Media System	0,5283	2	CC6 Holo Medi	0,5049	4
CC7 Interview.top	0,4619	8	CC7 Interview,i	0,5030	8
CC8 InstaAdver	0,5121	4	CC8 InstaAdve	0,5030	8
CC9 Pillars of Light	0,4520	9	CC9 Pillars of L	0,5020	9
CC10 E-Cup	0,3274	14	CC10 E-Cup	0,4990	14
CC11 Elxy	0,4776	7	CC11 Elxy	0,4990	14
CC12 Electro teacher	0,4513	10	CC12 Electro te	0,4990	14
CC13 Cyberstick	0,6234	1	CC13 Cyberstic	0,5000	11
CC14 Econd	0,4811	6	CC14 Econd	0,5000	11

Рисунок 2 – Результат роботи програмного застосунку для оцінювання інноваційних проєктів

Також варто зазначити, що дані, які були застосовані в даній роботі є реальними, вони були отримані при оцінці експертами

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Hwang, C.L., Yoon, K., “Multiple Attributes Decision Making Methods and Applications”, Springer, Berlin Heidelberg, 1981.
2. L.A. Zadeh, “Fuzzy sets”, Inform. and Control, 1965, 8, pp. 338-353.
3. Chen, C.T., “Extensions of the TOPSIS for group decision making under fuzzy environment”, Fuzzy Sets and Systems, 2000, 114, pp. 1-9.

УДК 005

Гончарова Н. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПРИНЦИПИ ПОБУДОВИ ДІАГРАМ ВПЛИВІВ ДЛЯ ВИРІШЕННЯ ЗАДАЧ СЦЕНАРНОГО АНАЛІЗУ РИЗИКОУТВОРЮЮЧИХ ФАКТОРІВ В ЗАКЛАДАХ ВИЩОЇ ОСВІТИ

Важлива роль у роботі будь-якої організації приділяється її здатності пристосовуватися до сучасних умов динамічних науково-технічних, соціально-економічних, політичних змін. Діяльність закладів вищої освіти (ЗВО) передбачає низку ризикоутворюючих факторів, серед яких можна виділити такі: набір студентів першого курсу, отримання фінансування освітньої і наукової діяльності, набір і підготовка необхідної кількості кваліфікованих викладачів та інші. Прогнозування багатьох ризиків, які виникають під час таких ситуацій, може допомогти підвищити ефективність діяльності ЗВО. Одним з широко використовуваних інструментів прогнозування є сценарний аналіз.

Існує низка методів, які застосовуються для побудови і аналізу сценаріїв. Серед них можна виділити діаграму впливів.

Цей метод був розроблений і удосконалений професором Рональдом Ховардом зі Стенфордського університету та його колегою доктором Джеймсом Метісоном в кінці минулого століття як зручна нотація для зображення проблеми прийняття рішень, яка доповнює дерева рішень [1].

Діаграми впливу (ДВ) являють собою ациклічні орієнтовані графи, що складаються із різнотипних вузлів, що відповідають множині факторів, відносно яких приймається рішення, і ребр – відношень між ними.

Будь-яка діаграма впливу включає три типи вузлів [2]:

- 1) вузли рішень D (decision);
- 2) вузли подій C (chance variable);
- 3) вузли цінностей V (value).

Вузли рішень відображають множину альтернатив рішень. На діаграмі вони позначаються у формі квадратів або прямокутників. Вузли подій відображають множину подій, які прямо або опосередковано обумовлюють результати альтернатив і на графі зображуються колами. Вузли цінностей позначають функції, за значеннями яких оцінюються результати альтернатив. На діаграмі вони позначаються ромбами.

Діаграма може включати в себе такі типи зв'язку між вузлами (рис. 1) [2]:

а) зв'язок між двома вузлами подій – позначає можливий з певною ймовірністю (але не обов'язковий) зв'язок між вузлами i та j ;

б) між вузлом рішень і вузлом подій – подія j може бути асоційованою із рішенням вузла i ;

в) між вузлом подій і вузлом рішень – зв'язок впливу. Він означає, що людина, що приймає рішення, знатиме про результат виникнення події i і може зважати на це у процесі прийняття рішення у вузлі j ;

г) між двома вузлами рішень – аналогічно попередньому типу означає, що прийняте рішення у вузлі i , може вплинути на рішення у вузлі j ;

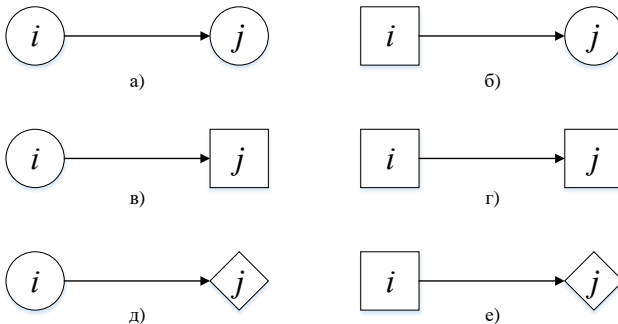


Рисунок 1 – Типи з'єднання на діаграмі впливу

д) між вузлом подій і вузлом цінності – позначає, що результати альтернативних рішень можуть залежати від подій у вузлі i ;

е) між вузлом рішення і вузлом цінності – результати альтернативних рішень залежать від рішення у вузлі i .

Серед підходів формування ДВ виділяють її ціленаправлену генерацію та інтерактивну побудову [2]. Згідно з інтерактивним підходом спочатку визначають загальні вузли рішень, цінності і подій і зв'язки між ними, а далі діаграма послідовно доповнюється новими елементами і зв'язками безпосередньо у ході аналізу сценаріїв.

Метод ціленаправленої генерації ДВ включає декілька етапів. Перший етап передбачає побудову мінімальної діаграми впливів. Для цього визначається кількість проміжних рішень, що відповідає кількості вузлів рішень на діаграмі, і яким чином будуть оцінюватись результати прийнятих рішень. На другому етапі визначаються випадкові події, які обумовлюють результати альтернативних рішень, і діаграма розширюється шляхом доповнення її вузлами подій. Далі визначається, чи існують ще групи подій, які можуть вплинути на виникнення подій, які вже входять до ДВ. Аналіз продовжується доти, поки не будуть встановлені всі невизначені фактори, що можуть впливати на результати. Побудова діаграми впливів за таким підходом дозволяє поступово розробити широку діаграму до початку аналізу сценаріїв.

Далі наведено приклад побудови діаграми впливу. Нехай, ЗВО має провести рекламну кампанію для залучення нових студентів. Отже, вузлом рішення D буде рішення, чи погоджуватися на отримання послуг від певної рекламної компанії. Критерієм його цінності буде вартість послуг V . На основі отриманих вузлів будується мінімальна ДВ (рис. 2, а).

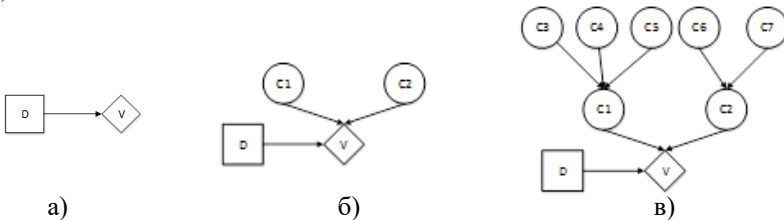


Рисунок 2 – Процес побудови ДВ

Далі визначаються, які чинники можуть вплинути на ціну – нехай, це будуть C_1 – вартість розповсюдження реклами, C_2 – вартість створення рекламних матеріалів – і діаграма доповнюється, у цьому випадку, двома вузлами подій C_1 і C_2 (рис. 2, б). На наступному етапі, необхідно визначити, чи можуть на ці події вплинути інші фактори. Наприклад, на вартість розповсюдження реклами можуть вплинути такі

чинники: C_3 , C_4 , C_5 – можливість економії на розміщенні реклами у соцмережах, на телебаченні або роздачі листівок; а на вартість її створення: C_6 , C_7 – можливість економії на створенні відеоролика і листівок відповідно. Так діаграма розширюється шляхом доповнення її новими вузлами подій C_3 , C_4 , C_5 , C_6 і C_7 (рис. 2, в).

Отже, одним з методів графічного представлення сценаріїв для подальшого їх аналізу і розробки стратегії дій можуть бути ДВ. Вони складаються із різнотипних вузлів і зв'язків. Одним з методів побудови ДВ є ціленаправлене їх генерування, який є доцільнішим для використання, оскільки передбачає повну побудову ДВ ще до початку аналізу сценаріїв.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Коваленко І. І., Чернов С. К., Швед А. В., Чернова Л. С., Антіпова К. О. Методи системного аналізу в задачах морських кластерів. Х., ЧП «Издательство «Новое слово»», 2017. 300с.

2. Борисов А. Н., Ужга-Ребров А. І., Савченко К. І. «Вероятностный вывод в интеллектуальных системах». Рига, 2002. 190 с.

УДК 004.02

Грохольська А. І., Сіденко Є. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ДОСЛІДЖЕННЯ МЕТОДУ БАГАТОКРИТЕРІАЛЬНОГО ПРИЙНЯТТЯ РІШЕНЬ FUZZY MARCOS ПРИ ФОРМУВАННІ РЕЙТИНГУ УНІВЕРСИТЕТІВ

В даній роботі розглядається процес рейтингування університетів, оскільки різні рейтинги мають специфічні методології, які можуть не відповідати власним пріоритетам при виборі університету для конкретної особи. Вибір університету є достатньо суб'єктивним для кожного студента, тому дану задачу можна розглянути як багатокритерійне прийняття рішень з використанням нечіткої логіки.

Для подальшого використання назви критеріїв було замінено на умовні позначення, зокрема: академічна репутація (K1), репутація від роботодавця (K2), співвідношення викладачів і студентів (K3), цитування на факультет (K4), статті в Nature та Science (K5), публікації в Scopus (K6), нагороджені випускники (K7), нагороджені співробітники (K8), дохід у галузі (K9), коефіцієнт міжнародного студентства (K10),

коефіцієнт міжнародного персоналу (K11). Вхідний набір даних для формування рейтингу наведено у таблиці 1.

Таблиця 1 – Вхідні дані

Критерії Університету	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11
Delft University of Technology	70,70	87,90	43,90	75,80	22,00	18,31	10,7	0,00	97,7	85,60	99,80
Katholieke Universiteit Leuven	83,40	55,50	9,40	77,50	25,40	28,17	0	0,00	99,20	36,10	90,70
Ludwig-Maximilians-Universität München	92,90	80,70	49,10	43,90	31,70	31,66	25,60	18,50	100,00	36,20	57,60
Ruprecht-Karls-Universität Heidelberg	80,60	41,80	99,10	34,20	30,00	28,25	18,50	24,50	56,90	49,50	52,30
Sorbonne University	90,10	47,40	38,90	49,30	30,20	49,75	37,1	26,50	38,20	57,10	34,00
The University of Warwick	74,30	92,00	47,80	51,40	15,10	13,47	13,10	28,80	41,60	99,20	98,10
University of Bristol	75,50	82,90	70,80	44,80	27,20	21,80	5,30	15,80	39,80	80,60	91,50
University of Copenhagen	72,80	38,70	100,00	31,20	37,40	29,43	19,2	18,20	54,4	26,50	91,80
University of Glasgow	73,00	66,80	60,40	33,50	21,00	20,02	5,3	0,00	40,70	97,10	93,80
University of Zurich	59,70	60,60	98,00	45,90	32,00	24,32	0,00	23,20	64,10	59,80	100,00

При дослідженні методу fuzzy MARCOS [1] було розглянуто вплив об'єктивних методів визначення ваг критеріїв на результат рейтингування, таких як:

1) ентропія [2]:

$$x_{ij}^* = x_{ij} / \sum_{i=1}^m x_{ij}, e_j = \sum_{i=1}^m X_{ij}^* \ln(X_{ij}^*) / \ln(m), w_j = (1 - e_j) / \sum_{j=1}^n (1 - e_j), i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}, \quad (1)$$

де x_{ij} - оцінка i -ї альтернативи по j -му критерію, x_{ij}^* - нормалізована оцінка i -ї альтернативи по j -му критерію, e_j -

інформаційна ентропія j -го критерію, w_j - ваговий коефіцієнт j -го критерію;

2) стандартне відхилення [2]:

$$\begin{aligned} \sigma_j &= \sqrt{\sum_{i=1}^m (x_{ij} - \bar{x}_j)^2 / m}, w_j = \\ &= \sigma_j / \sum_{j=1}^n \sigma_j, i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\} \end{aligned} \quad (2)$$

де \bar{x}_j - середня оцінка альтернатив по j -му критерію, σ_j - стандартне відхилення по j -му критерію;

3) статистична дисперсія [2]:

$$\begin{aligned} \phi_j &= \sum_{i=1}^m (x_{ij} - \bar{x}_j)^2 / m, w_j = \phi_j / \sum_{j=1}^n \phi_j, \\ i &\in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\} \end{aligned} \quad (3)$$

де ϕ_j - статистична дисперсія по j -му критерію;

4) CRITIC [2]:

$$\begin{aligned} x_{ij}^* &= (x_{ij} - x_j^{\min}) / (x_j^{\max} - x_j^{\min}), C_j = \sigma_j \sum_{k=1}^m (1 - r_{jk}), w_j = \\ &= C_j / \sum_{j=1}^n C_j, i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\} \end{aligned} \quad (4)$$

де x_j^{\min} - мінімальна оцінка по j -му критерію, x_j^{\max} - максимальна оцінка по j -му критерію, r_{jk} - коефіцієнт кореляції між j -м та k -м критеріями, C_j - кількість інформації, що входить до j -го критерію;

5) MEREC [3]:

$$\begin{aligned} x_{ij}^* &= \min_i x_{ij} / x_{ij}, S_i = \ln \left(1 + \frac{1}{n} \sum_{j=1}^n |\ln(x_{ij}^*)| \right), S'_{ij} = \\ &= \ln \left(1 + \frac{1}{n} \sum_{k=1, k \neq j}^n |\ln(x_{ik}^*)| \right), i \in \{1, \dots, m\}, j \in \{1, \dots, n\} \end{aligned} \quad (5)$$

$$\begin{aligned} E_j &= \sum_{i=1}^m |S'_{ij} - S_i|, w_j = E_j / \sum_{j=1}^n E_j, \\ i &\in \{1, \dots, m\}, j \in \{1, \dots, n\} \end{aligned} \quad (6)$$

де S_i - загальна продуктивність i -ї альтернативи, S'_{ij} - загальна продуктивність i -ї альтернативи при вилученні j -го критерію, E_j - ефект

від вилучення j -го критерію.

Результати рейтингування університетів методом fuzzy MARCOS з використанням даних методів визначення ваг критеріїв наведено у таблиці 2.

Таблиця 2 – Результати методу fuzzy MARCOS

Університет и	Ентропія	Стандартне відхилення	Статистична дисперсія	CRITIC	MEREC
Delft University of Technology	7	1	1	1	8
Katholieke Universiteit Leuven	10	10	9	10	10
Ludwig- Maximilians- Universität München	4	5	6	3	6
Ruprecht- Karls- Universität Heidelberg	2	7	7	7	3
Sorbonne University	1	9	10	6	2
The University of Warwick	3	3	3	4	5
University of Bristol	8	4	4	5	7
University of Copenhagen	6	8	8	8	4
University of Glasgow	9	6	5	9	9
University of Zurich	5	2	2	2	1

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Stanković M, Stević Ž, Das DK, Subotić M, Pamučar D. A New Fuzzy MARCOS Method for Road Traffic Risk Analysis. Mathematics. 2020; 8(3):457. URL: <https://doi.org/10.3390/math8030457>.

2. Paradowski, B.; Shekhovtsov, A.; Baćkiewicz, A.; Kizielewicz, B.; Sałabun, W. Similarity Analysis of Methods for Objective

Determination of Weights in Multi-Criteria Decision Support Systems. Symmetry 2021, 13, 1874. URL: <https://doi.org/10.3390/sym13101874>.

3. Keshavarz-Ghorabae, M.; Amiri, M.; Zavadskas, E.K.; Turskis, Z.; Antucheviciene, J. Determination of Objective Weights Using a New Method Based on the Removal Effects of Criteria (MEREC). Symmetry 2021, 13, 525. URL: <https://doi.org/10.3390/sym13040525>.

УДК 004.02

Лубковський В. О., Кондратенко Ю. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

СППР ДЛЯ ОЦІНЮВАННЯ ПРОЄКТІВ ЕЛЕКТРОСТАНЦІЙ НА ОСНОВІ МЕТОДУ FUZZY BEST-WORST

Вигоди та витрати електростанцій, включаючи їхній вплив на навколишнє середовище, залежать від їх технології та від того, скільки електроенергії насправді виробляє кожна станція. Для оцінювання проєктів електростанцій, слід враховувати місце діяльності людини для забезпечення суспільного задоволення та економічної вигоди у великих масштабах, поваги до екологічних проблем, здоров'я громади.

Метою дослідження є оцінювання проєктів електростанцій з подальшим вибором найкращого, з використанням методу Fuzzy Best-Worst (FBW).

У цій роботі вирішується задача групового прийняття рішень (GDM) за допомогою багатогранного лінгвістичного підходу на основі Fuzzy Best-Worst методу.

Діаграма процесу прийняття рішень зображена на рисунку 1.

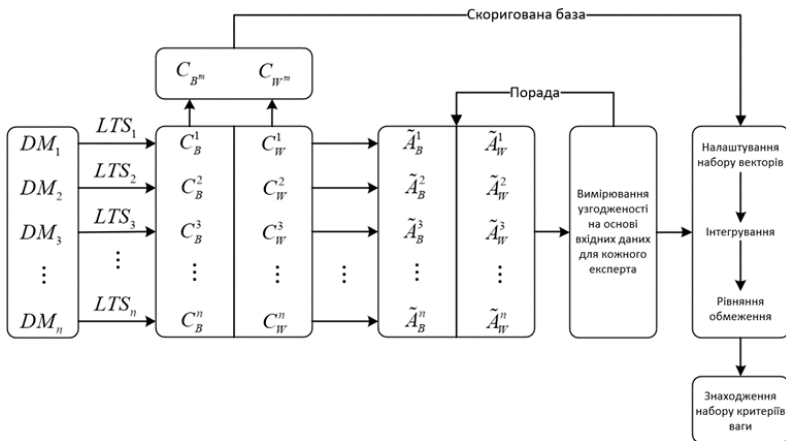


Рисунок 1 – Діаграма процесу прийняття рішень

У реалізованому способі обираються лінгвістичні набори термів (LTS) та надаються експертам, щоб ті висловили свою індивідуальну думку у вигляді нечіткої оцінки. Потім вдосконалений FBW метод використовується для обчислення ваги критеріїв із формою нечітких чисел.

З метою визначення нечіткої ваги критеріїв виконано нечітке порівняння за відносними критеріями. Для визначення нечітких ваг альтернатив щодо різних критеріїв, пов'язані альтернативи слід нечітко порівнювати за кожним критерієм. Оцінки нечіткого ранжування альтернатив можуть бути отримані з нечітких ваг альтернатив щодо різних критеріїв, помножених на нечіткі ваги відповідних критеріїв [1].

Коли особа, яка приймає рішення (DM), надає попарні порівняння, важливо перевірити прийнятну невідповідність і забезпечити раціональність оцінки.

Щоб перевірити, наскільки непослідовним може бути повний набір попарних порівнянь, можна використати кілька індексів узгодженості. Одним з методів є вимірювання консистенції на основі вхідних даних, яке є простим у використанні та має кілька бажаних властивостей [2]. Використовуючи простий розрахунок вимірювання консистенції на основі вхідних даних, легко надати DM негайного зворотного зв'язку. Коефіцієнт узгодженості можна отримати після завершення всього процесу виявлення, що означає, що він дає DM чітке й негайне уявлення про його рівень узгодженості.

Саме завдяки методу Fuzzy Best-Worst розроблена СППР для оцінювання проєктів електростанцій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Y. Liu, C. M. Eckert, and C. Earl, "A review of fuzzy ANP methods for decision-making with subjective judgements," *Expert Syst. Appl.*, vol. 161, Dec. 2020, Art. no. 113738.
2. F. Liang, M. Brunelli, and J. Rezaei, "Consistency issues in the best worst method: Measurements and thresholds," *Omega*, vol. 96, Oct. 2020, Art. no. 102175.

УДК 004.42

Паленко Р. О., Сіденко Є. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

КОМБІНОВАНИЙ FUZZY TOPSIS-DEMATEL ДЛЯ БАГАТОКРИТЕРІЙНОГО ПРИЙНЯТТЯ РІШЕНЬ В МЕДИЦИНІ

У наш час, новітніх технологій та стрімкого інформаційного потоку, все більше виникає задач вибору кращої альтернативи. Тому вже майже неможливе прийняти зважене рішення не користуючись тим чи іншим методом підтримки прийняття рішень. Сфери застосування даних методів необмежені, але в наш час світової пандемії, необхідно звернути увагу на сферу медицини. Кожного року на ринок виходять нові лікувальні засоби, мікстури, таблетки. Лікарям доступні різні шляхи лікування багатьох захворювань. Пацієнт може самостійно обирати яку вакцину собі вколоти. Кожна з цих проблем має безліч альтернатив, що лише збільшується, а з плином часу та проведенням нових досліджень кількість критеріїв що позитивно чи негативно впливають на альтернативи зростає.

Метою роботи є дослідження можливості комбінування методів FUZZY TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) та DEMATEL (DEcision MAKing Trial and Evaluation Laboratory) для покращення прийняття рішення при виборі найкращої вакцини від COVID-19.

Первинною задачею став пошук підходящого датасету з експертними оцінками кожної вакцини. Тому базуючись на попередніх дослідженнях про вакцини: їх властивості, недоліки та переваги, було сформовано необхідний датасет (рис. 1). Експертам в сфері медицини були надані таблиці для заповнення даних, що характеризують кожену

вакцину за обраними критеріями.

	Crispr1					BNT162b2					mRNA1273				
	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5
Ефективність	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE	VE
Температура зберігання	f	f	p	f	e	VP	VP	p	VP	f	p	f	VE	f	e
Кількість учасників клінічних дослідів 3 фази	VE	VE	e	f	f	VE	e	e	e	VE	VE	VE	e	VE	e
Кількість країн	e	e	f	p	f	VE	VE	e	VE	VE	e	e	f	VE	f
Кількість Вакцинованих	VE	VE	f	VE	f	VE	VE	VE	VE	VE	VE	e	VE	e	e
Побочний вплив	VE	VE	e	e	VE	VP	f	p	p	p	VP	f	p	e	p
	AIZ6.COV2.5					CmAdOx1.5					BBIBP-CoV				
	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5
Ефективність	e	e	VP	f	p	e	f	p	p	f	e	e	f	f	e
Температура зберігання	p	e	VE	e	e	VE	VE	f	e	VE	VE	VE	f	VE	VE
Кількість учасників клінічних дослідів 3 фази	VE	e	e	e	VE	f	p	p	VP	VP	e	e	e	VE	e
Кількість країн	e	e	f	VE	f	VE	e	e	VE	VE	p	p	f	p	f
Кількість Вакцинованих	p	p	p	p	VP	e	e	f	VE	e	p	p	e	VP	p
Побочний вплив	VE	VE	f	VE	e	VE	VE	f	VE	e	e	e	VP	VE	VP
	Gam-COVID-vac					NVX-CoV2373					CoronaVac				
	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5	Експерт 1	Експерт 2	Експерт 3	Експерт 4	Експерт 5
Ефективність	VE	e	VE	VE	VE	VE	VE	VE	VE	e	e	VP	VP	p	VP
Температура зберігання	p	e	VE	VE	e	VE	VE	f	VE	VE	VE	VE	f	VE	VE
Кількість учасників клінічних дослідів 3 фази	e	e	f	f	p	f	f	f	e	p	f	f	p	f	p
Кількість країн	p	p	p	p	p	p	p	p	VP	VP	p	VP	p	VP	p
Кількість Вакцинованих	VP	VP	VP	p	p	VP	VP	p	VP	p	p	p	f	VP	f
Побочний вплив	VE	e	f	f	e	e	VP	f	p	f	e	e	f	VE	f

Рисунок 1 – Таблиця експертних оцінок

Точкою комбінування FUZZY TOPSIS та DEMATEL було обрано крок визначення рішень FPIS (Fuzzy Positive Ideal Solution) та FNIS (Fuzzy Negative Ideal Solution) в методі TOPSIS. На ньому можна провести трансформацію параметра залежного від оцінок критеріїв. Для цього були використанні вихідні параметри методу DEMATEL: D+R та D-R, що позначають визначеність та відношення критеріїв. Саме завдяки цьому ми можемо скорегувати базове значення параметру важливості критеріїв, отриманих методом TOPSIS, та зробити результати більш точними.

В ході досліджень можливих способів взаємодії методів FUZZY TOPSIS та DEMATEL, було створено 5 моделей їх комбінування. Також були отримані результати, в яких для визначення FPIS та FNIS взагалі не були використані дані методу TOPSIS, а лише трансформовані виходи DEMATEL.

В результаті дослідження було виявлено позитивний вплив від комбінування методів. Таким чином, комбінуючи FUZZY TOPSIS та DEMATEL, не тільки найкращий результат (рішення) стає більш чітко виражений, але й серед інших з'являється позитивна кореляція.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Zulqarnain, R.M., et. al. Selection of Medical Clinic for Disease Diagnosis by Using TOPSIS Method. International Journal of Pharmaceutical Sciences Review and Research, 61(1), pp. 22-27, 2020.
2. Ehsanifar, M. Applying Fuzzy DEMATEL Method to Analyze Supplier Selection Criteria. International Research Journal of Finance and

Economics, pp. 76-86, 2013.

3. Cayvaz F., Tuzkaya G., Kalender Z.T., Kilic H.S. Analysis of Supply Chain Disruption Factors Under the Effect of COVID-19 Pandemic via Neutrosophic Fuzzy DEMATEL. In: Kahraman C., Cebi S., Cevik Onar S., Oztaysi B., Tolga A.C., Sari I.U. (eds) Intelligent and Fuzzy Techniques for Emerging Conditions and Digital Transformation. INFUS 2021. Lecture Notes in Networks and Systems, vol 308. Springer, Cham, 2022. https://doi.org/10.1007/978-3-030-85577-2_41.

УДК 004.9

Стовманенко В. О., Давиденко Є. О.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ОГЛЯД РЕАЛІЗАЦІЙ СИСТЕМ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

Системи підтримки прийняття рішень (СППР) дозволяють використовувати експертні знання без залучення великої кількості людей, які мають оцінити ситуацію і сформувані колективне рішення. СППР на основі нечіткої логіки, нейронних мереж, генетичних алгоритмів або інших алгоритмів інтелектуальної обробки даних набули розповсюдження у різних галузях [1]. В цілому їх можна розділити на декілька категорій:

– *Серверні СППР.* Виконуються на віддаленому сервері, дозволяють виконувати розрахунки на декількох фізичних пристроях, використовують окремі підсистеми для збереження даних (бази даних, OLAP-сховища, спеціалізовані сховища) [2], мають спеціальний інтерфейс для користувачів (HTTP, gRPC тощо) з підтримкою одночасного доступу до системи багатьох користувачів одночасно.

– *Локальні СППР.* Виконуються на локальному пристрої, в межах операційної системи, реалізують певний інтерфейс користувача з яким може працювати тільки один користувач у конкретний проміжок часу (при тому може існувати можливість запустити декілька екземплярів програмного забезпечення на тому ж комп'ютері), усі підсистеми об'єднані в єдиний програмний комплекс (можуть бути виключення для сховища даних, яке може бути віддаленим).

– *Вбудовані СППР.* Виконуються на мікропроцесорі, який часто містить обмежені обчислювальні ресурси та має обмеження на кількість енергії, які він може спожити. Такі системи не надають інтерфейсу користувача, програмуються заздалегідь і містять

мінімальне сховище для збереження даних. Можуть слугувати проміжною ланкою для СППР іншого типу. Вхідні дані в таких системах часто збираються безпосередньо з сенсорів, які вимірюють їх.

Ці особливості СППР характерні для кожної системи відповідно до типу, не залежно від ядра системи та алгоритму, який виконує виведення конкретного рішення. Кожна система має отримувати дані з певного джерела для обробки і виконувати певні розрахунки та перетворення даних для того аби ці розрахунки виконати.

Проектування СППР залежить від вхідних даних та бажаних вихідних даних. При тому для кожної системи можна виділити декілька основних напрямів, які вимагають уваги і можуть бути оптимізованими:

– *Розподілення*. Чи підтримує алгоритм прийняття рішення можливість розділити вхідні дані й виконувати частини розрахунків паралельно? Чи може інфраструктура, що використовується системою працювати в паралельному режимі? (для вбудованої системи, де є лише одне обчислювальне ядро, не можна розподілити вхідні дані, всі розрахунки мають бути виконані послідовно і тільки потім можна буде отримати певне рішення [3], у той час як серверна система, що часто містить процесор з багатьма обчислювальними ядрами і має змогу залучати системи обміну повідомленнями для виконання на декількох пристроях одночасно, може опрацьовувати одночасно дані з багатьох джерел).

– *Робота із старими даними*. Чи потрібні попередні дані для прийняття поточних рішень? Який обсяг старих даних треба обробити і як швидко? (серверні системи можуть використовувати OLAP куби для швидкого доступу до даних і можуть утримувати значні, >1 ТіБ, обсяги даних, у той час як можливості локальних СППР більш обмежені, фізичні накопичувачі часто не мають значного достатнього обсягу та можуть бути не настільки швидкими).

– *Інтеграція*. Чи є СППР ізольованим компонентом, що виконується самостійно? З якими системами потрібна інтеграція? (локальні системи вимагають реалізації драйверів для пристроїв, які мають виконувати певне завдання, в залежності від результату роботи системи, у той час як для вбудованих така ж інтеграція може бути значно простішою; в той же час локальні системи можуть відносно просто використовувати мережу для звітування про свою роботу або зв'язок з іншими сервісами, в той час як реалізація подібного функціоналу для вбудованої системи вимагатиме значної кількості ресурсів, які не можна буде задіяти для основного завдання).

Кожен тип СППР має потенціал для оптимізацій, але не завжди один тип оптимізацій буде доречним для іншої системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Верес О. М. Види архітектури систем підтримки прийняття рішень. Вісник Національного університету «Львівська політехніка» : Комп'ютерні системи проектування. Теорія і практика, 2010. № 685. С. 190–197.
2. Нестеренко О. В., Савенков О. І., Фаловський О. О. Інтелектуальні системи підтримки прийняття рішень : навч. посібн. за ред. П. І. Бідюка. Київ : Національна академія управління, 2016. 188 с.
3. Catania V., Ascia G., Palesi M., Patti D., Di Nuovo A. G. Fuzzy decision making in embedded system design. Proceedings of the 4th International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS '06), 2006. pp. 223–228. doi: 10.1145/1176254.1176309.

УДК 004.02

Таранчук Д. О., Кондратенко Г. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

РЕКОМЕНДАЦІЙНА СИСТЕМА ДЛЯ ВИБОРУ МОБІЛЬНИХ ПРИБОРІВ НА ОСНОВІ МЕТОДІВ ПРИЙНЯТТЯ РІШЕНЬ

В даній роботі розглядається процес прийняття рішень з вибору оптимальних мобільних пристроїв. В якості мобільного пристрою розглядається мобільний телефон. Актуальність дослідження визначається складністю формування оцінки та вибору мобільного пристрою через різноманіття та багатогранність відомої інформації про можливі альтернативи, складністю виділення найбільш важливих та впливових критеріїв для подальшого прийняття рішень. Метою роботи є синтез технології оцінки мобільних пристроїв на основі методів прийняття рішень з урахуванням великої кількості критеріїв.

Нехай дано 7 альтернатив – мобільних пристроїв: Xiaomi Mi 11 Ultra, Oppo Find X3 Pro, Huawei P50 Pro, Google Pixel 6 Pro, Vivo X70 Pro+, Apple iPhone 13 Pro Max, Vivo X70 Pro (MediaTek).

Які необхідно оцінити за 6 критеріями: основна камера (якість, виробник, функціональність, тощо) Q_1 , фронтальна камера (якість, виробник, функціональність, тощо) Q_2 , аудіо Q_3 , дисплей Q_4 , батарея Q_5 , безпека та збереженість даних Q_6 . За допомогою анкетування було отримано оцінки кожного мобільного пристрою за 6 критеріями від 6 експертів. Агрегована експертна оцінка має наступний вигляд:

$$e_i^{agr} = \sum_{n=1}^N K_i e_i, \quad (1)$$

де N – кількість експертів; K_i – нормований коефіцієнт компетентності експертів;

e_i – оцінка, надана експертом за i -м критерієм.

Таблиця 1 – Матриця рішень щодо оцінки альтернатив

Мобільні пристрої	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6
Xiaomi Mi 11 Ultra	4,43	3,93	3	3	3,95	3,82
Oppo Find X3 Pro	4,16	3,72	3,43	3,24	3,85	3
Huawei P50 Pro	5	4,62	3,28	2,68	3,32	3,51
Google Pixel 6 Pro	3,96	3,13	2,94	2,9	2,78	2,99
Vivo X70 Pro+	4,66	4,31	2,74	2,12	2,42	2,49
Apple iPhone 13 Pro Max	4,85	4,62	4,07	3,5	4,34	4,73
Vivo X70 Pro (MediaTek)	4,24	4,1	3,56	2,52	2,75	3,3

Використаємо отриману матрицю рішень щодо оцінки мобільних пристроїв для вирішення поставленої задачі методом VIKOR.

Далі розрахуємо за формулою «ідеальні» та «найгірші» значення за кожним із критеріїв:

$$f_i^* = \max_j f_{ij}, \quad f_i^- = \min_j f_{ij}. \quad (2)$$

де f_{ij} – оцінка j -ї альтернативи по i -му критерію.

Таблиця 2 – «Ідеальні» та «найгірші» значення за кожним із критеріїв

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6
f_i^*	5	4,62	4,07	3,5	4,34	4,73
f_i^-	3,96	3,13	2,74	2,12	2,42	2,49

Розрахуємо значення відхилень критеріїв від «ідеального» значення за формулами:

$$S_j = \sum_{i=1}^n \frac{w_i (f_i^* - f_{ij})}{(f_i^* - f_i^-)}, \quad R_j = \max_i \left[\frac{w_i (f_i^* - f_{ij})}{(f_i^* - f_i^-)} \right], \quad (3)$$

де w_i – ваговий коефіцієнт i -го критерію, наприклад, методом простого ранжування.

Таблиця 3 – Значення відхилень критеріїв від «ідеального» значення

	$\frac{w_i(f_i^* - f_{ij})}{(f_i^* - f_i^-)}$						S_j	R_j
Xiaomi Mi 11 Ultra	0,08 8	0,07 4	0,08 8	0,09 1	0,05 1	0,02 8	0,42	0,09 1
Oppo Find X3 Pro	0,12 9	0,09 7	0,05 3	0,04 7	0,06 4	0,05 4	0,44 4	0,12 9
Huawei P50 Pro	0	0	0,06 5	0,14 9	0,13 3	0,03 8	0,38 5	0,14 9
Google Pixel 6 Pro	0,16	0,16	0,09 3	0,10 9	0,20 3	0,05 4	0,78	0,20 3
Vivo X70 Pro+	0,05 2	0,03 3	0,11	0,25	0,25	0,07	0,76 6	0,25
Apple iPhone 13 Pro Max	0,02 3	0	0	0	0	0	0,02 3	0,02 3
Vivo X70 Pro (MediaTek)	0,11 7	0,05 6	0,04 2	0,17 8	0,20 7	0,04 5	0,64 4	0,20 7

Далі проводиться розрахунок значень Q_j , що дозволять ранжувати альтернативи з урахуванням ваги ν , яка визначає ступінь впливу на прийняття рішення двох підходів: краща альтернатива визначається за правилом «більшості» ($\nu > 0,5$) або краща альтернатива визначається на основі максимальної «групової корисності» ($\nu < 0,5$), при інших умовах приймається ($\nu = 0,5$).

Таблиця 4 – Фінальне ранжування за методом VIKOR

	E_1	E_2	E_3	E_4	E_5	E_6	E_7	Ранжування
S_j	0,42	0,444	0,385	0,78	0,766	0,023	0,644	E6 \succ E3 \succ E1 \succ E2 \succ E7 \succ E5 \succ

								E4
R_j	0,091	0,129	0,149	0,203	0,25	0,023	0,207	E6 γ E1 γ E2 γ E3 γ E4 γ E7 γ E5
Q_j	0,412	0,512	0,517	0,896	0,991	0	0,815	E6 γ E1 γ E2 γ E3 γ E7 γ E4 γ E5

В таблиці 5 наведено результати застосування різних методів багатокритерійного прийняття рішень.

Таблиця 5 – Аналіз застосування методів багатокритерійного прийняття рішень

	Методи MCDM						
	VIKOR	TOPSIS	ELECTRE	MOORA	SAW	ISM	LCM
Xiaomi Mi 11 Ultra							
Oppo Find X3 Pro							
Huawei P50 Pro							
Google Pixel 6 Pro							
Vivo X70 Pro+							
Apple iPhone 13 Pro Max	*	*	*	*	*	*	*
Vivo X70 Pro (MediaTek)							

З результатів видно, що найкращим є альтернативне рішення Apple iPhone 13 Pro Max.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Q. Jianxun, Z. Zhiguang and K. Feng, "Selection of Suppliers based on VIKOR algorithm," 2007 Chinese Control Conference, 2007, pp. 146-148, doi: 10.1109/CHICC.2006.4347102.

УДК 004.211

Франчук О. В., Кондратенко Ю. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПЛАНУВАННЯ ТРАНСПОРТНИХ МАРШРУТІВ ДЛЯ ЗАДАЧ CVRP В УМОВАХ НЕВИЗНАЧЕНОСТІ

Однією з базових галузей економіки є транспортна, яка забезпечує розвиток бізнесу, створює передумови для задоволення потреб клієнтів у наданні транспортних послуг і включає морські порти та річкові термінали, розгалужену мережу залізниць, аеропорти, автомобільні шляхи та авіаційні сполучення.

Вантажні перевезення підвищують ефективність економіки, сприяють розширенню транспортних сполучень та створенню робочих місць, забезпечують ефективне функціонування і розвиток промисловості.

За даними Державної служби статистики за 11 місяців 2021 року транспортні компанії перевезли 566,1 млн. тонн вантажів, що на 3,6% більше аналогічного періоду минулого року, в тому числі: залізничним транспортом - 287 млн. тонн, що на 2,9% перевищує показник минулого року, автомобільним - 204 млн. тонн (ріст на 16,9%). Дещо знизились вантажні перевезення водним (на 5,7%) та авіаційним (на 4,8%) транспортом.

В провідних промислово-розвинених країнах одержання 20-30% валового національного продукту пов'язане з логістичними системами. Згідно зарубіжного досвіду, збільшення обсягів продажів фірми на 10% можливо досягти лише за допомогою зменшення логістичних витрат на 1%.

Впровадження логістичного менеджменту дозволяє компаніям суттєво прискорити оборотність капіталу, знизити собівартість виробництва, забезпечити задоволення попиту споживачів на товари та сервіси.

В Україні відстежується низький рівень цифровізації та автоматизації транспортної галузі і сама сфера вантажоперевезень потребує створення цифрових продуктів для учасників вантажообігу.

Національна транспортна стратегія України на період до 2030 року, схвалена розпорядженням Кабінету Міністрів України від 30 травня 2018 р. № 430-р., серед основних завдань, які необхідно виконати для успішного функціонування транспортної галузі передбачає:

- впровадження інтелектуальних транспортних систем та систем управління рухом на наземному та водному транспорті;
- використання у транспортно-дорожньому комплексі інноваційних технологій.

Наразі вже вирішено значне коло задач, пов'язаних з оптимізацією транспортних вантажопотоків, проте недостатньо розроблені алгоритми та математичні моделі транспортної логістики в умовах невизначеності з врахуванням різних програм замовлень.

Задача маршрутизації транспортних засобів (CVRP) – це комбінаторна задача оптимізації, в якій надається мережа клієнтів із заданими вимогами. Мета полягає в тому, щоб знайти набір маршрутів, які починаються та закінчуються у вузлі депо, при цьому кожен транспортний засіб має обмежену місткість для вантажоперевезень. Ці маршрути повинні проходити таким чином, щоб запити всіх клієнтів у мережі були задоволені, а витрати, пов'язані з обходом цих маршрутів, були мінімальними.

У реальних ситуаціях попит на будь-який товар залежить від різних неконтрольованих факторів, таких як сезон, час доставки, ринкові умови та багато іншого. Через ці фактори не завжди можна повідомити про попит заздалегідь, а точну інформацію щодо попиту майже неможливо отримати. Тому в реальному житті вимоги клієнтів завжди бувають неточними і випадковими. Рішення, прийняті клієнтами щодо вимог, також можуть мати певні вагання. Для того, щоб задовольнити такі запити клієнтів у мережі, у магістерській магістерській кваліфікаційній роботі використовуються нечіткі випадкові величини (fuzzy demands). Клієнт може зробити замовлення використовуючи невизначені терміни, такі як «приблизно N_1 », «між N_2 і N_3 », «принаймні N_4 », «не менше N_5 », «не більше N_6 », де N_i – деяка величина, яка відображає обсяг замовлення. Такі терміни можна представити нечіткими числами, наприклад, як нечіткі числа з трикутною функцією належності. Була представлена математична модель, що відповідає CVRP з нечіткими вимогами (CVRPFD – Capacitated Vehicle Routing Problem in Conditions of Fuzzy Demands). Використання CVRPFD дає змогу ефективно вирішувати завдання побудови маршрутів транспортування в умовах невизначеності, гнучко враховуючи вимоги до них [1].

Щоб розширити інтерфейс користувача було прийнято рішення використовувати Google Maps Platform. Даний API дозволяє інтегрувати

google maps в застосунок. Перевагою використання карт є можливість вибору існуючих пунктів призначення та відстаней між даними точками. За допомогою карти можливо обирати місце розташування замовників, складу, відображати сплановані маршрути.

Задачі, вирішені у даному дослідженні дозволяють полегшити компаніям з управління логістикою здійснювати формування маршрутів, якими слід керуватися для отримання мінімальних операційних витрат і максимального прибутку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. V. P. Singh, Kirti Sharma, Debjani Chakraborty. Solving Capacitated Vehicle Routing Problem with Demands as Fuzzy Random Variable. 2021. 19 p.
2. Werners B., Kondratenko Y. Alternative Fuzzy Approaches for Efficiently Solving the Capacitated Vehicle Routing Problem in Conditions of Uncertain Demands. - Springer International Publishing AG 2018.
3. Конференция Организации Объединенных Наций по торговле и развитию - TD/B/68/2, 12 September 2018. С. 18 - URL: https://unctad.org/system/files/official-document/cimem7d17_ru.pdf

УДК 004.42

Юр'єва А. О., Кондратенко Г. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ДОСЛІДЖЕННЯ ВПЛИВУ ВИРОБНИЦТВА ХАРЧОВИХ ПРОДУКТІВ НА ДОВКІЛЛЯ З ВИКОРИСТАННЯМ МЕТОДІВ БАГАТОКРИТЕРІЙНОГО ПРИЙНЯТТЯ РІШЕНЬ

Зміна клімату впливає на життя та здоров'я людей різними способами. Під зміною клімату розуміється зміна умов навколишнього середовища. Причини кліматичних змін поділяються на природні та антропогенні, тобто викликані діяльністю людини. Основним чинником зміни клімату є парниковий ефект - процес, за якого парникові гази затримують сонячну енергію на поверхні Землі та в атмосфері і перешкоджають її поверненню назад у космос. Більшість з цих парникових газів виникають у природі, але діяльність людини (генерація електроенергії, промисловість, транспорт, будівлі, сільське господарство) збільшує концентрацію деяких з них в атмосфері. На

виробництво харчових продуктів припадає приблизно 26% (1/4) глобальних викидів парникових газів. Незалежно від місця, де людина купує їжу або продуктів, які вона вирішить їсти, харчування кожного з нас матиме значний вплив на навколишнє середовище. Тому було вирішено дослідити саме вплив виробництва харчових продуктів на довкілля. *Метою роботи* є дослідження впливу виробництва харчових продуктів на довкілля з використанням багатокритерійних методів прийняття рішень, визначення найбільш вагомих критеріїв оцінки впливу та харчових продуктів, виробництво яких найбільше шкодить навколишньому середовищу.

Багатокритерійне прийняття рішень (MCDM – multiple-criteria decision-making) є дослідницькою сферою, яка охоплює повсякденне життя, соціальні науки, інженерію, медицину, та багато інших областей. Воно допомагає структурувати та формалізувати процес прийняття рішень у прозорий і послідовний спосіб. Коли існує багато альтернатив для однієї проблеми, важливо знайти найбільш підходящу альтернативу з найкращими значеннями відповідно до критеріїв. Проте мета MCDM полягає не в тому, щоб запропонувати найкраще рішення, а в тому, щоб допомогти особам, які приймають рішення, вибрати одну або декілька альтернатив, які відповідають їхнім вимогам та вподобанням.

Існує багато методів, які можна використовувати для розв'язування задач. Кожен метод MCDA має свій власний алгоритм обчислення, за допомогою якого альтернативи ставляться в чергу, і неможливо стверджувати, що використання конкретних методів з однаковими вхідними даними призведе до того ж кінцевого результату.

Перед початком дослідження було розглянуто наступні методи:

- Analytic Hierarchy Process (AHP);
- Élimination et Choix Traduisant la REalité (ÉLECTRE);
- Vlekriterijumsko KOMpromisno Rangiranje (VIKOR);
- Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS);
- Preference Ranking Organization METHod for Enrichment Evaluation (PROMETHEE).

Після аналізу відповідних методів було вирішено, що метод AHP є найбільш підходящим для вирішення даного завдання. Метод аналізу ієрархій (MAI, AHP (англ.)) – це метод організації та аналізу складних рішень із використанням математики та психології. Він був розроблений Томасом Л. Сааті в 1970-х роках і з тих пір вдосконалювався. Він містить три частини: кінцеву мету або проблему, яку намагаються вирішити, усі можливі рішення, які називаються альтернативами, і критерії, за якими будуть оцінюватися альтернативи. AHP забезпечує раціональну основу для необхідного рішення шляхом

кількісної оцінки його критеріїв і альтернативних варіантів, а також для зв'язку цих елементів із загальною метою.

Для виконання даного дослідження був знайдений відповідний набір даних «Environment Impact of Food Production». Набір даних містить оцінки впливу на середовище окремих харчових продуктів відповідно до критеріїв. Так як кількість продуктів (альтернатив) є досить великою, було вирішено редагувати цей набір та залишити ті продукти, які виготовляються на території нашої країни. Після проведеного редагування цей набір містить 21 продукт, кожен із яких оцінюється по 11 критеріям. Всі продукти були відсортовані на 5 підгруп: овочі та фрукти, м'ясо та риба, молочні продукти, продукти зернового походження та інші.

Під час виконання дослідження було визначено, що група «М'ясо та риба» має найбільший негативний вплив на середовище відповідно до наявних критеріїв (рис. 1). Аналогічні дослідження також були проведені в середині кожної з груп щодо відповідних продуктів (Таблиця 1).

Для зручної роботи було створено програмну реалізацію за допомогою Angular.

#6: Results

$F \& V = 0.059 * 0.048 + 0.016 * 0.055 + 0.079 * 0.043 + 0.054 * 0.035 + 0.046 * 0.298 + 0.031 * 0.053 + 0.020 * 0.073 + 0.172 * 0.036 + 0.108 * 0.072 + 0.140 * 0.036 + 0.275 * 0.115 = 0.07647406966538639$

$M \& F = 0.059 * 0.504 + 0.016 * 0.609 + 0.079 * 0.603 + 0.054 * 0.592 + 0.046 * 0.509 + 0.031 * 0.289 + 0.020 * 0.442 + 0.172 * 0.576 + 0.108 * 0.535 + 0.140 * 0.584 + 0.275 * 0.504 = 0.5373027138957855$ - has the biggest impact

$DP = 0.059 * 0.140 + 0.016 * 0.225 + 0.079 * 0.213 + 0.054 * 0.203 + 0.046 * 0.060 + 0.031 * 0.079 + 0.020 * 0.289 + 0.172 * 0.209 + 0.108 * 0.278 + 0.140 * 0.217 + 0.275 * 0.293 = 0.2275349116269659$

$CP = 0.059 * 0.028 + 0.016 * 0.055 + 0.079 * 0.070 + 0.054 * 0.102 + 0.046 * 0.092 + 0.031 * 0.289 + 0.020 * 0.123 + 0.172 * 0.071 + 0.108 * 0.072 + 0.140 * 0.054 + 0.275 * 0.061 = 0.07347541896072601$

$Other = 0.059 * 0.279 + 0.016 * 0.055 + 0.079 * 0.070 + 0.054 * 0.067 + 0.046 * 0.040 + 0.031 * 0.289 + 0.020 * 0.073 + 0.172 * 0.108 + 0.108 * 0.043 + 0.140 * 0.109 + 0.275 * 0.028 = 0.08521288585113629$

Рисунок 1 – Вибір групи продуктів з найбільшим впливом на довкілля

Таблиця 1 – Виявлення продуктів з найбільшим впливом на довкілля

Група		Група		Група		Група		Група	
Овочі та Фрукти		М'ясо та риба		Молочні		Зернові		Інші	
Назва	Оцінка	Назва	Оцінка	Назва	Оцінка	Назва	Оцінка	Назва	Оцінка
Яблука	0.07408	Яловичина	0.27416	<u>Сир</u>	<u>0.6641</u>	Вівсянка	0.20340	<u>Чорний шоколад</u>	<u>0.7786</u>
Ягоди	0.16661	Риба	0.15708	Яйця	0.17356	<u>Соняш. олія</u>	<u>0.6028</u>	Вино	0.22138
<u>Кукурудза</u>	<u>0.27456</u>	<u>Баранина</u>	<u>0.33789</u>	Молоко	0.16232	Пшениця і жито	0.19373		
Цибуля	0.05034	Свинина	0.15021						
Горох	0.19619	Птиця	0.08067						
Картопля	0.06143								
Коренеплоди	0.04942								
Помідори	0.12734								

З результатів (рис. 1, Таблиця 1) видно, що кожна група продуктів має свій вплив на довкілля, є і певні продукти, вирощування яких слід зменшити для зниження шкідливого впливу на довкілля.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ishizaka A., Nemery P. Multi-Criteria Decision Analysis. John Wiley & Sons, 2013. 293 p.
2. Саати Т. Принятие решений. Метод анализа иерархий: пер. с англ. М., 1993. 316 с.
3. Alessio I., Ashraf L. Analytic Hierarchy Process and Expert Choice: Benefits and Limitations / I. Alessio, L. Ashraf.: ORInsight, Vol. 22(4), 2009. pp. 201–220.

Автоматизація та інтегровані комп'ютерні технології

УДК 681.5

Войтасик А. М.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

РОЗРОБКА ЛАБОРАТОРНОГО СТЕНДУ ПО ДОСЛІДЖЕННЮ ПРИНЦИПУ РОБОТИ ЕЛЕКТРОАКУСТИЧНОГО ПЕРЕТВОРЮВАЧА

Лабораторний стенд призначений для його використання в рамках проведення навчального процесу зі студентами. Він надає можливості: закріплення теоретичних знань студентів шляхом отримання практичного досвіду роботи з електроакустичним перетворювачем; вивчення конструкції та принципу роботи електроакустичного перетворювача на базі конденсаторного електретного мікрофону; дослідження процесу перетворення акустичних коливань в електричні; аналізу осцилограм перехідних процесів.

До основних складових частин лабораторного стенду належать: АС/DC перетворювач напруги; ноутбук (НБ); підсилювач потужності звукової частоти (ППЗЧ); світловий модуль (СМ); мікрофон (МК); акумуляторна батарея (АКБ); динамік (ДН).

АС/DC перетворювач напруги необхідний для перетворення змінної напруги 220 В, 50 Гц в постійну напругу 18 В для забезпечення живлення таких споживачів електричної енергії як НБ, ППЗЧ, СМ. Основою лабораторного стенду є МК та ДН розмішені всередині звукоізольованої конструкції стенду. Для підсвічування цих складових застосовується СМ.

Порядок дослідження наступний. З НБ надходить звуковий сигнал на ППЗЧ, який у подальшому підсилюється та подається на ДН [1-3]. Звукові коливання з ДН надходять на МК, автономне живлення якого реалізується застосуванням LiFePO₄ АКБ 3,7 В. Далі вже перетворений акустичний сигнал надходить до НБ, на якому за допомогою спеціалізованого програмного забезпечення можливо отримувати осцилограми дослідженого процесу перетворення для подальшого їх аналізу. В якості звукових доріжок, що будуть відтворюватися на ДН застосовується різний тональний сигнал. В

рамках проведення лабораторних досліджень студентам запропоновано застосовувати гудки різноманітних морських суден. Вигляд стенду під час лабораторного дослідження наведено на рис. 1.

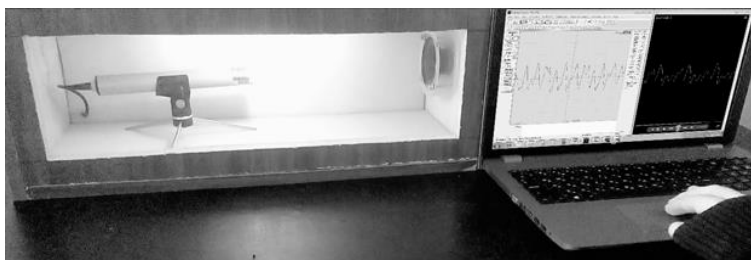


Рисунок 1 – Зовнішній вигляд лабораторного стенду

В рамках проведення дослідів студенти мають можливість осцилографувати перетворення звукового сигналу в електричний застосовуючи запропоноване програмне забезпечення *TrueRTA*. Керуючись вказівками викладача студенти мають можливість отримати графічні залежності електричної напруги від часу (рис. 2), рівня інтенсивності звуку від частоти (рис. 3). Обертаючи РГ на лабораторному стенді студенти можуть спостерігати зміну даних характеристик відповідно до заданого рівня звуку. Результати досліджень можна роздрукувати або зберегти у внутрішній пам'яті НБ для подальшого більш детального аналізу.

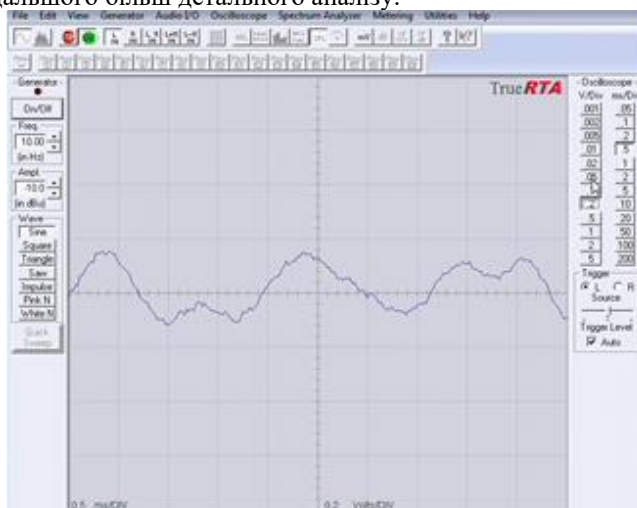


Рисунок 2 – Графічна залежність електричної напруги від часу

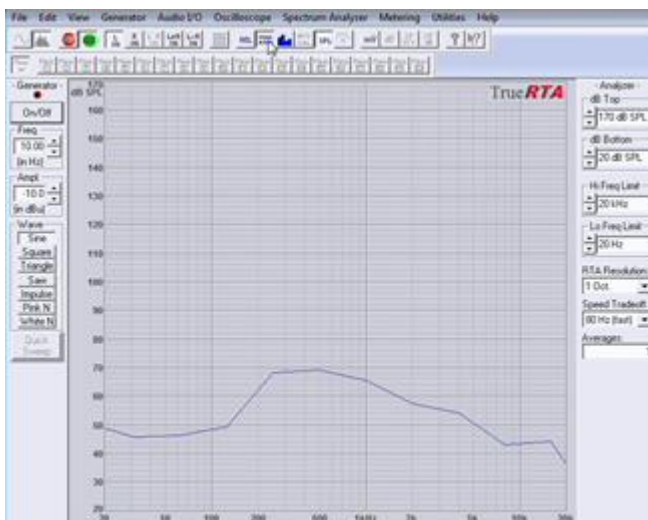


Рисунок 3 – Графічна залежність рівня інтенсивності звуку від частоти

Загальні технічні характеристики лабораторного стенду представлені у вигляді табл. 1.

Таблиця 1 – Технічні характеристики лабораторного стенда

Найменування параметра	Значення
Номінальна змінна напруга живлення, В	220
Частота мережі, Гц	50
Конектори живлення	<i>ALCGB, AC-166</i>
Сигнальна лампа	<i>AD22-22DS</i>
Дослідний конденсаторний електретний МК	МКЕ-271
Дослідний ДН	<i>AIYIMA, №4/20</i>
Потужність на виході ДН, Вт	20
Регулятор ППЗЧ	<i>Roxton, T6</i>
Комплект комутаційних кабелів	<i>3,5mini jack</i>
Джерело живлення ППЗЧ	<i>AC/DC 220/18-2</i>
Опір підключення ДН, Ом	4
Діапазон підсилюючих частот, Гц	30...80000
Світловий модуль	<i>COB 2W</i>
Лабораторний НБ	<i>HP, G6-2322SR</i>
Програмне забезпечення НБ	<i>TrueRTA</i>
Вихідна напруга для підключення НБ, В	18
Розетка для підключення НБ	<i>V:KO</i>

Діапазон робочих температур, градусів Цельсія	+10...+50
Габарити, мм	700x245x230
Маса, кг	7,2

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Стахів П.Г., Коруд В.І., Гамола О.Є. Основи електроніки: функціональні елементи та їх застосування: підручник. Львів: «Новий Світ-2000»; «Магнолія плюс», 2003. 208 с.

2. Победаш К.К., Святненко В.А. Силові напівпровідникові прилади і перетворювачі електричної енергії: навч. посіб. Київ: КПІ імені Ігоря Сікорського, 2017. 244 с.

3. Квітка С.О. Електроніка та мікросхемотехніка: підручник. Мелітополь: Видавничо-поліграфічний центр «Люкс», 2019. 223 с.

УДК 004

Головченко Д. С., Кубов В. І.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПЕРЕДАЧА ДАНИХ З СЕНСОРІВ ДИСТАНЦІЙНИХ ОБ'ЄКТІВ

При задачах збору та обробки даних виникає проблема як ці дані отримувати, особливо, з об'єктів де нема інтернет кабелю або точки Wi-Fi. Яку архтектуру вибрати? Які інтерфейси та протоколи?

Мета дослідження — розробити архтектуру, якою можна користуватися у реальних задачах автоматизації, з урахуванням ціни, безпеки та надійності.

На початку в нас є сенсор, що є джерело даних, в залежності від самого сенсору, в нього будуть різні інтерфейси підключення до контролеру. Мікроконтролер слугує перетворювачем даних та джерелом сигналів для GSM, що у свою чергу буде відправляти дані у JSON форматі на сервер за допомогою POST запита. Веб сервер слугує розподільником навантажень та додатковою ланкою безпеки. Мікросервіс REST API слугує приймачем даних на сервер, де з ними можна проводити маніпуляції та записати дані у базу даних, крім того через цей мікросервіс можна отримувати інформацію з бази даних у інтерфейсі. Інтерфейс слугує для користувачів як засіб відображення інформації та деяких налаштувань.

Для комплектації пропонуються наступні компоненти:

1. мікроконтролер – STM32;
2. GSM модуль – SIM900/800;
3. Веб сервер – Nginx;
4. Фреймворк (Framework) для “Інтерфейсу” – Vue.js;
5. REST API – FastAPI (Python);
6. База даних – Postgres.

Веб сервер Nginx – легкий, тому і швидкий, в порівнянні з іншими (наприклад, Apache), безкоштовний та легкий в налаштуванні. FastAPI – це сучасний мікрофреймворк для написання REST API, вже вбудований з автоматично генеруємою документацією Swagger та ReDoc. На рис. 1 зображена функціональна схема розробленої архітектури.

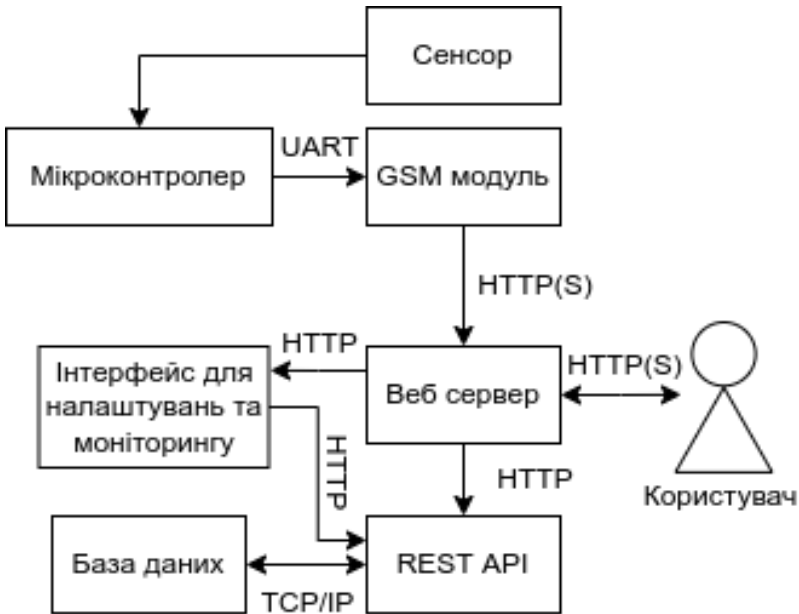


Рисунок 1 – Розроблена архітектура передачі даних

Якщо потрібно розташувати декілька датчиків, і вони більше ніж на 30 метрів один від одного, звісно, можна використати дроти, але якщо це не можливо зробити, це можна організувати за допомогою одного контролера (Raspberry pi, наприклад), що буде отримувати дані з всіх датчиків та відправляти на сервер. В такому випадку мікроконтролери можуть спілкуватися за допомогою nRF24L01,

мікроконтролери при цьому можуть бути найпростішими, наприклад, Attiny13, це буде залежати від сенсорів та задач, що поставлені у конкретній точці об'єкта.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media, 2017. 624 p.

УДК 65.011.56

*Мальцев Є. Є., Кудінов Д. В., Сідєлєв М. І.
Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

SCADA-СИСТЕМИ В СИСТЕМАХ АВТОМАТИЗАЦІЇ ВОДЯНИХ НАСОСНИХ СТАНЦІЙ

Потужні водяні насоси використовуються, наприклад, в системах водопостачання питної води для міст та великих підприємств. Особливістю їх використання є віддалена експлуатація, що потребує постійного контролю та втручання в логіку керування системою. На сучасному етапі розвитку виробничих підприємств активно розвивають комп'ютерно-інтегровані технології на основі SCADA-систем (Supervisory Control And Data Acquisition — диспетчерське управління і збір даних), що передбачають використання деякої кількості датчиків, системи віддаленої передачі даних та моніторингу і керування процесом виробництва.

Промислові насосні станції для води. Призначення промислових насосних станцій - перекачування рідин з одного місця в інше. Ці системи комплектуються: насосами для перекачування води; трубопроводами з арматурою; автоматикою для оптимізації роботи і захисту обладнання; приладами для вимірювання і контролю параметрів.

За призначенням промислові насосні станції поділяються на 4 види:

1. Першого підйому - забору і перекачування води в накопичувальні резервуари або на водоочисні споруди.

2. Другого підйому - транспортування води до кінцевих споживачів.

3. Циркуляційні - для забезпечення потрібного показника тиску в замкнутих системах.

4. Підвищувальні - для підвищення напору води на певній ділянці мережі.

Проектування автоматизованої системи керування віддаленими насосами передбачає виконання наступних завдань:

1. Автоматизація виробництва. На основі заданих параметрів система здійснює запуск, контроль, зупинку обладнання, аналізує і миттєво реагує на зміну умов.

2. Реєстрація, збір, фіксація інформації про роботу обладнання і виробничих процесів. Спеціальні датчики збирають, зберігають, передають відповідні дані на робоче місце оператора в режимі реального часу, а також фіксують в базах даних.

3. Розпізнавання і реакція на нестабільну роботу обладнання і його вузлів. Впровадження автоматизованих систем дозволяє реєструвати будь-які відхилення, збої, відмови в роботі пристроїв і техніки після збору потрібних даних. Реакція на подібні зміни може бути автоматичною або передаватися оператору для прийняття рішення.

4. Автоматизація технологічних процесів і виробництв в віддаленому режимі. Це можливо вручну або з робочого місця оператора.

5. Організація гнучкого доступу до управління. Опція має на увазі настроювання багаторівневого доступу до системи через персоналізовані паролі.

6. Автоматизована система контролю і обліку електроенергії та зберігання даних онлайн і оффлайн.

У зв'язку з тим, що довгий час в Україні управління технологічними процесами ґрунтувалося на досвіді персоналу підприємства, переклад на управління через SCADA-системи став здійснюватися набагато пізніше. Однак в даний час в Україні налічується безліч компаній, що займаються розробкою і впровадженням SCADA-систем. Ось тільки деякі популярні продукти даного ринку: Trace Mode, InTouch, Genesis, PROFICY iFIX, MasterSCADA, Simatic WinCC.

Будь-який з цих продуктів надає механізми збору та аналізу даних, управління "Alert", а також забезпечує візуалізацію процесу і диспетчерське управління технологічними операціями. SCADA-пакети дозволяють розробляти складні застосунки і дають можливість стежити за всіма етапами процесу управління, за станом обладнання і ресурсами, а також забезпечує високий рівень надійності. Функції SCADA-пакетів не обмежуються базовими операціями з даними, такими як збір даних з приладів обліку, контроль над заданими значеннями, команди управління. Системи реалізують перевірки відповідності параметрів

заданим значенням, контролюють достовірність отриманих даних, проводять складні розрахунки, що дозволяють спростити роботу з управління процесами. Так системи дозволяють підтримувати безперебійне транспортування води до споживача через автоматичну підтримку, з високою точністю задаються технологічних параметрів: значення тиску в мережі в реальному часі, аналіз якості та витрати води, рівень в резервуарах. Отримані дані зберігаються в архіві. Періодичність архівування варіюється в широких межах. Дані SCADA-пакети дозволяють досягти реальних результатів: поліпшити ефективність роботи, звітність, працездатність і ремонтпридатність обладнання.

Таким чином, після реалізації системи автоматизації керування водяними насосами отримуємо наступні переваги: усі ланки виробництва повністю автоматизовані, автоматизований аналіз компонентів водопостачання, оператор спостерігає за усім процесом через монітор комп'ютера, при необхідності може втрутитись у технологічний процес, звукові сигнали супроводжують виробничі події, що значно полегшують роботу операторів та підвищують швидкість реакції, а тим самим зменшують час простою обладнання.

УДК 62.52

Нерега М. В., Беліков О. Є.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АВТОМАТИЧНИЙ ДЕЗІНФЕКТОР ДЛЯ РУК

Разом із поширенням інфекційних хвороб виникає потреба у регулярній дезінфекції рук, так як одним із найбільш поширених способів передачі інфекції є потрапляння клітин вірусу до слизових оболонок людини з поверхні рук при торканні обличчя. Повсюдне носіння рукавичок не може вирішити цю проблему, так як неможлива достатньо часта їх зміна. Портативні дезінфектори для рук не завжди є в наявності при потребі дезінфекції.

Стаціонарні дезінфектори для рук та пульверизатори для дезінфікуючих розчинів потребують безпосереднього контакту із органами управління ними, що при їх використанні великою кількістю людей може мати зворотний ефект по відношенню до запобігання поширенню інфекцій. Тому найбільш оптимальним рішенням задачі дезінфекції рук та/або інших поверхонь і предметів є безконтактні засоби [1].

На рис.1 показано спрощену функціональну схему автоматичного дезінфектора для рук.

У мінімальній конфігурації система складається з мікроконтролера, датчика присутності рук, помпи та одного розпилювача. У розширеній інтерактивній конфігурації наявні: однопоточний комп'ютер, датчик температури, датчик присутності рук, датчик присутності людини, а також підсилювач з динаміком для відтворення голосових команд, дисплеї, вебкамера, помпа, 3 форсунки для розпилювання, та модуль передачі даних для контролю та моніторингу.

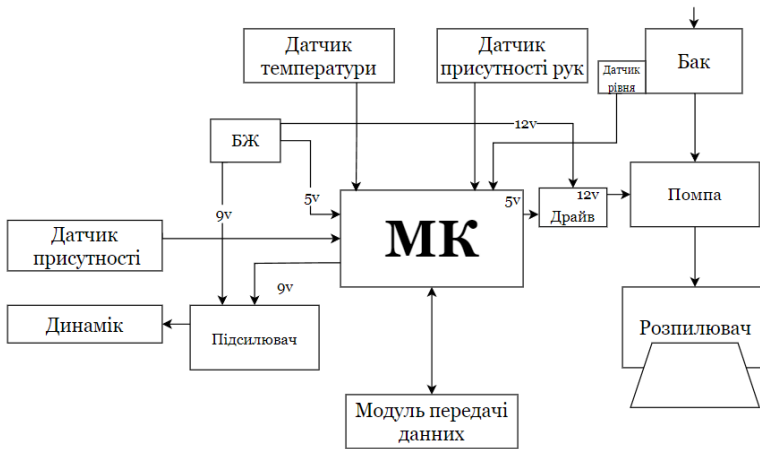


Рисунок 1 – Функціональна схема автоматичного дезінфектора для рук.

Розпилювач працює за принципом мікро-розпилення, що дозволяє використовувати мінімальну кількість антисептика та заощадити дезінфікуючий засіб. Розпилення рідкого антисептика на руки здійснюється, не торкаючись пристрою, що забезпечує чистоту рук. Апарат не є спеціальною медичною технікою, але може знайти широке застосування у лікувально-профілактичних установах України. Завдяки здатності пристрою відтворювати голосові команди, ми залучаємо більше людей і запобігаємо поширенню Covid-19. Також наявність безконтактного датчику температури та модулю передачі даних дає можливість створити певні пропускні пункти. При підвищеній температурі пристрій буде сигналізувати це звуковою командою та при необхідності відправляти данні на певний сервер для моніторингу та контролю людей з високою температурою під час епідемії.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Корисна модель засобу дезінфекції рук. Ukrpatent Special Information System: вебсайт. URL: https://sis.ukrpatent.org/media/UTILITY_MOD/2020/u202003506/99_144991-%D0%9A_D_UA.pdf

УДК 004

*Паламарчук А. О., Димитров Ю. Ю., Кубов В. І.
Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ДИСТАНЦІЙНЕ УПРАВЛІННЯ ОБЛАДНАННЯМ СИСТЕМ АВТОМАТИЗАЦІЇ ОБ'ЄКТІВ ВОДНИХ РЕСУРСІВ

Станом на листопад 2021 року Регіональному Офісу Водних Ресурсів Миколаївської області підпорядковано понад тисячі об'єктів водокористування та близько 20 насосних станцій.

Ці об'єкти розташовані на великій відстані один від одного та від Регіонального центру.

Велика відстань між окремими об'єктами, та віддаленість від населених пунктів примушує розглядати бездротовий зв'язок – а саме мобільний GSM зв'язок, як найбільш перспективний.

Задачу дистанційного контролю можна без значних витрат реалізувати на базі комплексу мікроконтролерної плати та GSM-модема. Мікроконтролерна плата типу Arduino, на базі процесора ATmega328 дає можливість під'єднати до 10 цифрових, та 6 аналогових сенсорів, або під'єднати 16 цифрових виконавчих пристроїв - актуаторів, що цілком достатньо для більшості задач малої автоматизації. Найбільш популярний GSM-модем на базі Sim900 має достатньо простий послідовний інтерфейс з використанням текстових AT-команд [1].

На рис.1 показано узагальнену функціональну схему системи керування на базі GSM-модему.

У мінімальній конфігурації система складається з мікроконтролерного блоку керування обладнаного GSM-модемом, що приймає та передає інформацію до мережі оператора мобільного зв'язку у вигляді SMS-повідомлень. Користувач може контролювати та керувати станом обладнання за допомогою мобільного телефону.

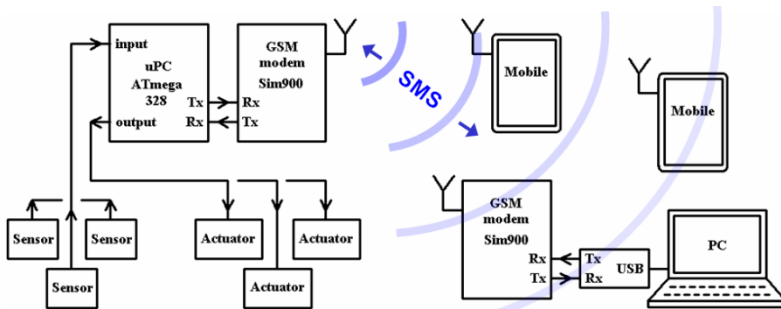


Рисунок 1 – Функціональна схема системи керування на базі GSM-модему

У розширеній конфігурації до системи може додатково під'єднуватися комп'ютер з GSM-модемом. Для сполучення GSM-модему Sim900 с комп'ютером необхідно використовувати перетворювач послідовного UART-інтерфейсу модема до USB-інтерфейсу комп'ютера. Для підтримки обміну SMS-повідомленнями на комп'ютері, в найпростішому випадку, можна використовувати будь яку комунікаційну програму, що емулює термінал.

Для організації безпроводного зв'язку кожного об'єкту з центральним диспетчерським пультом пропонується використовувати комплект радіотерміналів системи стільникового зв'язку GSM-900/1800, типу ОВЕН ПМ210 (Сертифікат експертизи типу №2694-СЕТ). Радіотермінал ПМ-210 встановлюється по 1 шт. на кожен об'єкт спостереження та керування. Кожен радіотермінал отримуватиме дані по промислового інтерфейсу RS-485 (протоколи Modbus або ОВЕН) від промислових контролерів, і передаватиме дані по GSM мережі у спеціалізоване хмарне сховище сервісу OwenCloud. Дані про стани об'єктів диспетчеризації будуть зберігатись на хмарному сховищі. Дані з хмарного сховища передаватимуться на диспетчерське автоматизоване робоче місце. Крім того за наявності паролю можна переглядати дані і графіки безпосередньо у хмарному сховищі з браузера будьякого пристрою, за наявності доступу до інтернет.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. SIM900 AT-Commands Manual V1.11. URL: https://simcom.ee/documents/SIM900/SIM900_AT%20Command%20Manual_V1.11.pdf

СИСТЕМА ELECTRONIC STABILITY PROGRAM (ESP) ДЛЯ ВІТЧИЗНЯНИХ АВТОМОБІЛІВ

Electronic Stability Program або скорочено ESP — це найпопулярніша серед численних автомобільних аббревіатур, які означають одну річ — динамічну систему стабілізації руху авто. Залежно від виробника, називається вона може по-різному: VDC, ESC, DSC, VSC і т. д., але суті це не міняє: система стабілізації допомагає водієві впоратися з автомобілем у різних ситуаціях.

У разі виникнення будь-якої аварійної ситуації, повернути автомобіль у вихідне положення і направити його на колишній курс система може завдяки вибірково пригальмовуванню одного або кількох коліс. Залежно від ситуації, система визначає яке з коліс потрібно сповільнити — зовнішнє чи внутрішнє, переднє чи заднє.

ESP контролює поперечну динаміку автомобіля, допомагаючи водієві в критичних ситуаціях, тим самим запобігаючи зрив автомобіля в занос або в бічне ковзання. По суті, система стабілізації зберігає курсову стійкість, траєкторію руху і стабілізує автомобіль під час виконання маневрів. А особливо на високій швидкості або на поганому покритті, коли схильність до зносу або заносу набагато вища. Звідси впливає й інша народна назва — противозаносна система[1].

На рис.1 показано узагальнену функціональну схему системи Electronic Stability Program (ESP).

Основні елементи системи являються 4 датчика обертів, які зчитують кількість обертів кожного колеса автомобіля окремо і передають інформацію на центральний процесор ESP, датчик кута повороту руля, який зчитує кут повороту руля автомобіля і передає інформацію на центральний процесор системи ESP, акселерометр який вимірює прискорення автомобіля і передає інформацію на центральний процесор системи ESP і центральний процесор ESP який передає отримані дані на центральний процесор автомобіля.

Система ESP автомобіля тісно зв'язана з системою АБС (антиблокувальна система) що запобігає блокуванню коліс при гальмуванні.

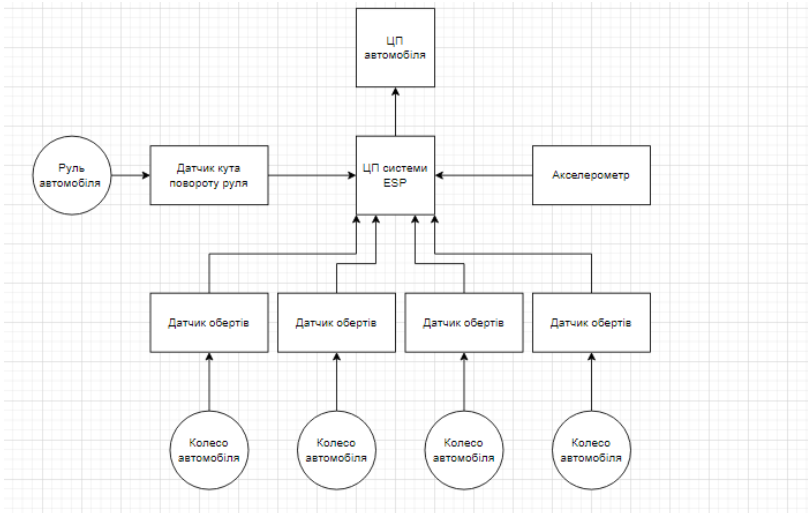


Рисунок 1 - функціональна схема Electronic Stability Program (ESP)

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Що таке система ESP і навіщо вона потрібна? Auto-ria: вебсайт. URL: <https://auto.ria.com/uk/news/autoservice-technology/221280/chto-takoe-sistema-esp-i-zachem-ona-nuzhna.html>

УДК 615.451.13

Терещенко Г. Ю., Бліков О. Є.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АВТОМАТИЗОВАНА СИСТЕМА ВИДАЧІ ЛІКАРСЬКИХ ЗАСОБІВ ЗА ПРИПИСОМ ЛІКАРЯ З ФУНКЦІЄЮ ВІДДАЛЕНОГО СПОВІЩЕННЯ

Сучасні хвороби не тільки заважають нормальному життю людей, а й смертельно загрожують йому. Для своєчасного лікування та зменшення ризиків необхідно звертатися до лікарів та слідувати інструкціям щодо прийому лікарських засобів. Незалежно від того, скільки вам років, ви можете забувати про те, що вам потрібно приймати ліки. Значною мірою від цього страждають пенсіонери. Саме вони частіше звертаються по допомогу до лікарів.

За даними ВОЗ, у період з 2015 по 2050 рік частка людей похилого віку у світі подвоїться, приблизно з 12% до 22%. Очікується, що кількість людей віком від 60 років з 900 мільйонів збільшиться до 2 мільярдів. В цілому, на порушення пам'яті скаржаться від 50 до 75% людей похилого віку. За оцінками, у світі 50 мільйонів людей живе з деменцією, причому 60% хворих на деменцію проживає в країнах з низьким і середнім рівнем доходу.

Професор медицини Ендрю Бадсон опублікував статтю про порушення когнітивної функції мозку внаслідок COVID-19. У цій публікації він посилається на китайське дослідження за участю тридцяти людей, які одужали після коронавірусу. Вони всі страждали через проблеми з концентрацією та сплутаною свідомістю: їм було важко запам'ятати схему прийому ліків, рахувати, вчитуватися в тексти і навіть свідомо розмовляти [1].

Одним із способів запобігти проблемам з прийомом ліків є дозатори пігулок [2]. Сьогодні на ринку існує безліч моделей та варіантів.



а



б



в

Рисунок 1 – Приклади існуючих дозаторів пігулок

Популярним дозатором є e-Pill MedSmart Plus (Рисунок 1а). Він має блокування ключем, що запобігає завчасному прийому пігулок. Програмою передбачено шість щоденних будильників, які повідомляють пацієнту про ліки. Недоліком такого дозатору є відсутність додаткових сигналів сповіщення на випадок, якщо пацієнт страждає на вади слуху.

На значно іншому рівні знаходиться дозатор для пігулок Hero Automatic Medication Dispenser (Рисунок 1б). Прилад пропонує звукові та візуальні нагадування про час прийому ліків, а також мобільний застосунок. Додатково встановлюється PIN -код, а ємність для пігулок розрахована на 90 днів. Недоліком такого апарату є його висока вартість та одностороння направленість.

Дозатор MedaCube (Рисунок 1в) розрізняє до 16 видів пігулок, дає можливість слідкувати за графіком та перевіряти пропущені сеанси прийому. Однак він достатньо важкий, займає багато місця та дорого коштує, що знижує конкурентоспроможність на вітчизняному ринку.

Розробляючи власний прилад, слід урахувати недоліки та переваги лідерів ринку дозаторів. Корпус має бути виготовлений з гіпоалергенних матеріалів. Для зручності та економії простору, пігулки будуть зберігатися у барабанах з секціями, які розташовані один над одним. Спеціальні датчики слідкують за наявністю пігулок, відповідають за їх сортування та температуру. Опитуючи їх, мікроконтролер, слідкує за мікрокліматом всередині апарату, надсилає повідомлення про прийом ліків, їх наявність або пропущенні сеанси. Передбачається розробка мобільного застосунка та системи сповіщення на телефон за рахунок GSM-модуля.

Актуальність теми зумовлена не тільки сьогоденною епідеміологічною ситуацією, а й тенденцією на старіння націй планети. Тож, розробка таких приладів, як дозатори ліків, є важливою місією на плечах інженерів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Landscape of cognitive function in recovered COVID-19 patients. Science Direct: вебсайт. URL: <https://www.sciencedirect.com/science/article/pii/S0022395620308542?via%3Dihub>.
2. Best Pill Organizers and Dispensers to Keep Your Week on Track. Healthline: вебсайт. URL: <https://www.healthline.com/health/pill-dispenser>.

УДК 681.518.5

Тіхоміров К. А., Прищепов О. Ф.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АВТОМАТИЗОВАНА СИСТЕМА КЕРУВАННЯ БЕЗПІЛОТНИМ ЛІТАЮЧИМ АПАРАТОМ

В даний час завдання створення безпілотних літаючих апаратів (БПЛА) різного призначення і конфігурації є вкрай актуальною і перспективною темою. Дана тематика дуже приваблива для молодих конструкторських колективів з огляду на її затребуваності та інноваційність. Цивільні БПЛА застосовують для аерофотозйомки, патрулювання районів, інспекцій, доставки вантажів, забезпечення

зв'язком заданих областей, контролю повітряного простору і т. п.

Системи автоматичного керування БПЛА мають ідентичну морфологію, але в залежності від призначення літаючих апаратів (ЛА) архітектура їх системи керування (органи управління, склад бортових систем, система зв'язку, датчикова апаратура і т. п.) може бути сформована по-різному відповідно до конкретних завдань.

У зв'язку з цим система керування безпілотним літаючим апаратом повинна реалізовувати наступні функції:

- автоматичний зліт та посадка;
- автономний політ по заданій траєкторії, тобто програмне керування;
- стабілізація положення в просторі при зовнішньому збуренні (сильні повітряні потоки, зміщення центру мас БПЛА, тощо);
- високу маневреність, швидкодію та точність руху по траєкторіям;
- планування маршруту з уникненням статичних та динамічних перешкод.

Неповна комплектація і відповідність встановленим вимогам сучасних безпілотних літаючих апаратів не надають максимальної ефективності та успішності при виконанні задач позиціонування на місцевості та стабільності польоту. Тому дослідження в напрямку покращення керування безпілотними літаючими апаратами націленими на підвищення точності польоту є актуальними і мають важливе практичне значення [2].

Наше завдання забезпечити ефективне керування безпілотним літаючим апаратом у просторі, що дозволить з більшою точністю переміщатися визначеним маршрутом та підвищить стабільність польоту.

При вирішенні поставленої задачі використовувалися методи математичного аналізу та моделювання, синтезу модальних регуляторів та ПД-регуляторів для складних цифрових систем. Для математичних розрахунків застосовувався пакет прикладних програм MATLAB, моделювання систем виконувалося в графічному середовищі MATLAB/Simulink [1].

До числа наукових результатів, отриманих в роботі, відноситься: аналіз властивостей БПЛА; визначено характеристики та параметри, що впливають на точність та позиціонування БПЛА; висвітлено коло питань, які необхідно враховувати при побудові системи керування безпілотним літаючим апаратом; удосконалена формула розрахунку коефіцієнтів ПД-регулятора для складних цифрових систем.

Практичне значення одержаних результатів полягає в

наступному:

- запропонована система керування безпілотним літаючим апаратом дозволяє підвищити ефективність керування положенням БПЛА та його стабілізацію в польоті;

- запропоновано удосконалена формула розрахунку коефіцієнтів ПІД-регулятора для складних цифрових систем.

В процесі роботи був проведений огляд існуючих різновидів безпілотників; спроектована модель руху в просторі безпілотного літаючого апарату; наведена блок-схема системи управління квадрокоптера і розглянуті моменти і сили, що враховуються при розробці ММ системи управління польотом. Також провели огляд складу елементів безпілотника на прикладі квадрокоптера, були наведені загальні принципи обробки інформації з датчиків квадрокоптера.

В результаті була спроектована система управління безпілотним літаючим апаратом, яка забезпечує високу ефективність та керованість. На сьогоднішній день застосування безпілотного літаючого апарату є перспективним напрямком [3].

Були розглянуті різні методи, які можна застосувати до аналізу динаміки польоту квадрокоптера. Під час розроблення системи було виведено математичну модель БПЛА, яка є комплексом із трьох підсистем диференціальних рівнянь для керування положенням дрону, а саме швидкістю, висотою та кутами крену, тангажу та ристання. На основі знайдених систем було побудовано відповідні моделі в графічному середовищі MATLAB/Simulink. Подальше дослідження та аналіз моделей показав, що системи керування кутами є нестійкими, а система керування висотою має погані показники якості (перерегулювання становило 76,9%, час встановлення – 2 с, а час регулювання – 52,8 с). Для стабілізації та покращення якісних характеристик системи керування БПЛА було розроблено два види регуляторів. Синтез системи керування кутами було виконано з допомогою модального регулятора, завдяки якому вдалося перевести систему в стійкий стан та забезпечити високу швидкодію (час регулювання – 1,2 с, час встановлення – 0,612 с) та нульове перерегулювання, проте з'явилася стала похибка (0,56 – для системи керування кутом ристання, 0,263 – для системи керування кутом крену/тангажу). Для її усунення було застосовано модифікований модальний регулятор, який до того ж забезпечив кращі показники якості (перерегулювання – 0%, час встановлення – 0,597 с, а час регулювання – 1 с). Для системи керування висотою було розроблено цифровий ПІД-регулятор на основі удосконаленої формули розрахунку коефіцієнтів ПІД-регуляторів для складних систем. Даний регулятор

складається з одного ПД-ланцюгу, відповідно має просту структуру, та покращує якісні характеристики системи керування висотою (перерегулювання – 0%, час встановлення – 1,42 с, а час регулювання – 2,5 с), зводячи при цьому значення сталої похибки до нуля.

Проведені дослідження підтверджують придатність зазначених підходів до дослідження динаміки польоту квадрокоптера з перспективою їх вдосконалення.

Представлено вимоги безпеки нормативно-правових документів щодо роботи з друком на 3D DLP принтері, слюсарним обладнанням, електрикою.

Розрахована витяжна система вентиляції та підібрано технічне обладнання. Розрахунковими параметрами системи вентиляції та аеродинамічний опір $\Delta p=80\text{т}$ водяного стовпчика, в якості вентилятора підібрано відцентровий вентилятор Maico ECA 100 ірго Н з серії ECA ірго.

Можна підвести підсумок, що безпілотні літальні апарати є невід’ємною частиною робототехніки та автоматизації. Як факт, на продуктивність і на якість роботи БПЛА впливає його система автоматизації, а саме алгоритми управління і точність контрольно-вимірювальних приладів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Молчанов А. О. Методи інформаційної технології забезпечення безпеки руху безпілотних систем оптичними засобами : автореф. дис. на здоб. наук. ступеня канд. техн. наук : [спец.] 05.13.06 "Інформаційні технології". МОН України, Нац. аерокосмічний ун-т імені М. Є. Жуковського "Харківський авіаційний ін-т". Х., 2018. 20 с.

2. Ростопчин В. В., Бурдун І. Е. Безпілотні авіаційні системи: основні поняття. ЕЛЕКТРОНІКА: Наука, Технологія, Бізнес. 2016. №7. С. 82-88.

3. Сухов В. В., Козей Я. С. Особенности выбора траектории и этапов полета беспилотного летательного аппарата на солнечной энергии в условиях неспокойной атмосферы. Proceedings of the National Academy of Sciences of Belarus. Physical-technical series, 2019, vol. 63, no. 4, pp. 486–500.

Методи і засоби комп'ютерної інженерії

УДК 004.8

Важоха В. Б., Крайник Я. М.

*Чорноморський національний університет імені Петра Могили,
м.Миколаїв, Україна*

КОМПЛЕКСНА СИСТЕМА УПРАВЛІННЯ СЕРВІСАМИ З ВИКОРИСТАННЯМ ЗАСОБІВ КОНТЕЙНЕРИЗАЦІЇ

З метою досягнення поставленої задачі, а саме розробки комплексної системи управління сервісами з використанням засобів контейнеризації було проведено детальний аналіз предметної області і визначення поняття «сервіс» як динамічна система дій, спрямованих на отримання заданих результатів у багатокритеріальному полі протягом встановленого строку та в рамках виділених ресурсів із залученням виконавців, які володіють необхідними навичками та знаннями.

Проведено дослідження моделей управління сервісами та огляд існуючих аналогів. Проведений аналіз дає можливість зробити висновок про те, більшість існуючих на ринку систем використовують застарілі методи управління системою, а програмний продукт часто є не повністю функціональним та мають незручне керування.

На основі проведеного аналізу було здійснено вибір технічних та програмних засобів для реалізації системи. Систему було вирішено спроектувати на базі мікросервісної архітектури.

У рамках проведеного опису концепції контейнеризації та дослідження базових принципів та відмінностей технологій віртуалізації та контейнеризації саме систему «Docker» було обрано для забезпечення відповідного програмного середовища, а також проаналізовано основні підходи до налаштування процесів постійної інтеграції. Також було наведено основні питання які стосуються проблем безпеки міжсервісного спілкування в системі, а також підходів до масштабування застосунку.

Аналіз архітектури віртуальної платформи «Docker» та дослідження процесів роботи з образами також мали місце в процесі виконання дослідження та надали можливість сформулювати програмні вимоги до системи, серед яких є використання наступних технічних і програмних засобів: «Docker», «PHP», «Laravel», «PhpStorm», «MySQL», «VirtualBox», «Prometheus», «Grafana».

При розробці програмної частини системи було досліджено контейнеризацію системи, стандартизацію формату повідомлень, сам

процес розробки програмного забезпечення детально описано.

УДК 004.8

Горбуров Л. М., Крайник Я. М.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ОСОБЛИВОСТІ ПРОЄКТУВАННЯ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ ДЛЯ ДОСЛІДЖЕННЯ НЕЙРОННИХ МЕРЕЖ

Штучні нейронні мережі міцно увійшли в наше життя і в даний час широко використовуються при вирішенні різних завдань і активно застосовуються там, де звичайні алгоритмічні рішення виявляються неефективними або зовсім неможливими. Серед завдань, вирішення яких довіряють штучним нейронним мережам, можна назвати наступні: розпізнавання текстів, гра на біржі, контекстна реклама в Інтернеті, перевірка проведення підозрілих операцій з банківських карток, фільтрація спаму, системи безпеки та відеоспостереження – і це далеко не все.

У найпростішому вигляді машинне навчання використовує запрограмовані алгоритми, які отримують і аналізують вхідні дані, прогнозують вихідні значення з допустимого діапазону. По мірі надходження нових даних ці алгоритми навчаються та оптимізують свою діяльність, підвищуючи продуктивність та згодом розвиваючи «інтелект». Існують 4 типи алгоритмів машинного навчання: навчання з учителем, навчання з частковим залученням вчителя, навчання без вчителя та навчання з підкріпленням.

Нещодавні досягнення у сфері штучного інтелекту сприяли використанню низки алгоритмів. Для таких застосунків, як розпізнавання осіб, розпізнавання мовлення, миттєвий переклад та багато іншого.

Саме тому під час конференції з GPU-технологій, щорічний захід, що проводиться Nvidia. Був представлений Jetson Nano (див. рис. 1) який є одноплатним комп'ютером у вигляді модуля розміром 7 x45 мм, який коштуватиме близько 99 доларів США.

Цей новий комп'ютер з однією картою Jetson Nano. Він призначений для реалізації в роботах та інших пристроях на базі штучного інтелекту.

Новини про доступність Jetson Nano в майбутньому пішли за новинами про Jetson Xavier, систему на кристалі, призначеній для штучного інтелекту.



Рисунок 1 – Jetson Nano

Jetson Nano Developer Kit – це невеликий та потужний комп'ютер, який дозволяє паралельно запускати кілька нейронних мереж. Для таких програм, як класифікація зображень, виявлення об'єктів, сегментація та обробка мови. Все це на простій у використанні платформі, споживання якої становить близько 5 Вт.

Доповнивши цей комп'ютер камерою, можна досліджувати алгоритми машинного навчання: не лише для розпізнавання тексту або зображення яке знаходиться на комп'ютері, в мережі, а також отримувати зображення з відео потоку камери. Як приклад, розпізнавання обличчя в режимі реального часу.

Суть технології розпізнавання обличчя полягає у використанні серверів з програмним забезпеченням або спеціальних камер з вбудованою функцією розпізнавання обличчя для виявлення вузлових точок на обличчі та вимірювання відстаней між ними. Що дозволяє скласти карту особи (рис. 2.) та отримати так званий відбиток особи. Залежно від технології системі необхідно визначити близько 80 таких точок.



Рисунок 2 – Моделювання карти особи обличчя

Справжнім проривом стало усвідомлення того, що відбиток особи є унікальним кодом, який має кожна людина і який може бути вирахований на відстані. Швидкість розпізнавання та порівняння з існуючою картотекою може становити менше однієї секунди, а ймовірність помилки при цьому мінімальна. Тому технологія розпізнавання осіб має широкі перспективи для застосування.

Нейронні мережі здатні до навчання, завдяки чому їх можна оптимізувати та максимально збільшувати функціональність. Дослідження НМ – це одна з найперспективніших областей в даний час, оскільки в майбутньому вони будуть застосовуватися практично повсюдно, у різних галузях науки і техніки.

УДК 004.94

Дяденко А. С., Бойко А. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

БАГАТОКОРИСТУВАЦЬКИЙ СЕРВЕРНИЙ ЗАСТОСУНОК НА БАЗИ РУШІЯ UNREAL ENGINE

В сучасному світі все більшою популярністю в індустрії розваг користуються відео ігри, що особливо гостро стало відчуватися в останні два роки із-за самоізоляції під час пандемії коронавірусної хвороби. Розробка сучасних відео ігор є окремим видом мистецтва, що робить гравця героєм фантастичного всесвіту, даючи змогу пережити дивовижні події та впливати на них. Це дарує користувачу емоційні переживання, досвід та, у певній мірі, розширює кругозір. Більше того, зростаюча поширеність цифрових технологій, таких як смартфони та планшетні комп'ютери, робить доступним для більшості населення розважальне програмне забезпечення у вигляді казуальних відеоігор (VG) або гейміфікованих програм. В цілому, ігри мають масу позитивних властивостей. Наявні в цій галузі дослідження підтверджують, що активні відеоігри покращують характеристики уваги та зорового сприйняття. Доведено, що віртуальні розваги виявляються корисними при засвоєнні нових знань та розумінні контексту, а також для емоційних та мотиваційних характеристик діяльності, що вкрай важливо для навчання.

Актуальність розробки саме багатокористувацьких застосунків спричинена, крім вище перерахованих аспектів, ще і наявністю соціальної складової – змога грати з іншими користувачами з усього світу, що прибирає мовний бар'єр і поєднує людей. Сьогодні найбільші мобільні ігри включають захоплюючі багатокористувацькі світи, в яких

одночасно можуть знаходитися тисячі користувачів. Для повноцінної роботи таких застосунків потрібне серверне забезпечення, на якому відбувається та підтримується з'єднання користувачів.

Багатокористувацький застосунок планується розроблювати у тривимірному вигляді на базі рушія Unreal Engine [1]. Unreal Engine 4 – це потужна програма для розробки та створення контенту. Саме графічні можливості, потужні інструменти та чудова розширюваність роблять даний рушій одним з популярних та доступних платформ. Ігрові розробники, архітектори, вчені, творці фільмів та візуальні художники використовують UE4 для створення приголомшливого інтерактивного середовища.

Крім рушія - платформи, на якій буде запущено проєкт, необхідне спеціалізоване програмне забезпечення. У якості програми для скульптингу моделей було обрано Zbrush та Blender, програми для текстурування – SubstancePainter, для запікання карт – Marmost Toolbag. Крім того, планується використання Adobe Photoshop, як для створення 2D концепту майбутнього проєкту, так і для редагування карти поверхонь.

Задля можливості мережевої взаємодії у Unreal Engine планується розроблення мультиплеєра – багатокористувацької сесії. У сесії для кількох гравців інформація про стан гри передається між кількома машинами через Інтернет-з'єднання. Unreal Engine має надійну мережеву структуру, яка підтримує деякі з найпопулярніших онлайн-ігор у світі, допомагаючи спростити цей процес.

Мультиплеєр UE4 заснований на моделі клієнт-сервер. Це означає, що буде задіяний єдиний сервер, який матиме повноваження щодо стану гри, тоді як підключені клієнти підтримуватимуть близьке наближення. Сервер є важливою частиною мультиплеєра UE4. Він приймає всі важливі рішення, містить усі повноваження, обробляє клієнтські з'єднання, подорожі до нових карт і керує загальним процесом ігрового процесу.

У Unreal Engine є всі інструменти, необхідні дизайнерам та програмістам. Ці інструменти значно полегшують роботу фахівців. Наприклад, підтримують різні формати текстур, неймовірно повно відтворюють фізичні властивості матеріалів, дають можливість змінювати об'єкти в режимі реального часу, дозволяють вибирати джерела світла, додавати ефекти тощо. Багату колекцію ассетів надалі можна використовувати для розробки інших проєктів, а відкритий вихідний код дозволяє вносити до рушія необхідні зміни. Він легко імпортується на будь-яку платформу, адаптуючи ігри під ПК, консолі і мобільні гаджети.

Характеристики мультиплеєрного застосунку, що створюються

:

– у Unreal Engine 4 існує вбудована система візуального скриптингу – Blueprints. Сам рушій розроблено на C++ , тому, при потребі, можна скористатися цією мовою, наприклад для з'єднання з базою даних чи для збереження процесу. Для кращого результату та стабільної роботи застосунку буде скомбіновано Blueprints з класами C++.

– Спостереження за результатами роботи рівнів відбувається у в'юпорті – основною областю художніх робіт, де користувач може побачити створення гри. Камера налаштовується відповідно до попередньо налаштованого рівня гри. Після натискання кнопки «ВІДТВОРЕННЯ», вікно перегляду саме активується, і можна зв'язуватися з вікном перегляду та бачити різні рівні гри.

– Робота з ресурсами можлива через звичайний Drag&Drop.

– Підтримка імпортування великої кількості форматів файлів.

– Цифровий світ згенеровано процедурним шляхом з допомогою WorldMachine та Megascans World Machine допомагає створювати реалістичний тривимірний ландшафт. Потужний і гнучкий World Machine поєднує процедурне створення ландшафту, моделювання природи та інтерактивне редагування, щоб швидко та легко створити реалістичний ландшафт. Можна створювати та експортувати поля висоти, текстури та сітки високої роздільної здатності для гри або програмного забезпечення для візуалізації. У свою чергу Megascans — бібліотека текстур високої роздільної здатності та 3D-об'єктів, створених за допомогою фотограмметрії.

– У UE4 є потужний мережевий інструментарій, вбудований в рушій на базовому рівні.

– Підтримує як одиночну гру так і спільну з іншими користувачами.

– Наявна можливість колективної розробки з Multi-User Editing. За допомогою багатокористувацького редагування можна з'єднати кілька екземплярів Unreal Editor разом в спільній сесії редагування, створюючи єдиний віртуальний світ разом з колегами по команді в режимі реального часу.

У результаті виконання даної роботи отримується багатокористувацький застосунок на базі рушія UNREAL ENGINE зі серверною складовою, що дасть змогу використовувати продукт одночасно декільком користувачам.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Офіційний сайт Unreal Engine. URL: <https://www.unrealengine.com> (дата звернення 16.01.2022).
2. Офіційний сайт Blender. URL: <https://www.blender.org> (дата звернення: 17.01.2022).

Методи і засоби програмної інженерії

УДК 004.42

Голуб Р. Р., Дворецький М. Л.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ВИКОРИСТАННЯ АРХІТЕКТУРНИХ-ПАТЕРНІВ MVVM ТА MVP ПРИ РОЗРОБЦІ МОБІЛЬНИХ ЗАСТОСУНКІВ МОВОЮ KOTLIN

Для аналізу ефективності вибору технології для розробки Android застосунків варто досліджувати наступні критерії:

Час розробки. Якщо взяти фахівців одного рівня в кожній з технологій і дати однакове технічне завдання, скільки часу буде потрібно, щоб розв'язати цю проблему за допомогою кожної з архітектур.

Наявність фахівців. Наскільки швидко можна знайти розробників, які здатні створити продукт на високому якісному рівні, а також фахівців, які зможуть його в подальшому супроводжувати, використовуючи кожен з архітектур.

Зручність розробки та налагодження. Наскільки розвинені інструменти розробки і налагодження в рамках даної архітектури.

Ремонтопридатність. Легко додавати нові функції у ViewModel, якщо розробник програми має певний досвід роботи з конкретною бібліотекою.

Показники коду. MVP створює більше коду та класів, ніж архітектура ViewModel.

Також необхідно виділити наступні відмінності:

1. Дані MVP одержуються та маніпулюються зі стану даних, який присутній на рівні моделі. Ці дані передаються на рівень презентатора. У MVP немає взаємодії зі станом перегляду. MVVM має модель, розташовану в самій бізнес-логіці, де дані зберігаються у сховищі. Запити отримуються від моделі View, і дані розміщуються відповідно.

2. Модель View в MVP має користувацький інтерфейс, активність і фрагменти даних, а також взаємодіє з доповідачем. Модель View в MVVM взагалі не має бізнес-логіки і має лише користувацький інтерфейс.

3. Дані з моделі подаються через Presenter, і він контролює поведінку в застосунку в архітектурі MVP. Він спрямовує шар View та керує взаємодією між моделлю та view. Дані також зберігаються в моделі. Рівень презентатора відсутній у MVVM, а спостережувані створюються в кожному компоненті інтерфейсу користувача.

4. View у MVP не важливий або тісно пов'язаний із самим собою, щоб користувач міг легко ігнорувати шар перегляду. Шар View важливий, оскільки він створює міст між View та ViewModel в MVVM, так що неможливо не використовувати шар перегляду в MVVM.

5. Шари View та Presenter у MVP можна використовувати багаторазово, тому цю архітектуру легко підтримувати. Крім того, його можна підтримувати за допомогою читабельних кодів, написаних мовою розмітки або іншою мовою кодування. У MVVM немає взаємодії між представленням і моделлю, і тому код керується одиницями. Це допомагає робити модульне тестування в MVVM. Архітектура MVVM є більш привабливою для великої кількості розробників, тому що Google активно пропагує саме цю архітектуру через її переваги у розробці довготривалих та великих застосунків (більше 20 екранів), проте важче підтримувати та робити зміни, якщо розробник не має великою кількості досвіду роботи із цією архітектурою.

MVP є більш привабливим для застосунків, кількість екранів яких є менше двадцяти та за умови невеликої кількості досвіду розробника.

Якщо проаналізувати кількість викладених проєктів у мережі Github з тегом MVVM та тегом MVP (рис. 1), то можна побачити, що саме перша ахітектура має більший попит серед розробників мобільних застосунків мовою Kotlin, яку Google офіційно визнав основною мовою для розробки застосунків під Android.

Languages		Languages	
Kotlin	18,120	Java	17,106
Java	9,476	Kotlin	4,516

Рисунок 1 – Кількість викладених у вільний доступ проєктів з архітектурами MVVM та MVP відповідно

Проаналізувавши архітектури MVP та MVVM неможливо дати остаточну відповідь, яка з архітектур є найкращою. Кожна з архітектур залежить від багатьох умов, від навичок розробника до вимог клієнта. MVVM має допомогти нам позбутися деяких недоліків, які несе в собі MVP. З іншого боку, з'являться нові недоліки щодо MVVM, і ми повинні бути готові їх виправити, оскільки переваги добре розробленої архітектури завжди перевершують недоліки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The Global Smartphone Market Grew 13.2% in the Q2 2021 Despite Supply Concerns and Vendor Shakeups. IDC: вебсайт. 2021. URL: <https://www.idc.com/getdoc.jsp?containerId=prUS48120021>.

2. Загальна частка ринку мобільних ОС у продажах кінцевим споживачам з 1 кварталу 2012 року до 3-го кварталу 2021 року. Statista: вебсайт. URL: <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/>.

УДК 004.412

*Гусєва-Божаткіна В. А.
Національний університет кораблебудування
імені адмірала Макарова,
м. Миколаїв, Україна*

ВИЗНАЧЕННЯ МЕТРИК ТА ДОВІРЧОГО ІНТЕРВАЛУ ДЛЯ ПОБУДОВИ РЕГРЕСІЙНОГО РІВНЯННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНИХ ЗАСТОСУНКІВ, ЩО РОЗРОБЛЯЮТЬСЯ МОВОЮ COVOL ДЛЯ ПЛАТФОРМИ MAINFRAME

В наш час існує досить поширена проблема: не існує єдиного вимірювання, набору метрик та показників для оцінювання тривалості розробки програмного забезпечення (ПЗ). Існує багато методів для оцінювання тривалості програми, деякі з них базуються на кількості строк коду, інші методи обчислюють тривалість з функціональних, технічних або інших аспектів. Але більшість з них стають непридатними до використання через відсутність даних, ресурсів або експертних навичок в цій галузі.

Більшість з аспектів оцінювання походять від методу аналізу функціональних точок (FPA). Інший підхід полягає в тому, щоб провести функціональне вимірювання, для вираження функціональності у кількості, що представляє саме поняття

«тривалості» ПЗ. Також існують певні методи визначення тривалості програмного забезпечення, що включають оцінювання на основі варіантів використання (Use Case Measurement). Найпоширенішою та найбільш вживаною методологією визначення тривалості програмного забезпечення є підрахунок кількості рядків, написаних у вихідному коді програми.

Тривалість ПЗ є одним з найвагоміших факторів в управлінні процесом розробки ПЗ. Доведено, що тривалість ПЗ корелює з витратами, зусиллями та ресурсами, необхідними на його розробку. Також інформацію, отриману у результаті оцінювання тривалості ПЗ, можна використати для прогнозування зусиль розробки ПЗ за такими моделями як СОСОМО, які використовуються для надійного прогнозування різних параметрів, пов'язаних з проектом, та оцінюванням витрат. В якості «тривалості» ПЗ мається на увазі кількість рядків вихідного коду [1].

Наприклад, модель СОСОМО (COConstructive COst Model) розраховує трудомісткість розробки як функцію від тривалості ПЗ і безлічі «чинників вартості», що включають суб'єктивні оцінки характеристик продукту, проекту, персоналу і апаратного забезпечення. Ця модель часто використовується для надійного прогнозування різних параметрів, пов'язаних з проектом, та оцінювання витрат на розробку та впровадження. Але її використання для програмних застосунків викликає певні труднощі. В першу чергу це пов'язано з тим, що головною позицією для такої моделі є так звана «тривалість програмного забезпечення», який на стадії планування проекту не може бути передбаченим. Саме тому, на протязі останніх десятиріч вченими продовжується розробка і удосконалення різних математичних моделей прогнозування тривалості розробки (в тому числі і регресійних). Саме регресійні моделі описують тривалість розробки як випадкову величину. Враховуючи той факт, що такий розподіл не є гаусівським, необхідно застосувати нелінійні регресійні моделі, а їх побудову вести на основі багатовимірних нормалізуючих перетворень [2].

Актуальність проблеми отримання ефективної системи оцінювання кількості строк коду в даний час є важливим завданням, що вимагає удосконалення існуючих методів. Адже саме ефективність оцінки тривалості розробки програм може стати відправною точкою для успіху або невдачі проекту на ранньому етапі розробки.

Метою роботи є підвищення достовірності оцінювання тривалості розробки програмного забезпечення, розробленого для платформи MainFrame за допомогою мови програмування Cobol. Для досягнення поставленої мети, в першу чергу, необхідно вирішити наступні завдання:

- визначити необхідні метрики з кожного проекту, реалізованого на основі мови програмування Cobol, що може бути використано для перевірки розроблюваної моделі;
- визначити довірчий інтервал та інтервал прогнозування для нормалізованих даних.

Наукова новизна кінцевих результатів роботи полягає у розробці математичної моделі для оцінювання тривалості розробки застосунків на базі мови програмування Cobol, за рахунок використання нормалізуючого перетворення десятковий логарифм, та застосування нелінійного рівняння множинної регресії, що дозволить підвищити достовірність оцінювання тривалості програмного забезпечення.

Достатньо ефективною та результативною можна вважати модель з показником детермінації $R^2 > 0,8$. Якщо $R^2 = 1$, тоді лінія регресії точно відповідає усім спостереженням та вимогам, а модель можна вважати адекватною та достовірною. Коефіцієнт детермінації визначається за формулою:

$$R^2 = \frac{\sum_{i=1}^n (Y_{\text{розн}} - Y_{\text{сеп}})^2}{\sum_{i=1}^n (Y_{\text{факт}} - Y_{\text{сеп}})^2} \quad (1)$$

Величина коефіцієнта детермінації виступає важливим критерієм оцінки якості лінійних і нелінійних моделей. Чим вагоміша частка пояснюваної варіації, тим менша роль інших факторів, а отже, модель регресії краще апроксимує вихідні дані і такою регресійною моделлю можна скористатися для прогнозу значень результативного показника [3]

Довірчий інтервал лінійного рівняння регресії побудуємо за наступною формулою (2):

$$Y_{(x_k)} \pm t_{\left(\frac{\alpha}{2}, n-2\right)} * \left[D * \sqrt{\frac{1}{n} + \frac{(x_k - \bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}} \right],$$

$$\text{де } D = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n - p - 1}} \quad (2)$$

D^2 – залишкова дисперсія, що характеризує невизначену частину варіації y ; x_k – значення фактора, для якого будується довірчий інтервал; $t_{(\alpha/2, n-2)}$ – квантіль t -розподілу Стюдента; α - рівень значущості; n - кількість значень випадкових величин у вибірці; p – кількість змінних.

Для побудови нелінійного рівняння регресії необхідно також використати: вибірку восьмивимірних негаусових даних з фактичною тривалістю ПЗ в тисячах рядків коду (Y); загальну кількість процедур (x_1); середню кількість атрибутів (x_2); середню кількість операторів (x_3); середню кількість функцій (x_4); середню кількість методів PIC (x_5); середню кількість методів USAGE (x_6); кількість відношень між процедурами (x_7).

У результаті даної роботи проведено визначення формул для знаходження довірчого інтервалу та визначено критерії, за якими буде оцінюватися удосконалене регресійне рівняння для оцінювання тривалості розробки програмних застосунків, що розробляються мовою COBOL для платформи mainframe.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Голованова М.А., Надин Е.В. Оценка трудоемкости работ на ранних стадиях создания программного обеспечения. Системы обработки информации. Харків, 2014. № 8 (124). С.151-156.
2. Boehm V. Software Cost Estimation with Cocomo II. New Jersey, Prentice-Hall. 2000. 544 p
3. Prykhodko S. B., Prykhodko N. V., Smykodub T. G., Spinov A. V. Constructing the non-linear regression equation to estimate the software size of open source PHP-based information systems. Проблеми інформаційних технологій, 2018. № 1 (023). С.118-125. ISSN 1998-7005

УДК 004.434

Кандиба І. О., Ткаченко М. П.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АНАЛІЗ СКЛАДОВИХ ТЕЛЕКОМУНІКАЦІЙНОЇ МЕРЕЖІ ЗАКЛАДУ ВИЩОЇ ОСВІТИ ЗАСОБАМИ RUTN0N

Телекомунікаційна мережа закладу вищої освіти (ЗВО) являє собою складну структуру, що може включати комп'ютери з різними операційними системи та з різними апаратними архітекторами. Наведений факт робить необхідним гнучке налаштування мережевих служб, на приклад служби active directory. Ця служба підтримує можливість роботу з груповими політиками, що в свою чергу є реалізацією контролю над комп'ютерами домену.

В службі active directory комп'ютери можуть бути згруповані по групах, які заведено називати контейнери. Кожен контейнер містить

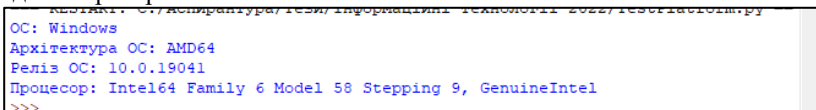
налаштування для визначеного набору комп'ютерів, що застосовуються при їх завантаженні. Недоліком системи є ймовірність включення до одного контейнеру комп'ютерів, що працюють на різних версіях операційних систем (ОС). Застосування однакових налаштувань до різних версій ОС часто призводить до несумісності та некоректної роботи комп'ютерів домену.

Описана вище проблема вирішується шляхом аналізу телекомунікаційної мережі. Цей аналіз може бути здійснений за допомогою використання вбудованих засобів операційних систем, наприклад інструменту *systeminfo*, але цей підхід вимагає витрат часу та участі фахівця. Спростити та автоматизувати процес збору поточної інформації про складові телекомунікаційної мережі можливо внаслідок створення прикладного програмного забезпечення, наприклад мовою Python. Мова програмування Python дозволяє підключення модулів *platform* та *psutil*.

Модуль *platform* дозволяє визначення архітектуру процесору, назву та версію операційної системи включаючи реліз, що є важливим чинником при роботі Windows 10. Ця операційна система має значні відмінності у релізах. Код для отримання інформації про ОС у Python виглядає наступним чином:

```
import platform
print("ОС:",platform.system())
print("Архітектура ОС:", platform.machine())
print("Реліз ОС:", platform.version())
print("Процесор:", platform.processor())
```

Функція `print()` виводить на екран отримані дані (рис. 1), але використовуючи модулі роботи з БД або протоколом SMB дані можуть бути надіслані на віддалений сервер для подальшої обробки адміністратором.



```
ОС: Windows
Архітектура ОС: AMD64
Реліз ОС: 10.0.19041
Процесор: Intel64 Family 6 Model 58 Stepping 9, GenuineIntel
>>>
```

Рисунок 1 – Виведення інформації бібліотеки *platform*

Інформація про встановлену операційну систему є корисною, але коректне налаштування мережевих служб вимагає визначення даних мережевого підключення. Реалізація цієї функції можлива з використання бібліотеки *psutil*. Ця бібліотека містить багато додаткових функцій визначення інформації про апаратне забезпечення, наприклад для виведення мережевих адаптерів реалізовано метод `psutil.net_if_addrs()`.

Результатом роботи `psutil.net_if_addrs()` є словник з іменами мережевих адаптерів у якості ключів та їх властивостями у якості значень. На одному комп'ютері може бути декілька мережевих адаптерів, що призводить до виводу значної кількості інформації (рис. 2).

```

Ethernet [nicaddr(family=<AddressFamily.AF_LINK: -1>, address='50-46-5D-56-02-09', netmask=None, broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET: 2>, address='169.254.59.172', netmask='255.255.0.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='fe80::e45a:f072:ice4b13bac', netmask=None, broadcast=None, ptp=None)]
----- Ethernet -----
vEthernet (Default Switch) [nicaddr(family=<AddressFamily.AF_LINK: -1>, address='00-15-5D-C1-6D-04', netmask=None, broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET: 2>, address='172.22.80.1', netmask='255.255.240.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='fe80::c08c:6a7:8f71:76c4', netmask=None, broadcast=None, ptp=None)]
----- vEthernet (Default Switch) -----
Подключение по локальной сети* 1 [nicaddr(family=<AddressFamily.AF_LINK: -1>, address='E-85-DE-31-84-D4', netmask=None, broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET: 2>, address='169.254.225.74', netmask='255.255.0.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='fe80::a56a:724d:6d55:3b7c', netmask=None, broadcast=None, ptp=None)]
----- Подключение по локальной сети* 1 -----
Подключение по локальной сети* 2 [nicaddr(family=<AddressFamily.AF_LINK: -1>, address='2E-85-DE-31-84-D4', netmask=None, broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET: 2>, address='169.254.225.74', netmask='255.255.0.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='fe80::c06e:bed3:d479:e58a', netmask=None, broadcast=None, ptp=None)]
----- Подключение по локальной сети* 2 -----
vEthernet (New) [nicaddr(family=<AddressFamily.AF_LINK: -1>, address='DC-85-DE-31-84-D4', netmask=None, broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET: 2>, address='192.168.88.252', netmask='255.255.255.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='fe80::69e8:c06e:ba18:c834', netmask=None, broadcast=None, ptp=None)]
----- vEthernet (New) -----
Loopback Pseudo-Interface 1 [nicaddr(family=<AddressFamily.AF_INET: 2>, address='127.0.0.1', netmask='255.0.0.0', broadcast=None, ptp=None), snicaddr(family=<AddressFamily.AF_INET6: 23>, address='::1', netmask=None, broadcast=None, ptp=None)]
----- Loopback Pseudo-Interface 1 -----

```

Рисунок 2 – Виведення інформації про мережеві адаптери

Active directory має широкий спектр можливостей, в тому числі віддалене встановлення програмного забезпечення. Контроль цього процесу вимагає наявності засобів моніторингу апаратного забезпечення: кількості оперативної пам'яті, частоти ЦП, кількості вільного простору.

Здійснити моніторинг апаратного забезпечення можливо `psutil`. Для визначення максимальних частот ЦП реалізовано метод `psutil`, що виводить максимальну, мінімальну та поточну частоти роботи ЦП. Визначення завантаженості по ядрах ЦП можливо методом `psutil.cpu_percent()`.

Моніторинг оперативної пам'яті здійснюється шляхом використання методу `psutil.virtual_memory()`, що повертає словник який містить кількість доступної, вільної та задіяної оперативної пам'яті.

Опрацювання інформації про стан накопичувачів здійснюється методами: `psutil.disk_partitions()` для визначення наявних розділів та `psutil.disk_usage(partition.mountpoint)`, де `partition.mountpoint` один з визначених розділів для визначення властивостей певного розділу. Отримані дані можуть бути надіслані на віддалений комп'ютер для подальшого аналізу або виведена на консоль з застосування форматування (рис. 3).

```
Керівник: С:/Абшкратура/Техніч./Інформаційні Технології 2022/2021/Інстру
***** ЦП *****
Максимальна частота ЦП: 2400.00Mhz
Мінімальна частота ЦП: 0.00Mhz
Поточна частота ЦП: 2400.00Mhz
Використання ядер ЦП:
Ядро 0: 1.6%
Ядро 1: 1.6%
Загальне завантаження ЦП: 4.5%
***** Оперативна пам'ять *****
Загальна: 3.92GB
Доступно: 389.81MB
Використовується: 3.54GB
Відсоток використаної: 90.3%
***** Диск *****
*** Пристрій: C:\ ***
Точка монтування: C:\
файлова система: NTFS
Загальний розмір: 445.92GB
Використано: 307.24GB
Вільно: 138.68GB
Відсоток зайнятого простору: 68.9%
*** Пристрій: D:\ ***
Точка монтування: D:\
файлова система: NTFS
Загальний розмір: 350.00GB
Використано: 308.39GB
Вільно: 41.61GB
Відсоток зайнятого простору: 88.1%
>>>
```

Рисунок 3 – Моніторинг апаратного забезпечення засобами psutil

Отримана інформація може не лише допомогти налагодити роботу мережеслужб, а й виявити шкідливе програмне забезпечення на основі моніторингу завантаженості ЦП та оперативної пам'яті.

УДК 004.912

Кожушан М. Г., Кунгурцев О. Б.
Національний університет «Одеська політехніка»,
м. Одеса, Україна

ФІЛЬТРАЦІЯ ТЛУМАЧЕНЬ ТЕРМІНІВ В ЕЛЕКТРОННОМУ СЛОВНИКУ

Виникнення електронних словників спричинило скорочення часу на пошук інформації в порівнянні з паперовими словниками. Завдяки їм можна розв'язати проблему великого обсягу словника, забезпечити одночасний пошук не тільки за назвою словникової статті, але і по всьому змісту. Результатами пошуку є велика кількість тлумачень, фільтрація яких проводиться вручну, значно збільшуючи час, що витрачається на пошук необхідного.

Використання комп'ютерної лексикографії [1] допоможе вибрати релевантні тлумачення, що відповідають інтересам користувача, значно підвищивши продуктивність праці спеціаліста.

Метою роботи є скорочення часу на пошук тлумачення терміна, що відповідає досліджуваній предметній області (ПО) користувача. Задля реалізації зазначеної мети у роботі необхідно вирішити такі завдання:

- створити математичну модель словникової статті електронного словника;
- розробити алгоритм фільтрації тлумачень термінів;
- провести апробацію результатів фільтрації.

Уявимо математичну модель словникової статті. Кожна стаття є кортежем:

$$De = \langle tr, Ti \rangle \quad (1),$$

де tr – термін; Ti – тлумачення терміна.

Тлумачення терміна можна зобразити у вигляді множини дефініцій:

$$Ti = \langle \{Df_1, Df_2, \dots, Df_n\} \rangle \quad (2),$$

де Df_i – дефініція, один з варіантів тлумачення;

n – кількість варіантів тлумачення;

Так само дефініцію представимо у вигляді:

$$Df_i = \{tx, po\} \quad (3),$$

де tx – текст дефініції; po – предметна область використання.

Множину пошукових термінів представимо у вигляді:

$$Sts = \{tr_1, tr_2, \dots, tr_n\} \quad (4)$$

У роботі [2] розглянуто методику аналізу ПО за допомогою лексико-статистичного аналізу дефініцій та введених термінів для пошуку. Невелика кількість пошукових термінів ускладнює подібний аналіз та не фільтрує результати пошуку. Для розв'язання цієї проблеми запропоновано алгоритм фільтрації (рис. 1).

На вхід надходить результат пошуку у словнику - словникова стаття, дефініції якої будуть проходити фільтрацію. Далі кроки алгоритму залежать від знання користувачем теми, до якої належить термін. Якщо користувач вказав тему або набір термінів, що її характеризують, множина (4) розширюється на набір введених слів.

Перед фільтрацією необхідно попередньо підготувати дані. Розглянемо етапи обробки пошукових термінів:

- Графематичний аналіз – вилучення лексичних одиниць.

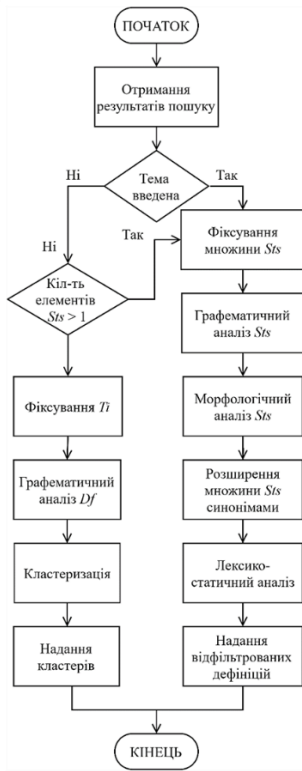


Рисунок 1 - Алгоритм фільтрації тлумачень

- Морфологічний аналіз – виділення лексичної одиниці, що несе основне семантичне значення (приведення до словникової форми).
- Доповнення синонімічними термінами, у яких єдиний варіант тлумачення.

Потім, у разі введення теми або множини термінів, переходимо до лексико-статичного аналізу [2]:

1. Підрахунок кількості пошукових термінів в дефініціях;
2. Сортування дефініцій за кількістю пошукових термінів;
3. Відкидання тлумачень з найменшою кількістю.

В іншому випадку переходимо до кластеризації дефініцій за допомогою латентно-семантичного аналізу [3].

В результаті застосування алгоритму користувач отримує кластеризований список дефініцій за тематиками або відфільтрований за досліджуваною ПО дефініції введених термінів.

Проведено апробацію алгоритму на 125-ти термінах у 4-х словниках. Час пошуку скоротився на 17%, відсоток коректно визначених тлумачень від загальної кількості пошукових термінів становить 74%.

Внаслідок застосування запропонованих методів час пошуку було скорочено, результати пошуку локалізовані під пошуковий запит.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Марчук Ю.М. Комп'ютерна лінгвістика АСТ, Схід-Захід, 2007. 141 С.

2. Kungurtsev Alexey B., Novikova N., Kozhushan M., Automation of searching for terms in the explanatory dictionary. 2020. DOI: 10.15276/opu.3.62.2020.11

3. Latent semantic analysis. URL: https://en.wikipedia.org/wiki/Latent_semantic_analysis

УДК 004.42:378.146

Кошовий Р. В., Кірей К. О.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ЗАСТОСУНОК ДЛЯ ОЦІНКИ ЗБІЖНОСТЕЙ ФАЙЛІВ ІЗ КОДОМ С-ПОДІБНИХ МОВ ПРОГРАМУВАННЯ

Нині з поширенням цифровізації майже всіх сфер життя спостерігається зростання попиту з боку бізнес-структури на ІТ-фахівців [1]. І як наслідок, зростають вимоги щодо підготовки студентів за відповідними спеціальностями. Одним із головних чинників забезпечення якісної підготовки майбутніх ІТ-фахівців є запобігання можливості появи плагіату серед робіт студентів на всіх етапах їхнього навчання.

За матеріалами Вікіпедії плагіат (від лат. *plagiarius* – викрадач) – це привласнення авторства на чужий твір або на чуже відкриття, винахід чи раціоналізаторську пропозицію, а також використання у своїх працях чужого твору без посилання на автора [2]. Наслідком плагіату є зниження вихідного рівня фахових знань студентів, формування негативних якостей особистості, таких як недоброросвісне ставлення до роботи, звичка халтурити тощо [3]. У процесі навчання студенти ІТ-спеціальностей постійно створюють сотні та тисячі рядків коду, який потребує перевірки та оцінки. Отже, такий процес перевірки є доволі трудомістким завданням. У процесі перевірки насамперед, зазвичай, необхідно з'ясувати чи був код запозичений, чи він є

оригінальним. Тут є декілька способів оцінки оригінальності коду. Перший – виявляти плагіат «у ручну», проте це потребує багато часу на пошук та оцінку розбіжностей. Другий – використовувати онлайн сервіси перевірки текстів на унікальність. Проте, у процесі перевірки треба враховувати особливості тексту з кодом, а цю особливість сервіси перевірки текстів на унікальність, зазвичай, не враховують. Отже, проблема автоматизації процесу перевірки текстів, що містять програмний код, на унікальність є актуальною.

У доповіді розглянуто можливі шляхи розв'язку проблеми перевірки програмного коду на запозичення в процесі здійснення поточного контролю знань студентів, що навчаються за спеціальностями галузі знань 12 «Інформаційні технології».

У процесі дослідження зазначеної проблеми були з'ясовані вимоги щодо застосування оцінки розбіжностей файлів із кодом С-подібних мов програмування. У якості предмету для поглибленого розгляду було обрано створення консольного застосування з використанням мови програмування С#, який має такі властивості:

- підтримка найпопулярніших С-подібних мов програмування (а саме: С, С++, С#, Java);
- декілька рівнів сканування коду;
- візуалізація процесу сканування;
- оцінка синтаксичного розміру коду та схожості файлів у числовому вигляді та вигляді відсоткового співвідношення;
- можливість вибору коду-донору та коду-акцептору для перевірки (важливо, якщо серед двох кодів є той, що більш ймовірно запозичений з іншого);
- коди подаються у вигляді файлів із розширеннями, які відповідають мові, підтримуваній даним «антиплагіатором».

В основі роботи програми оцінки розбіжностей файлів із кодом С-подібних мов програмування покладено посимвольний аналіз коду та його структурування на елементи ієрархічної (деревовидної) структури з рівнями «код», «блок», «команда», «слово» (синтаксична одиниця) згідно з типовою структурою програми на С-подібних мовах програмування. Цей процес реалізовано через автоматизацію виявлення всіх ключових для структури цих мов програмування символів (термінаторів, фігурних дужок тощо). Водночас враховано необхідність ігнорування коментарів та текстових (рядкових) літералів, які в протилежному випадку можуть вплинути на результат порівняння файлів, але не несуть собою важливої з точки зору самого коду інформації. Також ігноруються підключення бібліотек, просторів імен та інші команди, які зазвичай знаходяться на початку програмного коду

та, як і коментарі, не несуть смислового навантаження – їхній збіг або відмінність не свідчить про схожість програм.

Сканування реалізовано з допомогою системи підрахунку балів, кількість яких залежить від результату порівняння текстів коду програм. Водночас порівняння виконується через зіставлення окремих блоків двох програм між собою, під час якого уподібнюються окремі команди. Команди ж порівнюються відразу в декілька етапів, що мінімізує можливості реалізації «хитрощів» із приховування фактів запозичення програмного коду (пробіли, перестановки рядків та блоків тощо). Саме на етапі порівняння команд і нараховуються бали, але на всіх рівнях порівняння (команд, блоків) враховується саме найкращий збіг. Результатом же буде відношення фактично набраної кількості балів до теоретично можливої кількості за умови повної ідентичності програмних кодів, наданих у двох файлах.

Процес сканування та порівняння файлів супроводжується графічним відображенням у консолі, де є своєрідний рядок процесу виконання операції. Така візуалізація є компактною та наочно демонструє користувачу процес виконання програмою своєї роботи та приблизний час очікування результату. Процес порівняння займає певний час в залежності від фактичного розміру програмного коду у файлах, але вкладається в допустимі рамки для виконуваного завдання.

Отже, перевірка коду на оригінальність у навчальній практиці підготовки майбутніх ІТ-фахівців у ЗВО є важливим чинником забезпечення якісної ІТ-освіти. Проте проблема автоматизації процесу перевірки текстів, що містять програмний код, на унікальність ще не набула остаточного вирішення. Створення застосунку, який порівнює програмні коди та визначає наявність фактів запозичення або плагіату є корисним допоміжним засобом для викладача в процесі проведення поточного контролю набутих знань. Розглянутий у доповіді застосунок може бути удосконалений в залежності від більш конкретизованих цілей та сценаріїв використання, а також специфікований під інші мови програмування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Комп'ютер. *Wikipedia* : вебсайт. URL : <https://uk.wikipedia.org/wiki/Комп%27ютер> (дата звернення: 25.01.2022)\
2. Плагіат. *Wikipedia* : вебсайт. URL : <https://uk.wikipedia.org/wiki/Плагіат> (дата звернення: 25.01.2022)
3. Бурний В. Хронічний плагіат в українських ВНЗ. *Спільне*. 2016. : URL: <https://commons.com.ua/ru/pro-plagiat> (дата звернення: 25.01.2022)

МОБІЛЬНЕ ЗАСТОСУВАННЯ ДЛЯ ОБЛІКУ ОСОБИСТИХ КОШТІВ

Сучасний світ побудовано на догмі споживання. Дуже багато факторів життя пов'язано з використанням грошей. Багато-які послуги, речі та інші блага надаються виключно за оплату. Таким чином, люди витрачають час у погоні за грошима, а живеться легше тим, хто їх має. Деякі люди все життя присвячують збільшенню власного капіталу. В той же час, тих, хто не має достатньо коштів, зневажають. Однак мало просто заробляти гроші. Згідно деяких досліджень, розподіл коштів проводиться людиною хаотично, без певної системи чи структури. Таким чином виникає необхідність у кредитуванні або витраті раніше відкладених коштів[1]. Ефективним шляхом розв'язання цієї проблеми може бути контроль та облік фінансів. Планування може вважатись тим шляхом досягнення результатів, що дозволить не тільки грамотно та швидко накопичувати кошти, а й мати активну позицію у всіх у всіх питаннях, що пов'язані з персональним бюджетоутворенням[1]. Часом це допомагає якщо не отримати більше, то точно забезпечувати себе всім, що захочеться не тільки в поточний момент часу, а й у далекому майбутньому. Тому, очевидно, що контроль своїх витрат та доходів необхідний для грамотного розподілу фінансів. До того ж, саме невміння вести облік коштам часто стає причиною їх нестачі. Може виникнути дивна, однак доволі часта ситуація, коли зрозуміти, куди було витрачено доволі значні суми стає майже неможливо.

Базова логіка контролю бюджету полягає не просто у наявності відповідей на питання «Куди було витрачено кошти?» та «Яка сума витрат?», а у розподіленні всього бюджету на певні категорії та описі відповідних витрат. Так витрати можна розділити на ті, що відбуваються одноразово або дуже рідко, та регулярні. До останніх можна віднести, наприклад, комунальні платежі, витрати на транспорт, дітей, продукти харчування тощо[2]. Ці категорії є індивідуальними та визначаються у ході оцінки напрямків витрат, що має передувати плануванню бюджету на наступний період, якщо таке відбувається.

Описана логіка не є складною, однак має значну трудомісткість[2]. Необхідність записувати та аналізувати всі витрати може бути незручною, тому для полегшення процесу може бути

розроблене відповідне мобільне застосування. Ряд аналогів існує. Їх було переглянуто та проаналізовано. На основі цих спостережень і описано наступну концепцію програми для обліку особистих коштів.

Мобільне застосування в основі має список витрат користувача, що може бути розподілений за певними категоріями. Воно працює частково у фоновому режимі. За користувацькими налаштуваннями воно перехоплює пуш-повідомлення та смс від банків при оплаті картками, що знижує трудомісткість процесу обліку.

У списку своїх витрат користувач може переглянути їх та редагувати й розширити. Для цього треба натиснути елемент списку для перегляду деталей та двічі – для редагування. Останній пункт дозволяє не тільки внести зміни у вже записані дані, так і додати фото чеку. Фото може бути проаналізоване та переформатоване у набір записів, що дасть додаткову інформацію для аналізу. Дані зі списку видаляються свайпом вправо. Над списком є також поле, що відображає загальну суму витрат за місяць. Наявна кнопка, що сортує список за назвою місця витрати, її датою або сумою. Суми позначаються різними кольорами в певних діапазонах. Ця функція може бути налаштована користувачем.

Всі витрати мають мітки геолокації. Застосування пропонує карту місцевості з помітками. При натисканні на них можна переглянути деталі покупок у цьому місці – адресу, назву магазину, час покупки та відкрити вікно деталізації з фото чеку, якщо воно наявне.

Також доступний розділ статистики. В ньому відображаються дані, згруповані за призначенням покупки. Також можна вивести статистику за різні періоди, за місяцями. Статистика також відображається круговою діаграмою, де різними кольорами відображено ті ж групи витрат.

Користувач може експортувати дані, що його цікавлять, у текстовий формат: csv, txt, pdf, excel. Це робиться за відповідною кнопкою та параметрами налаштувань. Також ці дані можна зберегти у хмару, що є одним з розповсюджених способів організації зберігання даних та сумісного доступу до них. Для цього потрібно підключитись до хмари користувача, налаштувати формат збереження та експортувати.

Застосування аналізує статистику та видає поради щодо можливих змін у повсякденній рутині. Також користувач отримує відповідні повідомлення. В той же час є кнопка для отримання порад у будь-який час.

При використанні режиму сім'ї користувач може отримати список витрат не тільки свій, а й інших представників групи. Для цього всі учасники групи повинні бути авторизовані та надати дозвіл на зберігання копії їх даних про витрати на сервері застосування.

Налаштування програми дозволяють регулювати повідомлення, діаграми та поради. Повідомлення можуть надсилатись та не надсилатись, можна змінити частоту їх появи. Діаграми мають декілька стилістик зображення. Поради, що не приходять у повідомленнях, а отримуються у застосуванні за бажанням користувача можна не зберігати або видаляти у певні періоди. Також доступні зміни у кольоровій гамі діапазонів витрат та груп витрат.

ПЕРЕЛІК ДЖЕРЕЛ

1. Осадчий В.В. Методы ведения личного бюджета и стратегии управления расходами. CYBERLENINKA: вебсайт. URL: <https://cyberleninka.ru/article/n/metody-vedeniya-lichnogo-byudzheta-i-strategii-upravleniya-rashodami>

2. Чернова М.В. Моделирование семейного бюджета. CYBERLENINKA: вебсайт. URL: <https://cyberleninka.ru/article/n/modelirovanie-semeynogo-byudzheta>

УДК 004.725: 004.056

Фінік В. Ю., Журавська І. М.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ЦИФРОВІ ТЕХНОЛОГІЇ ОБРОБКИ НАБОРІВ ДАНИХ ДЛЯ ОЦІНЮВАННЯ РЕАБІЛІТАЦІЇ ПОСТ-ІНСУЛЬТНИХ ПАЦІЄНТІВ

Інсульт є однією з найбільш поширених і тяжких патологій, що характеризується високим ступенем летальності та рівнем інвалідизації населення. Хоч з прогресом медицини у світі рівень смертності внаслідок інсульту стає меншим, але Україна, станом на 2016 рік, посідає 41-е місце за рівнем смертності від інсульту (384 на 100 тис. населення) [1].

Проте, рівень захворюваності на інсульт майже вдвічі перевищує загальний рівень захворюваності в розвинених країнах Європи. При цьому, близько третиною хворих є працездатне населення. Хоча рівень смертності є відносно невеликим, але рівень інвалідизації після перенесення інсульту в Україні становить від 76 % до 85 %, що є

надзвичайно сумним, якщо порівнювати з країнами Європи, де даний показник варіюється від 25 % до 30 %.

Така невтішна статистика свідчить про необхідність створення певної системи аналізу, яка могла б оцінювати можливість реабілітації пацієнтів або ж навіть передбачувати ризик інсульту у здорових людей. Наразі не було знайдено прикладів подібного програмного забезпечення, тому резонним буде зробити висновок, що зараз дана відповідальність лежить на плечах лікарів, тож є доцільним створювати засоби полегшення їх роботи в даний час.

Для створення програмного забезпечення із вищезазначеними функціями по-перше необхідно розробити алгоритм для оброблення даних. У якості набору даних у такій розробці використаний DataSet, що зібраний Австралійським відділенням статистики з листопада 2012 р. по червень 2015 р. [3]. Діапазони параметрів, за якими проведений аналіз, наведений у табл. 1. Залежність параметрів руху від швидкості була проаналізована для підгрупи пост-інсультних пацієнтів на реабілітації обсягом $n = 27$ та для підгрупи умовно-здорових людей обсягом $n = 17$.

Таблиця 1. Діапазони параметрів формування DataSet для пост-інсультних пацієнтів на реабілітації

Назва параметру	Середнє значення		Середнє квадратичне відхилення		Діапазон	
	$n = 27$	$n = 17$	$n = 27$	$n = 17$	$n = 27$	$n = 17$
Вік	60,15	54,18	12,08	8,33	28–76	40–74
Обрана швидкість ходьби, м/с	0,73	1,2	0,32	0,19	0,29–1,23	0,75–1,46
Баланс Берга (бали)	47,70	–	6,79	–	25–55	–
Оцінка Фугла-Мейєра	22,85	–	6,95	–	9–34	–
Стать (чол./жін.)	18/9	9/8				

Було запущено процес розробки програмного забезпечення для аналізу даних руху пацієнта і подальшим виведенням медико-статистичних параметрів якості його реабілітації або ж можливості його реабілітації взагалі. Візуалізація результатів аналізу DataSet, що наведена на рис. 1, дозволяє покращити розуміння аспектів, що

впливають на можливість реабілітації після інсульту або ж на ризик захворювання.

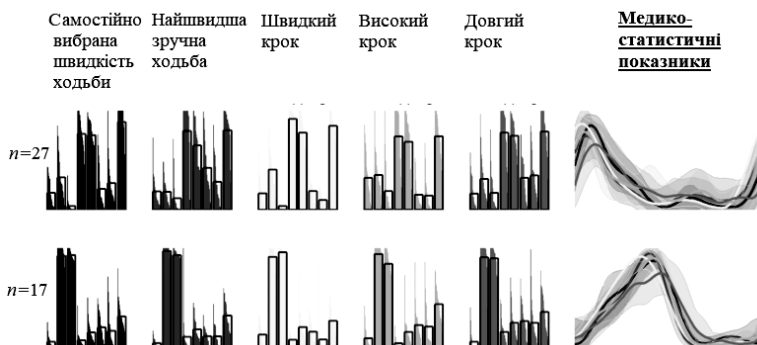


Рисунок 1 – Порівняння медико-статистичних показників пост-інсультних та умовно-здорових пацієнтів в залежності від характеристик руху

Згодом планується вдосконалити систему за допомогою передбачення ризику інсульту шляхом введення необхідних додаткових даних [3].

Таким чином, результати дослідження показують, що впровадження цифрових технологій для оцінки якості життя за різними медико-статистичними методиками (Берга, Фугла-Мейера, Спілбергера-Ханіна та ін.) може суттєво змінити уявлення про ефективність програм реабілітації пост-інсультних пацієнтів. Отримані результати підтверджують доцільність подальшого розвитку алгоритмів та програмного забезпечення для аналізу масивів даних із результатами програм медичної реабілітації для оцінки впливу реабілітації на якість життя пацієнтів після інсульту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Стасенко Т. Інсульт: впровадження успішних стратегій лікування. *Український медичний часопис*. Опубл. 12.01.2017. URL: <https://www.umj.com.ua/article/117650/insult-vprovadzhennya-uspishnih-strategij-likuvannya> (дата звернення: 30.12.2021).
2. Lynch E., Cadilhac D., Hillier S., Luker J. Assessment for rehabilitation after stroke collection / University of South Australia. URL:

<https://researchdata.edu.au/assessment-rehabilitation-stroke-collection/966394> (Last accessed: 30.12.2021).

3. Trunov A., Beglytsia V., Gryshchenko G., Ziuzin V., Koshovyi V. Methods and tools of formation of general indexes for automation of devices in rehabilitative medicine for post-stroke patients. *Eastern-European Journal of Enterprise Technologies*. 2021. Vol. 4, No. 2 (112). P. 35–46.

УДК 004.42

Червіцька К. В., Комлева Н. О.

*Національний університет «Одеська політехніка»,
м. Одеса, Україна*

ОСОБЛИВОСТІ АРХІТЕКТУРИ ВЕБСАЙТУ ДЛЯ КАФЕДРИ СИСТЕМНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Наразі будь-який структурний підрозділ у складі закладу вищої освіти повинен мати своє електронне представництво. Це надає можливість користувачам отримувати актуальну інформацію щодо діяльності цього підрозділу, його взаємодії з іншими підрозділами закладу вищої освіти, ефективно використовувати інформаційно-довідкові ресурси сайту [1, 2]. Використання сайту кафедри, в тому числі, для зворотного зв'язку зі студентами дозволяє підвищити якість навчання [3].

Метою роботи є створення та проектування гнучкого вебсайту кафедри системного програмного забезпечення Національного університету «Одеська політехніка» для отримання інформації студентами, абітурієнтами, викладачами, працівниками університету та іншими зацікавленими особами стосовно навчальної, методичної, наукової, організаційної та виховної роботи кафедри.

Інформація повинна легко оновлюватись, тому сайт потребує створення адміністративної панелі для зручної зміни як самого контенту, так і структури сайту. Основними функціональними вимогами до сайту є: можливість публікації нового контенту; редагування контенту; пошук необхідної інформації по сайту; зміна основних елементів сайту. У загальному вигляді динаміка роботи сайту наведена на UML-діаграмі (рис. 1).

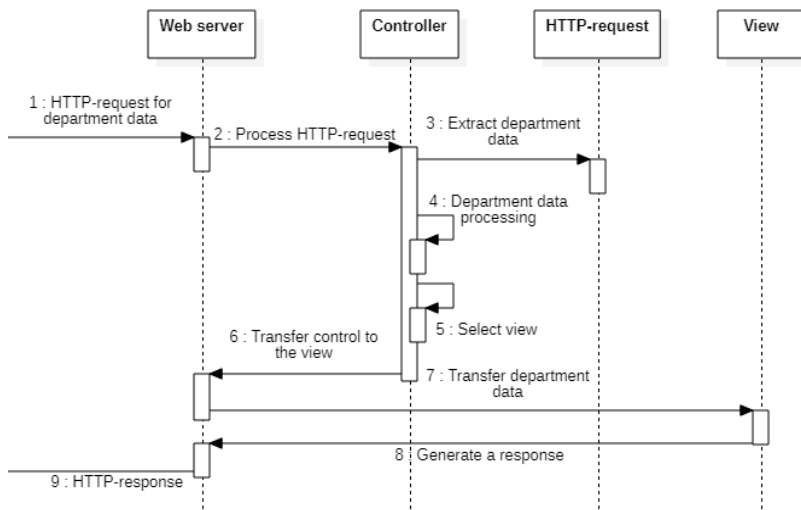


Рисунок 1 – Узагальнена діаграма послідовності

Особливістю даної архітектури є розбиття всієї системи на дві частини. А саме на адміністративну панель та головний сайт. Ці дві частини не взаємодіють один з одним напряму, їх зв'язок відбувається через базу даних (рис.2). Схема бази даних сайту наведена на рис.3. Тобто взаємодія з базою даних у вигляді редагування її полів відбувається лише у адміністративній панелі, а на головному сайті відбувається лише відображення цих даних, створити зміни у даних з головного сайту неможливо. Також з головного сайту немає доступу та можливості потрапити до адміністративної панелі.

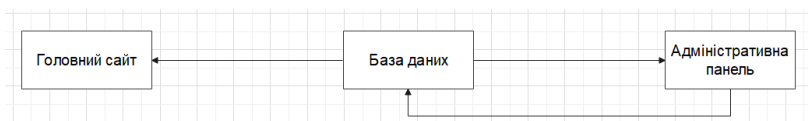


Рисунок 2 – Структура роботи вебсайту

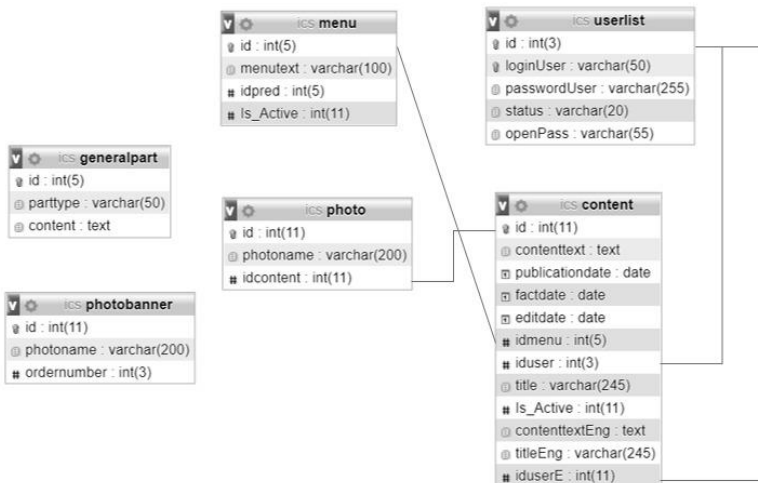


Рисунок 3 – Структура бази даних

Структура бази даних спроектована максимально гнучкою з тим, щоб можна було керувати структурою сайту: створювати, редагувати та видаляти рубрики, підрубрики, публікації, призначати права користувачам та ін.

Проведено тестування продуктивності розробленого сайту за таким показником як швидкість завантаження ресурсів. Для тестування використано стандартну платформу PageSpeed Insights від Google Inc. Даний ресурс дозволяє протестувати швидкість різних етапів завантаження веб сторінки. При тестуванні сайту враховувались такі етапи як час завантаження найпершого елемента інтерфейсу, індекс швидкості завантаження контенту та час до повної інтерактивності. Отримані результати відповідають сучасним вимогам щодо продуктивності.

Висновки. У роботі виконано аналіз та проєктування інформаційної структури сайту для кафедри. Визначено роль вебсайта кафедри як електронного представництва структурного підрозділу у складі закладу вищої освіти. Створено адміністративну панель, дизайн-макет та структуру бази даних вебсайту. Визначено особливості архітектури даного проєкту. Тестування продуктивності показало високу ефективність роботи сайту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Комлева Н.О., Любченко В.В., Зіноватна С.Л. Система підтримки прийняття рішень в управлінні якістю навчального процесу.

Науковий журнал «Проблеми програмування». Київ. 2020. № 2–3. С. 218–227. doi:10.15407/pp2020.02-03.218.

2. Комлева Н. О., Любченко В. В., Зіноватна С. Л. Інформаційна підтримка в управлінні якістю освіти //«Інформаційні управляючі системи і технології» (ІУСТ-ОДЕСА-2020) : матеріали ІХ Міжнародної науково-практичної конференції (24–26 верес. 2020 р., м. Одеса) / Одес. нац. політех. ун-т. Одеса : Екологія. 2020. С. 57–59.

3. Komleva N., Liubchenko V., Zinovatna S. Improvement of Teaching Quality in the View of a Resource-Based Approach. 16th International Conference on ICT in Education, Research, and Industrial Applications. October 06-10. 2020. Vol. 2740. P. 262–277.

УДК 004.42

*Червіцька К. В., Попова М. О., Комлева Н. О.
Національний університет «Одеська політехніка»,
м. Одеса, Україна*

ВИКОРИСТАННЯ ДОШКИ КАНВАН ДЛЯ ПРОЕКТУВАННЯ ВЕБСАЙТУ КАФЕДРИ ТА ОЦІНКИ ЙОГО ПОКАЗНИКІВ ЗРУЧНОСТІ ВИКОРИСТАННЯ

За останні два роки люди зрозуміли, що сучасні технології та Інтернет зайняли велику роль в звичайному житті та охопили всі сфери нашого життя. Адже прямо зараз всі знаходяться на карантині і дізнаються про події навколо лише через соціальні мережі та інтернет-портали. На даний час навчальні заклади постійно змінюють правила проведення навчання від повного переходу на онлайн-навчання до навчання у аудиторіях за певними правилами, а тому отримання актуальної інформації зростає. А тому й необхідність у розробці вебсайтів з релевантною та повною інформацією збільшилась у рази.

Метою роботи є підвищення гнучкості та прозорості процесу проектування та визначення показників зручності використання вебсайту кафедри системного програмного забезпечення національного університету «Одеська політехніка», що призначений для зручного отримання інформації щодо роботи кафедри.

Попередній аналіз вимог зацікавлених осіб щодо способу отримання інформації стосовно навчальної, методичної, наукової, організаційної та виховної роботи кафедри показав, що зручніше отримувати всю необхідну інформацію на одному ресурсі. Цей ресурс повинен мати інтуїтивно зрозумілий інтерфейс.

Дошка Kanban є інструментом гнучкого управління проектами [1]. Вона призначена для підвищення ефективності розробки проекту і може застосовуватися незалежно від того скільки осіб приймають участь у розробці. Дошка Kanban допомагає наочно уявити та упорядкувати завдання, при цьому кожне завдання виглядає як картка, що належить до певного списку (стовпця). Kanban-картки є візуальним артефактом, який призначений для відстеження завдань по мірі їх проходження по робочому процесу розробки проекту. На рис. 1 наведена Kanban-дошка для розробки сайту кафедри, створена з використанням зручного інструменту Trello.

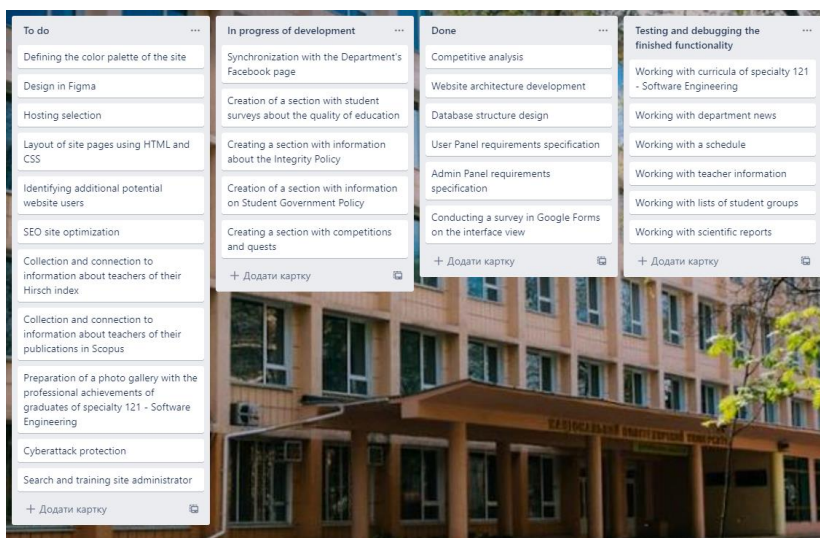


Рисунок 1 – Kanban-дошка Trello для розробки сайту кафедри

Як можна побачити, всі завдання згруповані за наступними стовпцями [2]:

- To do (потрібно розробити) – завдання сформульовано, ступень формалізації вимог може відрізнитись для різних карток, час, потрібний для реалізації завдань, може бути не визначений;
- In progress of development (у процесі розробки) – завдання, що вже взяті в розробку, але розробка ще не закінчена; з кожним завданням пов'язані виконавці (не показані на рисунку задля його компактності), час на виконання, зв'язки з іншими завданнями;

- Done (розробка завершена) – завдання, що відповідають функціональним та нефункціональним вимогам згідно зі специфікаціями;

- Testing and debugging the finished functionality (тестування та відладка реалізованих завдань у рамках супроводу продукту) – завдання, для яких незначно змінилися формальні вимоги, налаштування цих завдань можливо без повторного проходу по всім стовпцям Kanban-дошки.

Важливим показником створеного сайту є показник зручності його використання – юзабіліті (usability). Юзабіліті оцінюється по п'яти якісним критеріям: легкість у вивченні, ефективність, тобто наскільки швидко користувач орієнтуватися на сайті, запам'ятовуваність, рівень задоволення, а також те, як багато помилок роблять користувачі в користуванні сайтом.

При тестуванні юзабіліті в якості порівняльної характеристики обрано такі показники: кількість кроків для отримання інформації щодо навчального процесу, легкість використання сайту, заплутаність інтерфейсу, помилки в роботі користувача [3].

Складність інтерфейсу визначається оцінкою від 0 до 5, де 0 – інтерфейс простий у використанні, 3 – середня складність (багато кроків до досягнення мети або сплутаність кроків), а 5 – неочевидна поведінка сервісу, помилки, що заважають користуванню сервісом, відмови в роботі тощо. На основі даних тестування юзабіліті можна зробити наступні висновки:

- розроблений сайт потребує невелику кількість кроків для отримання інформації щодо навчального процесу та кафедри;
- сайт не має помилок, які б заважали користувачеві в процесі отримання інформації;
- інтерфейс сайту є максимально простим для користування.

У роботі показано, що застосування дошки Kanban сприяє прозорості процесу проєктування вебсайту кафедри та оцінки його показників зручності використання. Проведено аналіз, який показав, що всім зацікавленим особам, а саме студентам, абітурієнтам, викладацькому та адміністративному персоналу зручніше, коли вся актуальна інформація щодо діяльності кафедри знаходиться на одному сайті. За результатами проєктування сайту проведено анкетування зацікавлених осіб щодо ступеня їх задоволеності зручністю структури та вимогами до розроблюваного сайту. Результати анкетування показали достатній рівень задоволеності у 93% респондентів. Також приблизно 25% респондентів надали додаткові вимоги щодо структури сайту, що говорить про високий інтерес до цього усіх зацікавлених осіб.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Komleva N., Liubchenko V., Zinovatna S., Kobets V. Risk Management with Lean Methodology // Proceedings of 1st International Workshop on Computational & Information Technologies for Risk-Informed Systems (CITRisk-2020), Kherson, Ukraine, October 15-16. 2020. Vol. 2805. P. 266–281.
2. Komleva N., Liubchenko V., Zinovatna S. Evaluation of the Quality of Survey Data and its Visualization Using Dashboards. 15th International Scientific and Technical Conference «Computer Science and Information Technologies» Lviv Polytechnic National University. Lviv, Ukraine, September 23-26, 2020. Vol. 2. Lviv, 2020. P. 234–237. DOI: 10.1109/CSIT49958.2020.9321970
3. Liubchenko V., Komleva N., Zinovatna S., Pysarenko K. Framework for Systematization of Data Science Methods. Applied Aspects of Information Technology. 2021. Vol.4. No.1. Pp. 80–90. DOI: <https://doi.org/10.15276/aait.01.2021.7>

Вебтехнології та вебдизайн

УДК 004.05

Бабін О. С., Болюбаи Н. М.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ВЕБ-ЗАСТОСУНОК ЗАБЕЗПЕЧЕННЯ КОМУНІКАЦІЙНОЇ ДІЯЛЬНОСТІ В ІТ-КОМПАНІЯХ

Розбудова інформаційного суспільства в Україні супроводжується інтенсивним впровадженням інформаційних технологій у всі сфери життєдіяльності. Карантин, введений у зв'язку з пандемією COVID-19, прискорив процеси впровадження віддалених форм діяльності працівників різних компаній із використанням сучасних Інтернет-технологій та хмарних сервісів. ІТ-компанії усе частіше переходять до розробки програмного забезпечення в територіально розподілених командах, що обумовлює необхідність забезпечення ефективних та безпечних віддалених комунікацій для фахівців. Одним із шляхів вдосконалення та підвищення ефективності роботи ІТ-компаній в умовах територіально розподіленої роботи учасників команди є розробка вебзастосунку забезпечення їх

комунікаційної діяльності з надійною системою шифрування внутрішньо корпоративної інформації.

Процес комунікації в ІТ-компаніях є обміном інформацією між двома або більше учасниками проектних команд з метою забезпечення розуміння інформації, яка передається у процесі виконання ними функціональних обов'язків та особистих зв'язків усередині команди [1]. Використання Інтернет-технологій для розподіленого у просторі спілкування обумовило різке зростання кількості програмних засобів, які спрямовані на ефективне забезпечення комунікаційної діяльності організацій. До найкращих програм для ефективного командного ділового спілкування сьогодні можна віднести корпоративні месенджери Телеграм, Slack, Flock, Skype, Google Meet, Samerpage, хмарний сервіс Highfive, цифрове робоче місце, яке об'єднує розрізнені кадри Beekeser. Їх використання спрощує спілкування між колегами, дозволяє відправляти голосові та текстові повідомлення, забезпечує відеозв'язок, формує канали, які є робочим простором, дозволяє підтримувати та координувати різні графіки діяльності, що сприяє співробітництву та швидкому прийняттю рішень. Додаток MS Teams збирає в єдиний інтерфейс чати, збори, файли, завдання й календар, що дозволяє легко спілкуватися, вирішувати загальні завдання, обговорювати проєкт з колегами, планувати й втілювати плани в житті.

Проте питанням забезпечення безпеки корпоративні месенджери приділяють недостатню увагу. Одним із найбезпечніших месенджерів є Телеграм, у якому за безпеку відповідає власний протокол шифрування MTProto, що використовує відразу кілька технологій. Телеграм дозволяє організувати закритий корпоративний зв'язок, однак він недостатньо орієнтований на систему комунікацій ІТ-компаній. Тому доцільним є розробка вебзастосунку забезпечення комунікаційної діяльності в ІТ-компаніях із надійною системою шифрування внутрішньо корпоративної ділової інформації, який підтримує закриті спілкування та обмін діловою інформацією й допомагає усунути вузькі місця у проєкті і забезпечити безперерйну комунікацію з членами команди розробників.

Метою роботи є розробка вебзастосунку, який забезпечує ефективну та безпечну комунікацію в ІТ-компаніях із надійною системою шифрування інформації.

Із збільшенням кількості атак зловмисників на великі технологічні компанії зростають вимоги до рівня захисту як особистої так і корпоративної інформації. У результаті проведеного дослідження було виявлено, що для забезпечення надійності захисту використовують часткове або повне шифрування повідомлень та файлів, самознищення повідомлень і цілих чатів, блокування

можливостей для скріншотів. Різко знижує витік корпоративної інформації застосування багатофакторної автентифікації. Наскрізне шифрування (англ. end-to-end encryption), при якому доступ до інформації мають тільки ті користувачі, які приймають участь у спілкуванні, не дозволяє отримати доступ до криптографічних ключів третім особам, є найбільш надійним способом захисту даних. Для обміну ключами при цьому можуть бути застосовані симетричний та асиметричний алгоритми. Симетричне шифрування використовує єдиний ключ, який використовується обома сторонами, що спілкуються, для шифрування та дешифрування. Асиметричне шифрування використовує два ключі, один з яких є приватним, а другий – відкритим. Виявлено найбезпечніші алгоритми серед найбільш часто використовуваних алгоритмів шифрування: AES, Triple DES, Blowfish, Twofish, RSA [2].

Для розробки односторінкового вебзастосунку було використано JavaScript фреймворк React, стейт менеджери Redux та Thunk, бібліотека для створення інтерфейсу Bootstrap. Для створення серверної частини застосунку було використано програмну платформу Node.js, фреймворк для створення API ExpressJS, бібліотеку Speakeasy для двофакторної авторизації та бібліотеку Moments.js для роботи з датами. Для отримання й зберігання даних було обрано базу даних MongoDB та бібліотеку Mongoose. Для інтеграції з сервісами Google використано Google API. Тестування застосунку здійснювалося з допомогою середовища Jest та бібліотеки Enzyme.

Розроблений вебзастосунок надає можливість для проведення особистих онлайн зустрічей та сумісної роботи віддалено з інструментарієм для переписки, обміну файлами та управління проектами. Застосунок містить чат і соціальну мережу для підтримки спілкування всередині команди розробників, має інструментальні засоби для створення корпоративних каналів, які забезпечують спілкування фахівців під час роботи над проектами, дозволяє формувати та переглядати профайли розробників і створювати на їх основі резюме. Інтеграція з хмарним сервісом Google Calendar надає можливість створювати списки задач із вказівкою термінів їх виконання та відповідальними співробітниками, бот-канали для нагадувань та оповіщень співробітників, інтерактивні опитування, голосування, здійснювати розширений пошук по історії повідомлень. Застосування надійної системи шифрування надає принципово новий підхід до спілкування, дозволяє організувати закритий корпоративний зв'язок та підвищує надійність і безпеку роботи команди розробників ІТ-компанії, об'єднуючи їх у згуртований колектив.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Длугунович Н., Форкун Ю. Система комунікаційної діяльності в ІТ-компаніях. Вісник національного університету «Львівська політехніка». Серія: Інформаційні системи та мережі. Львів: Вид-во Львівської політехніки, 2017. № 872. С. 131–138.

2. Рогоза А., Задірака В. Порівняльний аналіз криптографічних алгоритмів. Матеріали III Міжнародної наукової конференції «Наукові теорії сьогодення та перспективи розвитку наукової думки» (ІСТУ-2019). Київ, 2019. С.105-109.

УДК 004.827; 004.89

Климчук А. М., Дворецький М. Л.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

АНАЛІТИЧНА СИСТЕМА КЛАСИФІКАЦІЇ КАНДИДАТІВ НА ВАКАНСІЮ НА ОСНОВІ АНАЛІЗУ ХАРАКТЕРИСТИК

Переважає більшість ІТ компаній, що існують у сучасному світі, та прагнуть розвитку, намагаються розширити бізнес шляхом залучення додаткових клієнтів, яким можуть запропонувати автоматизацію тієї чи іншої частини їх діяльності. При цьому невід’ємною передумовою якісних послуг є наявність людського ресурсу, що в змозі справитись із потоком інформації. Отже складовою діяльності ІТ компанії є пошук кваліфікованих співробітників, які роблять можливим її сталий розвиток. У деяких окремих випадках, та переважно у порівняно невеликих ІТ компаніях відділ кадрів, як таких, може бути відсутнім, а підбором персоналу при цьому займаються аутсорсингові компанії, але рано чи пізно зазначений відділ у тому чи іншому вигляді входить до складу компанії. Зазначимо, що до обов’язків відділу кадрів (або управління персоналом) окрім самого обліку входить також пошук кандидатів, із подальшою інтеграцією та оцінкою якості роботи.

Для ринку інформаційних технологій пошук та підбір (а також можливо подальше навчання) кандидатів на вакансію є чи не найголовнішим завданням фахівця з управління персоналу (HR-менеджера). Однією з причин може бути зазвичай вузька направленість ІТ спеціалістів, через що їх задіяння на різних проектах, над якими працює компанія, може стати проблемою. Крім того, утримання високого рівня спеціаліста ІТ галузі як правило обходиться не дешево, а отже компанія не може собі дозволити його утримання у надії на те,

що відповідний проєкт незабаром з'явиться. Через це HR-менеджеру постійно доводиться здійснювати пошук нових кадрів, виконуючи моніторинг резюме кандидатів на вакансію із використанням різних інформаційних ресурсів та онлайн-платформ. Через це також актуальною постає задача відстеження дублювання кандидатів на різних ресурсах, що значно ускладнює процес підбору і відповідно вимагає додаткового часу для ухвалення рішення.

Інформаційно-аналітична система класифікації кандидатів на вакансію на основі аналізу їх характеристик створить передумови для підвищення якості та швидкості процесу пошуку кандидата шляхом автоматизації збору та збереження даних з онлайн платформ пошуку роботи, інтеграції даних та їх відображення із використанням єдиного користувацького інтерфейсу. Розробка такого роду системи надає ряд наступних переваг та додаткових можливостей. Серед них наступні:

- спрощення процесу моніторингу нових резюме з ряду онлайн платформ;
- об'єднання дублікатів резюме кандидатів на вакансію;
- автоматична оцінка рівня корисності з використанням баз правил на основі нейронної мережі, що дозволяє оцінити залежності компетенцій та рівня кваліфікації;
- виконання аналізу ринку працівників в аналітиці вакансій у інших компаніях, із урахуванням досвіду, володіння стеком технологій розробки, враховуючи рівень заробітної плати та інші особливі характеристики кандидатів;
- виконання аналізу ринку працівників по вакансіям інших компаній, із урахуванням досвіду роботи, володіння тими чи іншими технологіями розробки, рівня заробітної плати та інших важливих властивостей кандидатів;
- оцінка та аналіз якості роботи менеджера і підбору персоналу;
- створення профайлів кандидатів та співробітників ІТ компанії, що надає повну інформацію, яка може бути задіяна для аналізу при пошуку кандидата та підвищить якість роботи інформаційної системи.

Під час реалізації системи у вигляді програмного вебзастосунку використано середовище Node.js та наступний базовий стек технологій веброзробки: HTML, CSS та JavaScript. Відповідно це дозволило виконати реалізацію бізнес логіки та інтерфейсу. Щодо механізмів видобутку даних кандидатів з онлайн ресурсів, його реалізовано на основі бібліотеки puppeteer, яка дозволяє працювати із використанням режиму емуляції браузера та отримування контенту сторінок у відповідному форматі. Однією з переваг бібліотеки є наявність гарної

документації та представлення вихідного коду на GitHub, що значно спрощує процес її застосування у межах проєкту. Виходячи із того, що формат даних на різних інформаційних ресурсах відрізняється, парсинг тексту реалізовано із використанням паттерну «адаптер». Наведений підхід дозволяє привести дані до одного формату із подальшим зберіганням результатів їх представлення у базі даних.

Для оцінки наскільки кандидат відповідає критеріям пошуку було застосовано нейронну мережу. Нейронна мережа дозволяє провести оцінку кандидата, із використанням навчальної вибірки на основі накопичених даних у системі. Реалізацію мережі виконано із використанням бібліотеки Brain.js [1], яка має у своєму розпорядженні механізми вирішення задач створення нейронних мереж з мінімальними часовими витратами у середовищі Node.js.

Надавши кожному кандидату оцінку рівня відповідності вимогам вакансії, залишається дублювання за рахунок надходження даних з декількох платформ. Для розв'язання задачі було розроблено та розроблено алгоритм, що дозволяє об'єднати декілька резюме в одне за результатами аналізу їх контенту. Використання алгоритму на тестових даних дає змогу говорити про зменшення кількості резюме у середньому на половину. А це у свою чергу дозволяє підвищити швидкість пошуку та підбору потенційних кандидатів на вакансію.

Алгоритм аналізу вмісту включає два альтернативні шляхи перевірки вмісту. Перший реалізовано на основі алгоритму шинглів [2], та дозволяє виконувати пошук по отриманих даних для знаходження схожих між собою резюме. Цей алгоритм надалі є ефективним тільки у випадку однакової мови написання резюме, тому було здійснено модифікацію алгоритму, що спочатку визначає мову резюме із подальшим виконанням подвійного перекладу. Для розв'язання задачі перекладу було використано API інтерфейс від Google Translate.

Другий спосіб передбачає аналіз фотокартки кандидата, яка у більшості випадків є ідентичною (або майже ідентичною). Прийнято рішення використати алгоритм перцептивного хешу [3], який задовольняє усім основним вимогам та показав свою ефективність. Об'єднання двох вищенаведених підходів підвищило рівень ефективності фільтрації.

Розроблена система дозволяє підвищити ефективність роботи менеджера з підбору персоналу шляхом автоматизації окремих операцій процесу підбору кандидата на вакансію, серед яких: перегляд та пошук нових резюме на онлайн ресурсах пошуку роботи, дослідження стану ринку а також оцінювання кандидатів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.

1. Getting Started with Machine Learning using Brain.js. Section: вебсайт. URL: <https://www.section.io/engineering-education/ml-with-brainjs/> (last accessed: 10.11.2021)
2. Зеленков Ю. Г., Сегалович И. В. Сравнительный анализ методов определения нечетких дубликатов для вебдокументов. URL: http://rcdl2007.pereslavl.ru/papers/paper_65_v1.pdf (дата обращения: 11.11.2021)
3. Приходько О. І., Карпенко С. Г. Пошук подібних зображень перцептивними методами. URL: http://apeps.kpi.ua/downloads/Приходько_О.І..pdf (дата звернення: 12.11.2021)

УДК 004.6

Кондратюк І. В., Воробйова А. І.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

КОНВЕРСІЯ ЕЛЕКТРОННОЇ ТОРГІВЛІ В SHOPIFY

Конверсія електронної торгівлі – це загальна кількість покупок, поділена на загальну кількість сеансів у e-commerce магазині Shopify. Як і показник відмов, важливо оцінювати показники як настільних комп'ютерів, так і мобільних пристроїв окремо, враховуючи, що коефіцієнти конверсії (і фактори, які їх змінюють), як правило, сильно різняться.

Коефіцієнт конверсії електронної торгівлі може мати безліч змінних, які можуть на нього вплинути, тому важливо мати структурований підхід для того, щоб покращити коефіцієнт конверсії. Згідно з дослідженнями порталу Littledata.io[1], у червні 2019 року були наступні показники :

1) Середній коефіцієнт мобільної конверсії для власників магазинів Shopify складає 0,9%.

Коефіцієнт мобільної конверсії більше 2,2% означає, що магазин входить до 20% найкращих магазинів Shopify.

2) Середній коефіцієнт конверсії настільних комп'ютерів для власників магазинів Shopify складає 1,4%.

Коефіцієнт конверсії настільних комп'ютерів більше ніж 3,1% означає, що магазин входить до 20% найкращих магазинів Shopify.

Процес оптимізації конверсії

Крок 1 – Якість трафіку

Інформація про показник відмов є корисною для показника конверсії, суть в тому, що якість означає «готовність до покупки».

Наприклад, часто буває, що при різкому збільшенні рекламного бюджету FB або бюджету PPC (тобто більше 25%) загальна якість трафіку може впасти. Якщо адміністратор Shopify або аналітика Google показують падіння коефіцієнта конверсії електронної комерції в ці періоди, це зазвичай вказує на те, що конкретна аудиторія вичерпана, і необхідно знайти нову аудиторію, щоб залучити кращий трафік, який бажає купувати.

Крок 2 – цільова сторінка для кожного джерела трафіку

Кожне платне або органічне джерело трафіку може мати одну або кілька певних сторінок, на які спрямовується трафік. Це може бути певна цільова сторінка, сторінка колекції або сторінка продукту. Використовуючи Google Analytics, можна вивчити потенційні проблеми з цільовими сторінками.

Sessions	Bounce Rate	Pages / Session	Avg. Session Duration	Ecommerce Conversion Rate	Transactions	Revenue	Avg. Page Load Time (sec)
1,903 % of Total: 28.52% (8,796)	3.10% Avg for Total: 2.82% (0.49%)	4.96 Avg for Total: 4.32 (1.16%)	00:01:12 Avg for Total: 0:01:32 (1.68%)	1.00% Avg for Total: 0.20% (7.60%)	19 % of Total: 24.12% (80)	£2,558.70 % of Total: 20.02% (87,751.48)	4.16 Avg for Total: 3.94 (2.59%)
45 (2.36%)	0.00%	4.53	00:00:51	2.22%	1 (0.50%)	£30.00 (1.17%)	8.04
391 (20.55%)	2.30%	4.59	00:01:06	0.26%	1 (0.50%)	£128.00 (5.00%)	6.57
76 (4.00%)	0.00%	7.71	00:01:48	1.32%	1 (0.50%)	£120.00 (4.69%)	4.49
417 (21.91%)	3.60%	3.63	00:00:12	0.00%	0 (0.00%)	£0.00 (0.00%)	3.84
59 (3.10%)	0.00%	4.92	00:01:24	1.69%	1 (0.50%)	£66.50 (2.60%)	3.82
20 (1.05%)	40.00%	5.55	00:02:41	0.00%	0 (0.00%)	£0.00 (0.00%)	3.77
150 (7.88%)	1.33%	3.48	00:00:51	0.67%	1 (0.50%)	£120.00 (4.69%)	3.72
44 (2.30%)	0.00%	2.31	00:00:05	0.00%	0 (0.00%)	£0.00 (0.00%)	3.52
284 (14.92%)	2.82%	8.87	00:02:32	2.82%	8 (40.00%)	£1,334.00 (51.67%)	3.51

Рисунок 1 – Звіт Campaigns в Google Analytics

На рис.1 використовується змінена версія звіту "Campaigns" в Google Analytics. Наведені дані належать до рекламних кампаній FB з однаковим рекламним контентом, але з різними цільовими сторінками. Як можна бачити, хоча рівень трафіку для двох кампаній загалом є схожим, кампанія, що веде до цільової сторінки з часом завантаження 6,57 секунд, має коефіцієнт конверсії 0,26%, тоді як кампанія, що веде до цільової сторінки з часом завантаження у 3,51 секунди має коефіцієнт конверсії 2,82%. Хоча це не доводить і не спростовує того, що швидкість є основним фактором, можна бачити, як цільові сторінки можуть суттєво вплинути на коефіцієнт конверсії магазину Shopify.

Крок 3 - Сторінка продукту

Дизайн та функціональність сторінки продукту, безперечно, є однією з найскладніших частин магазину Shopify для аналізу та оцінки. Залежно від того, як використовується сторінка продукту (цільова сторінка або вторинна сторінка з цільової сторінки), важливо подивитися, як відвідувачі переходять на сторінку.

З точки зору оцінки того, як відвідувачі переміщуються по сторінці продукту, використання таких інструментів як Lucky Orange або HotJar є відмінним способом виявлення проблемних питань.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. 6 essential benchmarks for Shopify stores by Edward. Littledata.io: веб-сайт. URL: <https://blog.littledata.io/2019/06/14/6-essential-benchmarks-for-shopify-stores/>
2. What is the average bounce rate from desktop Google search for Shopify? Littledata.io: веб-сайт. URL: веб-сайт. URL: <https://www.littledata.io/average/bounce-rate-from-desktop-Google-search/Shopify>
3. 40+ Ecommerce Metrics Benchmarks (2021). Store Growers: веб-сайт. URL: <https://www.storegrowers.com/ecommerce-metrics-benchmarks/>

УДК 004.9

*Лукіна А. В., Обухова К. О., Ткаченко М. П.
Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

РОЗРОБКА ВЕБСАЙТУ З SEO-ПІДТРИМКОЮ ТА МАРКЕТИНГОВИМ АНАЛІЗОМ

З розвитком мережі Інтернет поступово міняються підходи до управління компанією і відповідно маркетинговою діяльністю. У випадках неефективної та малоприбуткової роботи вебсайту підприємства для встановлення причин невдач слід застосувати маркетинговий аналіз, тобто комплексну оцінку та перевірку готового сайту з точки зору зручності користування, аналіз аудиторії сайту, стратегії просування, оцінку конкурентоспроможності ресурсу [1].

Актуальність теми дослідження обумовлено підвищенням значущості маркетингової діяльності як вагомого чинника в забезпеченні конкурентоспроможності торгово-посередницького підприємства в умовах ринкових відносин.

Співвідношення між попитом на ринку та можливостями підприємства повинно бути враховано при розробці нових напрямків та цілей підприємницької діяльності, а також маркетингових програм і стратегічних планів. Тому об'єктами маркетингового аудиту вебсайту підприємства можуть виступати: аналіз якості інформації сайту,

тестування існуючих сервісів сайту, аналіз поточної стратегії просування ресурсу тощо (рис. 1).



Рисунок 1 – Напрями маркетингового аналізу вебсайту

Вивчення впливу маркетингового внутрішнього та зовнішнього середовища на діяльність торгово-посередницького підприємства, вивчення потреб цільових споживачів на попит і реалізацію продукції обумовлюється необхідністю вдосконалення маркетингової діяльності, від ефективності якої залежить майбутнє компанії в цілому [2].

Метою роботи є розробка напрямків взаємозв'язку між створенням програмного забезпечення та маркетингом. У цій роботі, буде розглянуто питання, чому саме при розборці вебсайтів необхідно володіти навичками інтернет просування.

Об'єктом дослідження є процес створення сайтів та аналіз напрямків ефективної маркетингової діяльності в забезпечені конкурентоспроможності підприємства в сучасних умовах господарювання.

Предметом дослідження є теоретичні та методичні положення, щодо розробки шляхів забезпечення ефективної маркетингової діяльності підприємства та вихід на онлайн ринок ТОВ «GBAR»

У процесі розробки програмного забезпечення для Інтернет-магазину були використані методи створення та тестування програмного забезпечення, метод статистичних досліджень, метод експертних оцінок, метод порівняння та групування для дослідження підприємства «GBAR». У процесі роботи використовувалась інформація, надана підприємством «GBAR».

Результати роботи використовуються підприємством «ГВАР» для удосконалення внутрішнього та зовнішнього програмного забезпечення та виходу на новий рівень співпраці з аудиторією.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Корягіна С. В., Корягін М. В. Маркетинговий аудит: навч. посіб. Київ : Центр учбової літератури, 2014. 320 с.
2. Голяш І. Д. Маркетинговий аудит електронних ресурсів підприємства. URL: http://elartu.tntu.edu.ua/bitstream/lib/21046/2/TRM_2017_Holiash_I_D-Marketynhovyi_audyt_elektronnykh_79-92.pdf (дата звернення: 30.12.2021).

УДК 004.42

*Медяник Є. І., Рудніченко М. Д.
Державний університет “Одеська Політехніка”
м. Одеса, Україна*

ВИКОРИСТАННЯ ФРЕЙМВОРКУ LARAVEL ПРИ РОЗРОБЦІ ВЕБЗАСТОСУНКІВ НА ОСНОВІ MVC

У веброзробці потрібно багато часу для кодування серверної частини, включаючи бізнес-логіку та інтеграцію інтерфейсу, що є механізованим і високоповторним. Для вирішення цих завдань у внутрішньому вебкодуванні, використовується багата кількість вебфреймворків, зокрема (Spring, SpringMVC, Laravel) і шаблони розробки вихідного та кінцевого поділу. Інтегруючи механізм шаблонів у веброзробку PHP, можна досягти технології генерації коду для серверної частини веб, що сприяє штучному інтелекту вебкодування та економить час на розробку програмного забезпечення. Шаблон MVC — це інтерактивний шаблон дизайну структури інтерфейсу, а також ефективний шаблон дизайну. Шаблон проектування MVC [1] підкреслює, те що розробники проектують своє програмне забезпечення модульно на основі моделі даних та відображення інформації. Він поділяє прикладне програмне забезпечення на наступні три частини: Модель (Model) — це бізнес-логіка прикладного програмного забезпечення, включаючи дані та модель. Модель є основною функціональною частиною застосунку, таким чином інкапсулюючи дані та процес вирішення проблем. Користувач викликає модель, викликаючи контролер, щоб модель

могла надавати дані для Views, щоб зменшити надмірність коду. Перегляд — це частина прикладного програмного забезпечення, яка обробляє відображення даних, які у свою чергу відображають інформацію різного типу. Перегляд – це інтерфейс, який бачать і з яким взаємодіють користувачі. Для перегляду даних та процесу вирішення проблеми користувачі можуть надсилати запити на обслуговування до моделі, викликавши контролер для оновлення представлення. Контролер - це частина програмного забезпечення, яка обробляє взаємодії. Контролер перетворює введені дані такі, як сервісний запит для моделі та перегляду, і він відповідає на зміни моделі для перегляду. Користувачі взаємодіють лише з програмним забезпеченням Controller[2]. Один набір представлень і контролерів становить інтерфейс користувача, а модель може мати багато представлень. Якщо користувач змінює модель через контролер, контролер поширить цю зміну на всі представлення, щоб досягти єдності дисплея.

Middleware спільно використовує механізм, за допомогою якого воно фільтрує HTTP запити на введення програми. Наприклад, за замовчуванням існує проміжне програмне забезпечення в Laravel, яке зазвичай аутентифікує user, перевіряючи користувача. Якщо користувача не аутентифіковано, тоді middleware автоматично перенаправляє користувача на екран входу та якщо користувач пройде аутентифікацію, то це дозволить запиту рухатися далі в межах програми. Створення додаткового проміжного програмного забезпечення може слугувати для виконання різноманітних завдань, крім аутентифікації. Існує кілька проміжних програм, яких інтегровано в структуру Laravel, усе це проміжне програмне забезпечення знаходиться в каталозі app/Http/Middleware. Зазвичай розробник може змінювати існуючий або створити нове middleware для програми.

Laravel розроблений таким чином, що він покращує якість програмного забезпечення за рахунок зниження вартості початкових витратів на розробку та поточного обслуговування та забезпечення чіткого виразного синтаксису для покращення роботи з програмою та основним набором функцій ,щоб заощадити час виконання. Laravel є одним з небагатьох PHP фреймворків, які забезпечують справжню модульність коду. Це робиться за допомогою поєднання драйверів і серії пакети. Драйвери допомагають налаштувати та розширити функції кешування, сеансу, бази даних та швидкої аутентифікації. Laravel також пропонує передовий набір інструментів взаємодії з

базою даних. Можна легко спроектувати та змінити базу даних за допомогою міграцій баз даних незалежним від платформи, способом завдяки якому не буде проблем із сумісністю. Реалізація активного запису Laravel називається Eloquent. Це сучасний спосіб взаємодії з базою даних об'єктно-орієнтованим способом. З Eloquent, без необхідності писати SQL код, ми можемо створювати, отримувати, оновлювати та стирати записи бази даних. Eloquent має потужні інструменти управління, а також обробляє сторінки автоматично і заощаджує час.

Laravel надає наступні основні можливості створення вебзастосунків: Laravel пропонує 20 бібліотек і модулів, які об'єднані, що дозволяє розширити структуру. Це пропонує інтегрувати 20 модулів і бібліотек для покращення застосунків. Завдяки тому, що кожен модуль є покритим диспетчером залежностей Composer, це спрощує оновлення. Маршрутизація надає користувачеві універсальний спосіб окреслення маршрути у інтернет-службі. Це допомагає масштабувати застосунок кращим чином і підвищить його загальне значення продуктивності. Laravel надає елегантність поштовому сервісу, яка дозволяє надсилати пошту з багатим вмістом. Laravel має попередньо визначені функції, такі як вхід, реєстрація, "забули пароль" і нагадування пароля яке сприяє розробці аутентифікації. Основними перевагами Laravel є посиленій захист: Laravel забезпечує підвищену безпеку даних, а також користувачів, відмінна інтеграція поштових служб: зазвичай поштові послуги використовуються як сповіщення для користувачів, які слідкують за будь-якими подіями. Laravel фреймворк пропонує потужний і чистий API, над знаменитою бібліотекою Swift Mailer. Він також пропонує міцне рішення для SMTP, Amazon SES, Mandrill і «Send mail», що допомагає програмі надсилати пошту через хмарного оператора, надійна інтеграція інструментів для швидкої розробки: програма має бути гнучкою та структурованою. Таким чином, це, нарешті, призведе до значного зростання рентабельності інвестицій. Laravel підтримує такі серверні системи, як Redis і Memcached. Він ефективно використовує драйвер кешу файлів, оскільки він добре налаштований таким чином, щоб зберігати кешовані компоненти у файловій системі динамічно. Робота з автоматизованого тестуванням: тестування має вирішальне значення для будь-якої успішної розробки та розгортання вебсайту. Автоматизоване тестування підтримується Laravel, і це не тільки

економить багато часу, але також надавати точніші звіти, ніж ручне тестування.

У порівнянні з іншими фреймворками PHP, Laravel відрізняється простотою структурою каталогів, файлів і є найкращим у роботі з файловою системою, процесом міграцією з однієї платформи на іншу. Щодо розробки PHP, дуже важко перенести базу даних з однієї платформи до іншої, але для Laravel це легке завдання. Крім того, існують попередньо визначені команди Laravel, які вбудовані для розуміння базової архітектури Laravel. Так розробники краще та швидше можуть зрозуміти архітектуру фреймворку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Sambhav R. What are the reasons to choose Laravel MVC for веб development? Poole, United Kingdom: MSC IT. Research Methods and Professional Issues, 2019. 5 p.

2. Majida L., Khaoula B. A comparative study of laravel and symfony PHP frameworks. Morocco: IJECE, 2018. 10 p.

УДК 004.9

*Павлющенко М. М., Обухова К. О., Ткаченко М. П.
Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КОМПАНІЇ-РЕАЛІЗАТОРА АВТОМОБІЛІВ З ПРОБІГОМ

На даний час важко уявити ефективну роботу будь-якого типу підприємства без власного сайту. Для компанії-реалізатора автомобілів – це можливість представити широкий асортимент та швидкий вихід на клієнтів. Потрібно зазначити, що повна та доступна інформація зацікавлює клієнтів, що у майбутньому сприятиме збільшенню продажів.

Актуальність теми зумовлено тим, що автомобіль – це одна з найважливіших складових повсякденного життя:

- автомобіль потрібен для вирішення виробничих проблем – швидке пересування у будь-яку точку міста, доставлення вантажів;
- для людей з обмеженими можливостями та людей похилого віку автомобіль забезпечує мобільність та покращує якість життя;

– у віддалених місцевостях автомобіль є з небагатьох засобів пересування та зв'язку з районними та обласними центрами, сприяє розвитку приватного бізнесу;

– автомобіль потрібен для комфорту у повсякденному житті, робить вихідні та відпустку набагато змістовнішими та цікавішими.

Тому в сучасному світі автомобіль – це необхідність і, навіть, основний засіб заробітку для великої кількості людей.

На сьогодні простежується деяка нестача нових автомобілів у офіційних дилерів тих чи інших автомобільних брендів України та світу. Це пов'язано з сучасними карантинними обмеженнями у світі та переорієнтування великих виробників комплектуючих до автомобілів (а саме автомобільних плат) на випуск плат для іншої електронної техніки. Слід, зазначити що більшість таких виробників знаходяться у Китаї, тому європейські виробники (такі як Volkswagen, Škoda тощо) опинилися в надзвичайно скрутному становищі.

Величезна кількість людей, для того, щоб не чекати по 3–6 місяців, поки збереться потрібний автомобіль, готові розглядати автомобілі з пробігом і позитивною історією. Тому з кожним днем такі авто стають все більш актуальними. На сьогодні подібних варіантів достатньо, тому сайт для компанії-реалізатора автомобілів з пробігом – це один з актуальніших проєктів.

Метою цієї роботи є створення зручного та надійного сайту для продажу, перевірки історії та перегляду необхідної інформації про автомобілі, які продають фізичні або юридичні особи.

Однією з корисних можливостей сайту буде послуга «Викуп автомобілів»: для клієнтів, яким необхідно терміново продати свій автомобіль, компанія надасть послугу Trade-In, яка передбачає огляд автомобіля і повний викуп його за 70 % вартості від низу актуального ринку.

На сьогодні ринок України наповнюється автомобілями з пробігом із різних частин світу – Америки, Європи та Азії. Деякі клієнти задля економії готові розглядати варіанти авто, які ще не знаходяться на території України, але вже мають фіксовану ціну. Тому було прийнято рішення створити основне меню сайту, що складається з трьох частин, що мають такі назви:

- 1) «Авто в Україні»;
- 2) «Авто в дорозі»;
- 3) «Викуп авто».

Пункт «Авто в дорозі», в свою чергу, буде складатися з:

- 1) «Авто з США»;

- 2) «Авто з Європи»;
- 3) «Авто Кореї».

Карта сайту для інформаційної системи компанії-реалізатора автомобілів з пробігом представлена на рис. 1.



Рисунок 1 – Карта сайту

Розробка інформаційної системи є актуальною в цей час, тому що це підвищує ефективність діяльності компанії-реалізатора автомобілів й полегшує роботу з вебсайтом для клієнтів-користувачів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Як створити ефективний сайт, який продає в 2021 році. URL: <https://rubika.com.ua/ua/blog/jak-stvoriti-efektivnij-sajt-jakij-prodaie> (дата звернення: 30.12.2021).
2. Поняття про інтерактивні вебсторінки та засоби розробки сценаріїв. URL: <https://lektsii.com/1-30805.html> (дата звернення: 30.12.2021).

УДК 004.042

Полосмак Н. В., Донченко М. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ ІЗ ЗАСТОСУВАННЯМ МЕТОДІВ АНАЛІЗУ ДАНИХ

Технології не стоять на місці, кожна галузь знань постійно покращується, все більше стає поширеним купівля будь-яких товарів в інтернет-магазинах, багато компаній прагнуть перевести бізнес в

онлайн. Але серед великої кількості вебресурсів знайти дійсно зручну платформу стає все більш важче. Крім того, на сьогоднішній день дуже важко знайти спеціаліста, який би зміг підтримувати створений вебресурс, якщо він написаний на будь-якій мові програмування. Сучасні засоби створення вебсайтів дають можливість створювати інтернет-магазини за допомогою CMS – системи управління контентом, яка дозволяє змінювати, додавати, видаляти інформацію на сайті без залучення розробників та без необхідності знань програмування.

При реалізації e-shop було проведено якісний аналіз ринку e-commerce платформ. Метою аналізу є визначення найбільш цілісної і якісної CMS. У якості вхідних критеріїв використано:

- Проста інсталяція. Процес повинен бути максимально задокументований, спрощений і послідовний.

- Швидкий старт. CMS повинна бути максимально відкритою і простою до користувача вже з перших кроків використання системи.

- Якісна документація. Інструкції повинні бути настільки прості, щоб нетехнічний персонал міг зрозуміти їх, але досить докладні, щоб їх легко було виконувати «покроково».

- Унікальність. На жаль, більшість відкритих CMS проєктується мало не за єдиним образом і подобою. Функціональність може відрізнитися і позиціонуватися розробниками як перевага, проте часто в цілому один програмний пакет може походити на інший, як дві краплі води.

- Гнучкість. Гнучкість використання, розширюваність – можливість пристосування до всіляких специфічних потреб того чи іншого рішення, тієї чи іншої організації.

- Простота системи. Слід зупинитися на найбільш затребуваних і необхідних можливостях, інший функціонал нарощувати модулями. Одним з основних параметрів оцінки інтерфейсу користувача є швидкість його реакції, яка в значній мірі залежить від продуктивності сервера, швидкості з'єднання і інших зовнішніх факторів.

- Пошукова оптимізація. Це оптимізація HTML-коду, структури, контенту сайту і зовнішніх чинників з метою підняття його в пошукових системах. Оптимізація і просування сайту є комплексом робіт з підвищення рейтингу сайту в пошукових системах.

- Підтримка продукту. Простота оновлень. Будь-яка система управління має вразливості, і часто адміністратори забувають про оновлення системи управління, що може стати причиною злому сайту і всього сервера. У більшості систем управління автоматичні оновлення здійснюються частково за запитом адміністратора з системи управління.

- Безпека. Сюди входить стійкість до SQL-injection, XSS-

скриптингу, захист від підміни переданих параметрів. Обов'язкова можливість здійснення резервного копіювання та відновлення даних, а також захист від флуду.

– Внутрішня статистика. CMS повинна мати функціонал для отримання статистики про користувача, продаж товарів, активність, тощо.

В результаті аналізу, було виявлено дві системи, які більше інших відповідали критеріям: Opencart та Shopify. Кожна з цих систем має свої переваги і недоліки, а отже, потрібно провести глибше порівняння цих систем.

Розробка інтернет-магазину дозволить вирішити кілька завдань одночасно: вплинути на певну цільову аудиторію, залучити постачальників, почати реалізацію своїх товарів і послуг по всій країні.

Теоретичні результати дослідження можна використовувати у навчальному процесі ЗВО, під час вивчення різних аспектів вебпроекування.

УДК 004.031.42

Сапонько Д. С., Кулаковська І. В.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

ПАКЕТНИЙ МОДУЛЬ ДЛЯ НАЛАШТУВАННЯ АРХІТЕКТУРИ ТА ЛОГІКИ NODE.JS ПРОЄКТУ З ВИКОРИСТАННЯМ МІКРОСЕРВІСНОЇ АРХІТЕКТУРИ

Node.js — платформа з відкритим кодом для виконання високопродуктивних мережевих застосунків, написаних мовою JavaScript. Засновником платформи є Раян Дал (Ryan Dahl) [1]. Якщо раніше Javascript застосовувався для обробки даних в браузері користувача, то node.js надав можливість виконувати JavaScript-скрипти на сервері та відправляти користувачеві результат їхнього виконання. Платформа Node.js (див.рис.1) перетворила JavaScript на мову загального використання з великою спільнотою розробників.

Архітектура - це процес проєкування та створення ІТ-рішень, заснований на принципах та керівних принципах [3]. Але все ж таки у визначенні відсутнє те, що має точно включати чітко певну архітектуру. Архітектура повинна охоплювати та забезпечувати наступне.

– Це схема високого рівня ІТ-компонентів у застосунку, що охоплює розробку програмного та апаратного забезпечення.

– Демонструє основи того, як спеціально розроблені рішення або продукти постачальників розробляються та створюються для інтеграції з існуючими системами та задоволення конкретних вимог.

– Інтегрується в методологію розробки програмного забезпечення для розуміння та розробки специфікацій та моделей ІТ-програмного та апаратного забезпечення відповідно до стандартів, рекомендацій та специфікацій.

– Використовує передовий досвід для заохочення використання (наприклад, «відкритих») технологічних стандартів, глобальної сумісності технологій та існуючих ІТ-платформ (інтеграція, дані тощо).

– Забезпечує послідовний, послідовний та універсальний спосіб показати та обговорити розробку та реалізацію ІТ-можливостей рішення.

Важливу роль під час розробки проекту зіграло вибір архітектури яка буде використовуватись. Було проаналізовано чотири типи архітектури, вивченню документацію та розглянуто переваги і недоліки різних типів архітектури [2].

Вибір архітектури є дуже важливою складовою розробки проекту бо саме від цього залежатиме, за яким стилем має бути написана логіка. Було розглянуто і проведення порівняння переваг та недоліків таких архітектур: багат шарова архітектура, сервіс-орієнтована архітектура (SOA), мікросервісна архітектура.

Багат шарова архітектура. Завдяки цьому типу архітектури , застосунок ділиться на 3 шари. Уявлення, бізнес-логіка, передача даних.

Переваги

- Проста реалізація.
- Слабка зв'язність.
- Незалежність між шарами.

Недоліки

- Має недостатню масштабованість

Сервіс-орієнтована архітектура (SOA) – тип архітектури, який пропонує метод багаторазового використання програмних компонентів з допомогою інтерфейсів сервісів. Ці інтерфейси використовують загальні комунікаційні стандарти, щоб їх можна було швидко включити до нових програм, не вдаючись щоразу до глибокої інтеграції.

Переваги

- Незалежність від обраних технологій.
- Незалежність від особливостей передачі/зв'язку.

Недоліки

- Заблоковані канали зв'язку.
- Складна та неоднозначна специфікація.

Мікросервісна архітектура. Архітектура працює за принципом компонентизації послуг. Вона поділяє програмне забезпечення на різні ізольовані компоненти (сервіси), кожен із яких несе єдину відповідальність.

Переваги

- Пропонує слабку пов'язаність завдяки високому ступеню ізоляції.
- Підвищує модульність.
- Збій в одному сервісі не торкнеться всієї системи, оскільки вони ізольовані.
- Пропонує високу гнучкість та масштабованість.
- Простота модифікації може прискорити ітерацію.
- Дозволяє реалізувати покращену систему обробки помилок.
- Вирішує проблеми з потоками даних, які бувають з багатопотоковою архітектурою.

Недоліки

- Підвищений ризик збою під час обміну даними між сервісами.
- Великою кількістю сервісів важко керувати.

Обрана Node.js є втіленням надзвичайно настроюваної та масштабованої технології (рис.1). Серверний механізм використовує неблокуючу модель вводу-виводу на основі подій.

Після аналізу можливих типів архітектури було обрано «мікросервісну архітектуру» Node.js. Цей тип архітектури має багато важливих переваг, що робить її більш необхідною для розробки прм модуля з великою кількістю компонентів та зв'язків між цими компонентами. Ця архітектура допоможе, у випадку, якщо щось піде не так, захистити увесь застосунок від помилок, а працювати не буде лише той компонент саме у якому була помилка.

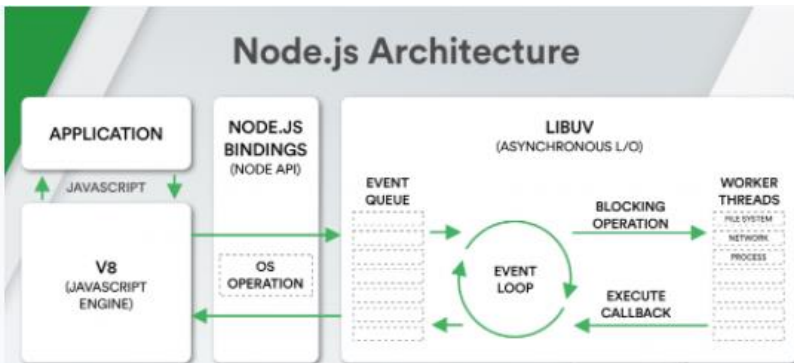


Рисунок 1 – Архітектура платформи Node.js

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. NodeJS. OpenJS Foundation: вебсайт. URL: <https://nodejs.org/en/>
2. Довідкова документація про API надає детальну інформацію про функції та об'єкти у Node.js. OpenJS Foundation: вебсайт. URL: <https://nodejs.org/uk/docs/>
3. Node-v0.x-archive. GitHub: вебсайт. URL: <https://github.com/nodejs/node-v0.x-archive>

УДК 004.42

Сокрут А. А. Воробйова А. І.

*Чорноморський національний університет імені Петра Могили
м. Миколаїв, Україна*

РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ РОЗДРІБНОГО ПРОДАЖУ ТОВАРІВ СУЧАСНОЇ МОДИ

Сучасні технології стимулюють активність електронної комерції. У міру переходу комерції в режим онлайн, на перший план виходять нові рішення та стратегії віддаленого керування і агрегування ринків E-commerce. Особливу активність та популярність набули інтернет-магазини підчас поширення пандемії COVID-19 по всіх континентах.

E-commerce стає віддалено керованою. Більшість компаній інвестують в унікальні, автономні платформи онлайн-продажів. Деякі вважають за краще інвестувати в комплексні платформи або власні розробки. Однак недоліками таких підходів можуть бути повільні терміни виходу на ринок, низька гнучкість каналів продажів і високі витрати на розробку. Інший підхід — архітектура віддалено керованою електронної комерції, яка викликала підвищений інтерес ринку[1].



Рисунок 1 - Архітектура віддалено керованої електронної комерції

Інтернет-магазин - це є найкраще місце, де можна придбати одяг не виходячи з дому. Інтернет-магазин цілодобово доступний для пошуку потрібного товару, наприклад одягу а саме: переглянути його властивості, розміри, матеріали і т.п., замовити його, отримати відгуки інших споживачів цього товару, а також при необхідності отримати підтримку при виборі бажаного розміру товару.

Інтернет-магазин дозволяє користувачам сформувати замовлення на покупку, вибрати спосіб оплати і доставки замовлення в мережі Інтернет.

Технології за допомогою яких можна зробити інтернет-магазин називаються вебтехнології. Це технології, які за допомогою програмного коду створюють в Інтернеті сторінки, на котрих розміщуються сайти. Також використовують шаблони для свого інтернет-магазину, вони зручні та економлять багато часу [2].

Ці шаблони називають CMS, популярними серед них є: PrestaShop, OpenCart, Bitrix, ModX, Magento тощо.

Шаблон має заготовку на інтернет-магазин, тобто він вже має допустимий функціонал для роботи з інтернет-магазином, залишається лише програмно доповнити цей шаблон потрібними клієнту функціями.

Інтернет магазин складається з двох частин: серверної та клієнтської.

Серверна частина – це та частина, яка обробляє запити на стороні сервера та, після успішної обробки, відправляє їх на клієнтську частину, це зумовлено тим, щоб не навантажувати пристрій користувача сайту.

Популярна програмна мова серверної частини PHP та його фреймворки.

Клієнтська частина – це та частина, яка оброблює запити на стороні клієнта, ці запити не повинні загрузати пристрій користувача сайту. На клієнтській частині використовуються такі інструменти як: HTML, CSS, JavaScript [3].

Найважливіша частина розробки інтернет-магазину – це вибір сервера. Сервер – це такий самий комп'ютер, де обробляються запити від клієнта(серверна частина), тільки в рази сильніший. Він повинен завжди працювати, якщо він буде вимикатися, то інтернет-магазин не буде працювати.

Роблячи вибір на користь відкриття інтернет-магазину, компанія переходить на більш високий рівень розвитку свого бізнесу і робить свій асортимент більш доступним для широкого кола покупців.

До позитивних якостей Інтернет-магазину можна віднести:

1. Інтернет-магазин працює 24 години на добу, без будь-яких вихідних, всі 365 днів в році. Користувач може зробити покупку, або замовлення одягу в будь-який зручний час доби;

2. відмінний вибір різного одягу;

3. лояльність до покупця;

4. спрощена навігація по сторінках, категорії одягу і їх підтем в інтернет-магазині дозволить ретельно вибрати товар і замовити його. Кілька актуальних способів доставки товару через Інтернет-магазин не змусить користувача довго чекати оплачену раніш покупку в мережі Інтернет.

5. перебуваючи навіть за містом, можливо відвідати будь-який Інтернет-магазин, не тільки той який знаходиться в межах вашої країни, але і інший Інтернет-магазин, який існує наприклад в США, або іншій європейській країні.

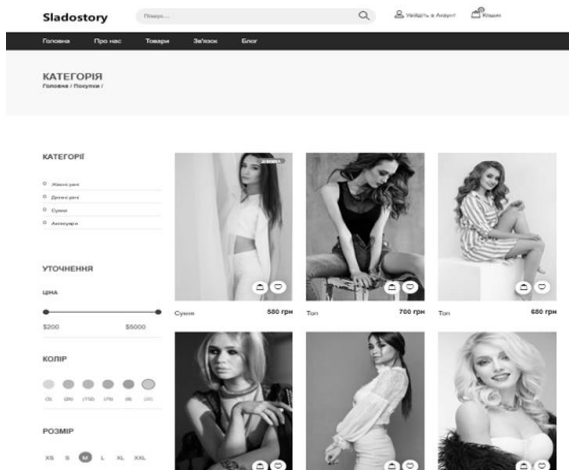


Рисунок 2 – Розроблений інтернет-магазин одягу

Висновки. Покупці мають однакові завдання, мета яких зводиться до отримання інформації про певний одяг, причому, покупки все роблять абсолютно різними способами, а що власники інтернет-магазинів мають однакові завдання, мета яких – продаж. В результаті проектної роботи, буде розроблений вебсайт, що представляє собою інтернет-магазин. Всі вимоги за функціями вебсайту будуть у процесі виконані.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bart Paalman, Patrick Bisceglia. New technology developments stimulate e-commerce investment and growth. URL: <https://www.bdo.global/en-gb/blogs/tech-media-watch-blog/june-2021/new-technology-developments-stimulate-e-commerce-investment-and-growth>
2. Дари К., Вільямс М. PHP и MySQL. Создание Интернет-магазина. 2010. 640 с.
3. Навчальний посібник HTML5. Metanit: вебсайт. URL: <https://metanit.com/web/html5/8.1.php>.

УДК 004.031.42

Третяк В. О., Лисенков Е. А.

*Чорноморський національний університет імені Петра Могили,
м. Миколаїв, Україна*

**ВЕБЗАСТОСУНОК ДЛЯ МЕНЕДЖМЕНТУ РОЗРОБКИ
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПО МЕТОДОЛОГІЇ SCRUM**

Задачею роботи є створення потужної платформи, що дозволить полегшити комунікацію за допомогою інтеграції чату та системи контролю версіями для швидкого перегляду змін за номером коміту, переглядати Canban або Scrum boards, планувати релізи, планувати спринти, мати можливість трекінгу та редагування задач, додавати коментарі, надавати права доступу до матеріалів або складових проекту на рівні менеджменту, можливість створювати звіти та документацію, переглядати беклоги проекту.

Сучасні тенденції та інтенсивність розвитку сфери ІТ створюють нові передумови для стандартів керування проектами та визначенням пріоритетів при розробці програмного забезпечення. Налагодження відносин між командами стає одним із найважливіших підходів у діяльності компанії на шляху досягнення поставлених задач.

Постановкою задачі є створення вебзастосунку для менеджменту програмного забезпечення. Програмне забезпечення, метою якого є бажаний та доведений результат, досягнутий у межах певного строку при заданих умовах реалізації проекту. Визначення мети проекту передбачає постановку задачі, що вимагає:

- Визначити результати діяльності на певний строк;
- Дати цим результатам кількісну оцінку;
- Довести, що ці результати можуть бути досягнуті;
- Визначити умови, за яких ці результати мають бути досягнуті.

Метою дослідження є створення онлайн платформи менеджменту у галузі розробки ПЗ, що зможе надати альтернативу одразу кількома сервісам, поєднавши функціонал, який зазвичай використовуються командами розробників в ІТ компаніях.

Відповідно до мети, виділені наступні завдання магістерської роботи:

1. Аналіз сфери ІТ-компаній. Постановка задачі.
2. Математичні моделі та архітектура для вирішення задачі створення вебзастосунку функціями менеджменту проектів, перегляду коду, та організації часу.
3. Створення макету. Програмна реалізація серверної та клієнтської частини вебзастосунку.

У зв'язку з тим, що платформа, на якій працює команда або окрема під команда, може змінюватися від проекту або вимог клієнта, прийнято рішення створити вебзастосунок та додати підтримку всіх браузерів.

Для виду вебзастосунку вибрано Single Page Application (SPA або одно сторінковий застосунок). Варіант, в якому використовуються бекенд і фронтенд. За допомогою їх взаємодії можна створити застосунок, який буде працювати зовсім без перезавантажень сторінки

в браузері. Або в спрощеному варіанті, коли переходи між розділами викликають перезавантаження, але будь-які дії в розділі обходяться без них. Це дозволить написати велику бізнес логіку, яка буде досить швидко працювати.

Для створення Single Page Application вибрано React. React це бібліотека для створення користувацьких інтерфейсів. Однією з її характерних особливостей є можливість використовувати JSX, мова програмування з близьким до HTML синтаксисом, який компілюється в JavaScript. Розробники можуть вимагати більшої продуктивності застосунків за допомогою Virtual DOM. С React ви можете створювати ізоморфні застосунки, які допоможуть вам позбавитися від неприємної ситуації, коли користувач з нетерпінням чекає, коли ж нарешті завершиться завантаження даних і на екрані його комп'ютера нарешті з'явиться щось крім анімації завантаження. Створені компоненти можуть бути з легкістю використані повторно. Високий відсоток повторного використання коду підвищує покриття тестами, що, в свою чергу, призводить до більш високого рівня контролю якості.

Для створення макету SPA, а саме розділів чату, дошки задач, документації, коментарів, використано програму розробки макетів Figma. Для проектування дизайну використані основні принципи UI/UX. Проаналізовано та створено уявлення бази даних відповідно до необхідних розділів застосунку. Поля та типи даних для кожного з уявлень спроектовано в межах архітектури та макету застосунку, що дозволило легко працювати з створенням серверної частини та запитами між сервером та SPA. Серверна частина створена, а саме запити для отримання, видалення, додавання та редагування повідомлень, кімнат чату, дошки з задачами, коментарів, документів та звітів. Запити перевірені на коректну роботу, та додана перевірка для формату полей.

Також, засобами React, Redux та Thunk програмно реалізовано застосунок та весь функціонал, який поставлено щодо кожного з розділів та окремих компонентів. Архітектура застосунку спроектовано модульно, та дозволяє легко змінювати та розширювати застосунок, якщо виникне така необхідність. Кожна з поставлених цілей розробки виконана, а весь функціонал покритий та перевірений.

Результатом дослідження виявлені основні концепції менеджменту проєктів, інструменти та кроки побудови проєкту від ідеї до створення чату, управління задачами та командою в різних умовах та написанню документації та звітів. Створено чіткий план для того, щоб автоматизувати рутинні задачі менеджменту проєктів та комунікації команди в єдину якісну систему, яка зможе покрити потреби менеджерів, бізнес-аналітиків, розробників. Система допоможе

швидко планувати задачі, спринти, релізи; отримати можливість створити Scrum або Kanban Boards; мати можливість трекінгу задач та створення коментарів; написання звітів та технічної документації; надавати права доступу на рівні проєкту, поліпшити комунікацію та ревью коду за допомогою інтеграції сервісу комунікації та GIT.

Інформаційні технології у навчальному процесі

УДК 519.85

*Дворецька М. М., Воробйова А. І.
Миколаївський муніципальний колегіум імені В. Д. Чайки,
м. Миколаїв, Україна*

ВИЗНАЧЕННЯ ОПТИМАЛЬНОГО РОЗПОДІЛУ ЗНАЧЕНЬ КОЕФІЦІЄНТІВ СКЛАДОВИХ КОНКУРСНОГО БАЛУ АБІТУРІЄНТА

Загальний конкурсний бал складається із результатів ЗНО по декільком предметам, середнього балу атестату, а також балу за мотиваційний лист та особливі успіхи. Всі складові мають свої вагові коефіцієнти, і відповідно, різний ступінь впливу на рішення про вступ абітурієнта до закладу вищої освіти, а також чи потрапить він на бюджетну форму навчання. Вагові коефіцієнти визначаються закладом вищої освіти, та не усюди є оптимальними з точки зору подальшого успішного навчання на тій чи іншій спеціальності.

Підбір оптимальних значень вагових коефіцієнтів, при яких успіхи студентів у навчанні будуть максимально відповідати їх конкурсному балу при вступі, є класичним прикладом задачі оптимізації, для рішення яких можуть бути використані приблизні методи. Вони є методами перебору рішень для тих завдань, в яких неможливо знайти рішення за допомогою математичних формул. Використання одного із них, а саме генетичного алгоритму, покладено в основу даної роботи.

Окремі складові конкурсного балу мають різний ступінь впливу на результати подальшого навчання студента залежно від обраної ним спеціальності. Відповідно при неоптимальному визначенні рівня коефіцієнтів складових конкурсного балу його загальне значення не відповідатиме рівню подальшої успішності студента.

Згідно правил вступу [1], на значення конкурсного балу абітурієнта впливає шість складових – результати з трьох предметів зовнішнього незалежного оцінювання (або двох предметів та творчого конкурсу), середнього балу атестату про загальну середню освіту, балу за мотиваційний лист та балу за особливі успіхи. На факультеті комп’ютерних наук Чорноморського національного університету імені Петра Могили, першими двома предметами є математика та українська мова, а третім є дисципліна за вибором – фізика або іноземна мова (англійська, французька, німецька або іспанська). Кожна шести вищезазначених складових має свій ваговий коефіцієнт, згідно до якого частина предметів має більший вплив на значення конкурсного балу у порівнянні із іншими.

Формула розрахунку конкурсного балу представлена нижче:

$$KB = \sum_{i=1}^3 \text{коэф}_i \times \Pi_i + \text{коэф}_A \times A_{200} + \text{коэф}_{\text{мл}} \times \text{МЛ} + \text{коэф}_{\text{оу}} \times \text{ОУ} \quad (1),$$

де Π_i – результат зовнішнього незалежного оцінювання з i -го предмету у 200 бальній шкалі, коэф_i – ваговий коефіцієнт i -го предмету, A_{200} – середній бал у 200 бальній шкалі, коэф_A – коефіцієнт середнього балу атестату, МЛ – бал за мотиваційний лист у 200 бальній шкалі, $\text{коэф}_{\text{мл}}$ – коефіцієнт мотиваційного листа, ОУ – бал за особливі успіхи, $\text{коэф}_{\text{оу}}$ – коефіцієнт балу за особливі успіхи [2].

Середньозважений бал за результатами сесії розраховується на основі балів з дисциплін, які отримав студент протягом сесії. Він є показником успішності навчання та рівня засвоєння матеріалу. У розрахунку середньозваженого балу також приймають участь вагові коефіцієнти дисциплін, що залежать від обсягу дисципліни у кредитах. Формула розрахунку середньозваженого балу має наступний вигляд (2.3):

$$PB = \sum_{j=1}^N \text{коэф}_j \times \text{Бал}_j \quad (2),$$

де N – кількість дисциплін, що викладалися протягом семестру; Бал_j – оцінка, отримана студентом з j -ї дисципліни; коэф_j – ваговий коефіцієнт j -ї дисципліни

Конкурсний бал абітурієнта представлено у 200 бальній шкалі, а середньозважений бал результату сесії у 100 бальній. Також слід враховувати, що область значень результату ЗНО – від 100 до 200 балів, а для оцінки з дисципліни, що отримує студент протягом сесії – від 60 до 100 балів. Для приведення конкурсного та рейтингового балів до однієї шкали, використано формулу природної нормалізації. Враховуючи, що для конкурсного балу абітурієнта $\min(KB) = 100$ та $\max(KB) = 200$, формула набуває наступного вигляду:

$$KB^{норм} = KB/100 - 1 \quad (3)$$

Для рейтингового балу студента $min(PB) = 60$ та $max(PB) = 100$, відповідно до чого:

$$PB^{норм} = PB/40 - 1,5 \quad (4)$$

Метою роботи є знаходження оптимальних значень вагових коефіцієнтів конкурсного балу при яких його значення буде максимально відповідати подальшим успіхам у навчанні студента, тобто значенню рейтингового балу за результатами сесії. Для k -го студенту маємо конкурсний бал $KB_k^{норм}$, розрахований за (3), та рейтинговий бал за результатами сесії $PB_k^{норм}$, отриманий згідно (4). В ідеалі різниця між ними дорівнює нулю.

Враховуючи наявність результатів сесії та ЗНО для m студентів, маємо систему з $m+1$ лінійних рівнянь, яка доповнюється обмеженням, що сума всіх вагових коефіцієнтів має дорівнювати одиниці. Але така система не матиме рішення, оскільки на практиці зменшення різниці конкурсного та рейтингового балу для одного студента може вести до її збільшення для іншого. Тому, рівність може бути доповнена дельтою, або різницею між конкурсним та рейтинговим балами. Також до системи рівнянь, згідно до умов вступу [1], додаються обмеження на область значень коефіцієнтів та умову щодо того, що сума коефіцієнтів дорівнює одиниці. Відповідно до цього, система набуває наступного вигляду (5) [2]:

$$\left\{ \begin{array}{l} PB_1^{норм} + 1 - (\sum_{i=1}^3 коеф_i \times \Pi_{i,1} + коеф_A \times A_{200,1} + коеф_{мл} \times ML_1 + коеф_{oy} \times OY_1) / 100 = \Delta_1 \\ \dots \\ PB_m^{норм} + 1 - (\sum_{i=1}^3 коеф_i \times \Pi_{i,m} + коеф_A \times A_{200,m} + коеф_{мл} \times ML_m + коеф_{oy} \times OY_m) / 100 = \Delta_m \\ \sum_{i=1}^3 коеф_i + коеф_A + коеф_{мл} + коеф_{oy} = 1 \\ 0,2 \leq коеф_1 < 1, \quad 0,2 \leq коеф_2 < 1, \quad 0,2 \leq коеф_3 < 1 \\ 0 \leq коеф_A \leq 0,1, \quad 0 \leq коеф_{мл} \leq 0,05, \quad 0 \leq коеф_{oy} \leq 0,01 \end{array} \right. \quad (5)$$

Така система має нескінченну кількість розв'язків, область значень яких обмежена 6-ма додатковими нерівностями, що визначають область значень вагових коефіцієнтів складових конкурсного балу. Критерієм оптимальності рішення виступає мінімізація середнього відхилення рейтингового та конкурсного балів по всім студентам [2], а функція мети для системи (5) представлена наступним чином:

$$f(коеф_1, коеф_2, коеф_3, коеф_A, коеф_{мл}, коеф_{oy}) = \sum_{i=1}^m |\Delta_k| / m \rightarrow 0 \quad (6)$$

Задачу вирішено із використанням модифікованого генетичного алгоритму, що відноситься до групи приблизних методів рішення.

Отримані в результаті проведення декількох спроб розрахунків хоч і за різну кількість кроків, але однакові значення коефіцієнтів складових конкурсного балу, свідчать про правильність роботи алгоритму та, відповідно, про достовірність результатів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Умови прийому для здобуття вищої освіти 2021 року. URL: <https://mon.gov.ua/ua/osvita/visha-osvita/vstupna-kampaniya-2021/umovi-prijomu-dlya-zdobuttya-vishoyi-osviti-2021-roku> (дата звернення: 26.11.2021)

2. Дворецька М. М. Використання генетичного алгоритму для розрахунку значень вагових коефіцієнтів складових конкурсного балу абітурієнта. URL: https://jasu2021.com/image_upload/ri_2.12.pdf (дата звернення: 16.11.2021)

*Плотніков М. С., Рудніченко М. Д.
Національний університет «Одеська політехніка»,
м. Одеса, Україна*

ІНТЕРАКТИВНА СИСТЕМА НАВЧАННЯ ДЛЯ ЗРУЧНОЇ ВІЗУАЛІЗАЦІЇ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

Щодня людина поглинає величезну кількість інформації. Старі методи та способи їх передачі починають старіти і виникає потреба у пошуку зручного та швидкого способу для навчання певної тематики.

Для забезпечення належного рівня освіти та залучення учнів необхідно використовувати сучасні технології мультимедії [1]. Такими засобами можуть бути мультимедійні дошки, використання різних планшетів, віртуальних шоломів та технології доповненої реальності.

Завдяки використанню таких технологій можна підняти зацікавленість учнів, та й залучити людей іншого віку. Інтерактивність дозволяє так само відчувати наближеною той чи інший елемент. Так можна за допомогою шолома доповненої реальності показати хімічну реакцію без взаємодії того, хто навчається з можливим небезпечним процесом. Також можна продемонструвати і процеси, що не досягаються.

Але у використанні кожної технології існують свої плюси та мінуси. Основним головним плюсом перерахованих вище технологій є їх наявність та цілеспрямована покупка для шкіл та інших навчальних закладів. А для використання шолома і повного занурення необхідний спеціальний простір [2].

Завдяки використанню технології доповненої реальності можна легко вирішити цю проблему. Для її використання учневі потрібно лише власник якимось пристроєм з камерою. Навівши на певний елемент, можна доповнити реальність за допомогою камери телефону або просканувати вже існуючий елемент для його вивчення [3].



Рисунок 1 – Демонстрація роботи електричного кола з використанням технології доповненої реальності

Також проблема набирає популярності у зв'язку з дистанційним навчанням та пандемією у світі. Більшість шкіл закрито на карантин, а проводити досліди та експерименти можливості немає. Тому доповнена реальність для використання її в домашніх умовах допомагає вирішити також поставлене завдання.

Інтерактивна система навчання з використанням доповненої реальності повинна допомогти вирішити ряди вище перерахованих проблем. Система повинна містити дуже гнучкий та зручний інтерфейс для користувача. Хороший та грамотний застосунок у використанні може забезпечити швидку та якісну працездатність.

З боку вчителів система повинна уявити невелику вебсторінку, на якій вчитель з простотою зміг би реалізувати або розширити поставлене завдання з використанням технології доповненої реальності. З себе ця вебсторінка повинна надавати зручний конструктор у стилі програми Blender або Photoshop, але не повинна мати перевантажений інтерфейс. Вчитель з простотою повинен завантажити або додати модельки, можливо зробити невелику пошукову систему з використанням якогось ресурсу за 3D моделями.

Інтерактивна модель також сприяє гарному запам'ятовуванню користувача. Проста демонстрація – це не завжди найкращий спосіб навчання. Якщо користувачеві надати можливість самим впливати на процес і поставити завдання, яке йому потрібно досягти, то з певною ймовірністю це запам'ятися у нього.

Таким чином інтерактивна система з використанням технології доповненої реальності може вирішити проблеми сучасного навчання та покращити якість отриманих знань.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Зайцев В.С. Мультимедійні технології в освіті: сучасний дискурс: Видавництво ЗАТ «бібліотека а. Міллера», 2018. 30 с.
2. Доповнена реальність в освіті. Віртуальні окуляри. URL: <https://virtualnyeochki.ru/stati/dopolnennaya-realnost-v-obrazovanii>. (дата звернення: 23.12.2021)
3. Плотніков М.С., Рудніченко М. Д., Граб К.В. Інтерактивна система вивчення астрономії з використанням технології доповненої реальності: Вісімнадцята всеукраїнська конференція студентів і молодих науковців : Південноукраїнський національний педагогічний університет імені к. д. Ушинського, 2021. 115-117 с.

ДЛЯ ПОДАТОК

Тези подано в авторській редакції.

Підп. до друку 7.02.2022.
Формат 60 × 84¹/16.
Гарнітура «Times New Roman».
Ум. друк. арк. 6,28. Обл.-вид. арк. 5,05.