

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти – другий (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 126 Інформаційні системи та технології
Освітньо-професійна програма – Інформаційні технології аналізу даних та зображень

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

 д.т.н., проф. Віталій МОКІН

«25» 09 2025 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
Караваєву Вадиму Олександровичу

1. Тема роботи: “Інформаційна технологія комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг”,

керівник роботи: Ігор ШТЕЛЬМАХ, к.т.н., ас. каф. САІТ,
затверджені наказом ВНТУ від «24» 09 2025 року № 313

2. Строк подання студентом роботи «28» 11 2025 року

3. Вихідні дані до роботи:

Моніторинг та екологічна оцінка водних ресурсів України
(<http://monitoring.davr.gov.ua/EcoWaterMon/GDKMap/Index>).

4. Зміст текстової частини:

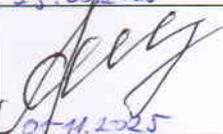
- загальна характеристика поставленої задачі;
- розроблення та впровадження інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг;
- побудова та оцінювання моделей нейронних мереж для прогнозування якості поверхневих вод;
- економічна частина.

5. Перелік ілюстративного матеріалу:

- обрані оптимальні інструменти і технології;
- підготовлені дані для геопросторового аналізу;
- інтерактивна карта з розташуванням постів;
- тематичні карти;
- функціональна схема роботи ШІ-агента;

- графіки отримані під час розвідувального аналізу;
- таблиця навчання моделей;
- графіками з результатами прогнозування.

6. Консультанти розділів МКР

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
2	Крижановський Є.М., к.т.н., доцент каф. САІТ	 25.09.2025	 05.10.2025
4	Олександр ЛЕСЬКО, к. е. н., проф. каф. ЕПВМ	 05.11.2025	 25.11.2025

7. Дата видачі завдання «25» 09 2025 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Загальна характеристика поставленої задачі	15.09.2025	25.09.2025	Викоріано
2	Розроблення та впровадження інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг	25.09.2025	05.10.2025	Викоріано
3	Побудова та оцінювання моделей нейронних мереж для прогнозування якості поверхневих вод	05.10.2025	25.11.2025	Викоріано
4	Удосконалення інформаційної технології	25.10.2025	05.11.2025	Викоріано
5	Економічна частина	05.11.2025	15.11.2025	Викоріано
6	Оформлення матеріалів до захисту МКР	15.11.2025	25.11.2025	Викоріано

Студент



Вадим КАРАВАЄВ

Керівник роботи



Ігор ШТЕЛЬМАХ

АНОТАЦІЯ

УДК 004.94:556.51:528.9(477.44/.46)

Караваєв В. О. Інформаційна технологія комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг. Магістерська кваліфікаційна робота зі спеціальності 126 – інформаційні системи та технології, освітньо-професійна програма – інформаційні технології аналізу даних та зображень. Вінниця: ВНТУ, 2025. 121 с.

На укр. мові. Бібліогр.: 20 назв; рис.: 48; табл.: 5.

У магістерській роботі проведено комплексний аналіз стану водних масивів басейну річки Південний Буг на основі відкритих даних моніторингу з використанням геоінформаційних технологій, нейронних мереж і великих мовних моделей. На основі виявлених закономірностей змін показників якості води обґрунтовано параметри нейронної мережі та відібрано оптимальну конфігурацію моделі для прогнозування екологічного стану. Об'єктом дослідження є процес моніторингу та аналізу якості поверхневих вод басейну річки Південний Буг.

Ілюстративна частина складається з 11 плакатів із результатами аналізу.

У розділі економічної частини розглянуто питання про доцільність розробки та впровадження інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг.

Ключові слова: геоінформаційні системи, моніторинг вод, геопросторовий аналіз, великі мовні моделі, нейронні мережі, адаптивний парсинг, аналіз даних.

ABSTRACT

Karavaev V. O. Information technology for comprehensive analysis of geospatial relationships and forecasting the state of water bodies in the Southern Bug River basin. Master's thesis in specialty 126 - information systems and technologies, educational and professional program - information technology data and image analysis. Vinnytsia: VNTU, 2025. 121 p.

In Ukrainian language. Bibliogr .: 20 titles; fig .: 48; table: 5.

The master's thesis provides a comprehensive analysis of the state of water bodies in the Southern Bug River basin based on open monitoring data using geoinformation technologies, neural networks, and large language models. Based on the identified patterns of changes in water quality indicators, the parameters of the neural network are justified and the optimal model configuration for predicting the ecological state is selected. The object of the study is the process of monitoring and analyzing the quality of surface waters in the Southern Bug River basin.

The illustrative part consists of 11 posters with the results of analysis.

In the section of the economic part the question of expediency of development and introduction of information technology for complex analysis of geospatial connections and forecasting the state of water bodies of the Southern Bug River basin.

ЗМІСТ

ВСТУП.....	4
1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ПОСТАВЛЕНОЇ ЗАДАЧІ.....	6
1.1 Аналіз предметної області	6
1.2 Огляд аналогів.....	9
1.3 Сучасний стан моніторингу басейну річки Південний Буг.....	12
1.4 Особливості прогнозування рівня забрудненості	18
1.5 Висновки.....	19
2 РОЗРОБЛЕННЯ ТА ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ АНАЛІЗУ ДАНИХ МОНІТОРИНГУ ЯКОСТІ ПОВЕРХНЕВИХ ВОД	20
2.1.1 GeoPandas	20
2.1 Вибір оптимальних інформаційних технологій	21
2.1.2 FastAPI та Flask.....	27
2.1.3 LangChain	28
2.1.4 Kaggle.....	30
2.2 Аналіз геопросторових зв'язків	32
2.3 Автоматизований збір, обробка та структуризація даних моніторингу з просторовою прив'язкою.....	46
2.4 Інтерактивна аналітика даних моніторингу з використанням ШІ.....	49
2.5 Розвідувальний аналіз і підготовка даних для нейронної мережі	55
2.6 Висновки.....	63
3 ПОБУДОВА ТА ОЦІНЮВАННЯ МОДЕЛЕЙ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПРОГНОЗУВАННЯ ЯКОСТІ ПОВЕРХНЕВИХ ВОД.....	64
3.1 Ідентифікація оптимальних параметрів нейронної мережі.....	64
3.2 Підготовка даних для нейронної мережі.....	68
3.3 Висновки.....	73
4 ЕКОНОМІЧНА ЧАСТИНА.....	74
4.1 Оцінювання комерційного потенціалу розробки	74
4.2 Прогнозування витрат на виконання науково-дослідної роботи.....	77
4.3 Розрахунок економічної ефективності науково-технічної розробки	83
4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності .	84
4.5 Висновки.....	87
ВИСНОВКИ.....	88

	3
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	90
Додаток А (обов'язковий). Технічне завдання	93
Додаток Б (обов'язковий). Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень	96
Додаток В (довідковий). Лістинг програми	97
Додаток Г (обов'язковий). Ілюстративна частина.....	107

ВСТУП

Актуальність теми. Сучасне управління водними ресурсами потребує комплексного аналізу стану водних масивів, оскільки забруднення негативно впливає на довкілля та здоров'я людини. Використання сучасних інформаційних технологій, зокрема адаптивного парсингу, ГІС і великих мовних моделей, дозволяє інтегрувати дані, аналізувати геопросторові зв'язки та прогнозувати зміни якості води, що підвищує ефективність управлінських рішень і зменшує антропогенний вплив на басейн річки Південний Буг.

Мета і завдання роботи. Метою роботи є підвищення швидкості проведення комплексного аналізу геопросторових зв'язків та збільшення точності прогнозування стану масивів басейну річки Південний Буг шляхом використання методів штучного інтелекту та геоінформаційних технологій.

Розроблення інформаційної технології комплексного аналізу геопросторових зв'язків і прогнозування стану водних масивів басейну річки Південний Буг передбачає виконання наступних задач:

- автоматизувати отримання даних та розробити інформаційну систему для їх подальшого аналізу;
- здійснити візуалізацію геопросторових зв'язків;
- застосувати ШІ для обробки природномовних запитів;
- підібрати оптимальні ІТ-рішення та моделі машинного навчання;

Об'єктом дослідження магістерської кваліфікаційної роботи є процес комплексного аналізу геопросторових зв'язків прогнозування стану водних масивів басейну річки Південний Буг із використанням геоінформаційних систем і мовних моделей.

Предметом дослідження магістерської кваліфікаційної роботи є методи та засоби комплексного аналізу геопросторових зв'язків прогнозування стану водних масивів басейну річки Південний Буг.

Методи дослідження. У дослідженнях використовувались методи ГІС-технологій для формалізації просторових даних ГІС, адаптивний парсинг

даних моніторингу, аналіз відкритих джерел інформації, застосування геоінформаційних систем для просторової візуалізації, використання великих мовних моделей для обробки природомовних запитів, а також побудову та навчання нейронних мереж для прогнозування екологічних показників водних масивів у середовищі Kaggle, використовуючи мову програмування Python.

Новизна одержаних результатів. Полягає у створенні інформаційної технології, що дістала подальший розвиток та об'єднує ГІС, великі мовні й інтелектуальні моделі для комплексного аналізу геопросторових зв'язків водних масивів і прогнозування якості поверхневих вод. Запропонований підхід забезпечує виявлення закономірностей змін якості води та підвищує точність прогнозування екологічних показників басейну річки Південний Буг.

Практичне значення роботи дає змогу оперативно виявляти екологічні ризики, підвищувати точність оцінки якості води та підтримувати ухвалення ефективних управлінських рішень у сфері водних ресурсів.

Апробація результатів магістерської кваліфікаційної роботи. Результати роботи апробовані на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного технічного університету (2024-2025).

Публікації результатів магістерської кваліфікаційної роботи. Опубліковано тези на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного технічного університету (2024-2025) [1], а також опубліковано дві фахових статті у «Вісник Вінницького політехнічного інституту» (2023-2025) [2-3]. Робота була представлена на Всеукраїнському конкурсі студентських наукових робіт «Інформаційні технології» за спеціальністю «Інформаційні системи та технології» і за результатами конкурсу отримано диплом 3 ступеня.

1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ПОСТАВЛЕНОЇ ЗАДАЧІ

1.1 Аналіз предметної області

Басейн річки Південний Буг є одним із найбільших і найважливіших річкових басейнів України, що відіграє значну роль у формуванні водного, екологічного та господарського балансу центральної та південної частин країни. Його площа становить близько 63,7 тис. км², і він охоплює території кількох адміністративних областей — зокрема Хмельницької, Вінницької, Кіровоградської, Миколаївської та Одеської, що забезпечує різноманітність природно-кліматичних умов та типів водокористування (рис. 1.1).

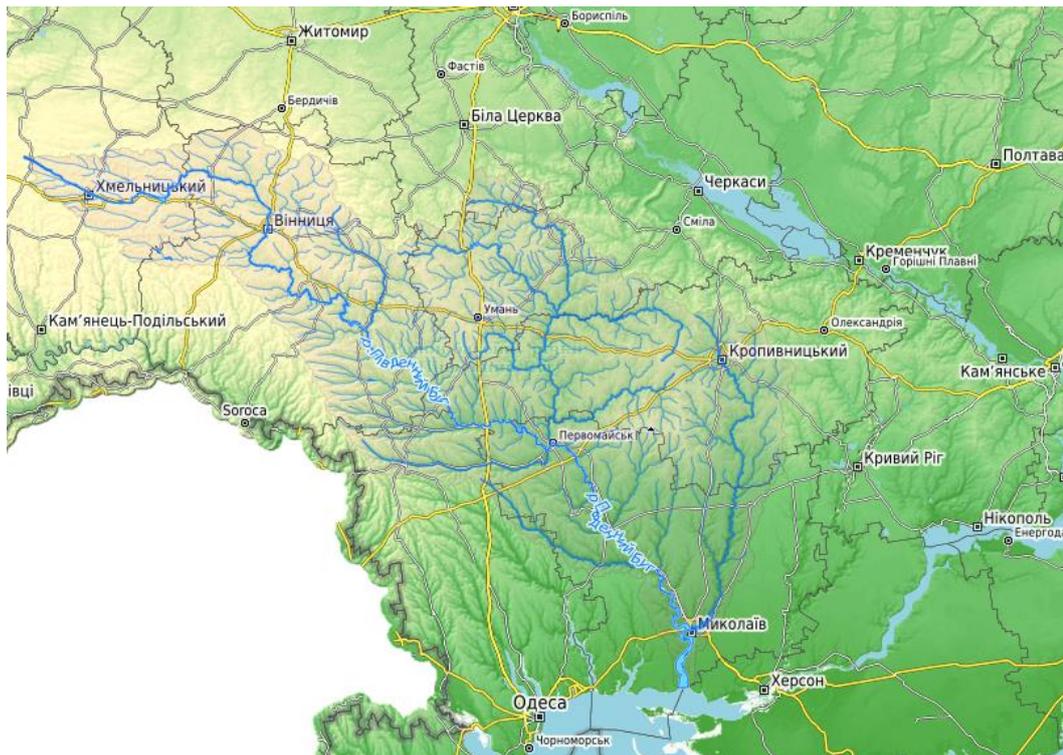


Рисунок 1.1 – Річка Південний Буг

Головною водною артерією басейну є річка Південний Буг, яка бере свій початок поблизу села Холодець Хмельницької області та впадає у Бузький лиман Чорного моря. Протяжність річки перевищує 800 км, а її гідрографічна

мережа є надзвичайно розгалуженою. До неї входять численні притоки, серед яких найбільшими є Снивода, Рів, Соб, Удич, Буг, Кодима, Інгул і Мертвовод. У межах басейну налічується понад 50 водосховищ різного призначення та близько 5000 ставків, що активно використовуються для зрошення, рибництва та регулювання стоку.

Водні ресурси Південного Бугу мають високе господарське значення. Вони забезпечують потреби промисловості, сільського господарства, комунального водопостачання, енергетики та рекреаційного використання. Зокрема, на річці діють ГЕС та ГАЕС, які відіграють важливу роль у стабілізації енергосистеми півдня України. Водночас водойми використовуються для зрошення сільськогосподарських угідь і рекреаційних потреб населення, що робить цей басейн одним із найінтенсивніше експлуатованих у країні.

Проте поряд із значною економічною цінністю, басейн Південного Бугу зазнає потужного антропогенного впливу. Основними чинниками деградації є скидання промислових і побутових стічних вод, забруднення важкими металами, фенолами, нафтопродуктами та іншими токсичними речовинами. Вагомий внесок у забруднення вносить і сільське господарство, зокрема надмірне використання мінеральних добрив та пестицидів, що призводить до евтрофікації водойм і зниження їхньої біологічної продуктивності [4].

Крім того, регулювання стоку через створення численних водосховищ і ставків змінює природний гідрологічний режим, впливає на самоочисну здатність річки, процеси замулення та якість води. Це, у свою чергу, позначається на стані водних екосистем, зокрема на чисельності іхтіофауни та біорізноманітті прибережних зон. Тому дослідження стану басейну Південного Бугу та розроблення сучасних методів моніторингу й управління його водними ресурсами має важливе екологічне, економічне та стратегічне значення для забезпечення сталого розвитку регіону.

Для ефективного та науково обґрунтованого управління водними ресурсами в сучасних умовах необхідно створити єдину інтегровану

інформаційну систему, здатну забезпечити повний цикл роботи з даними про стан водних об'єктів — від збору до аналізу, зберігання, моделювання й візуалізації. Така система повинна стати основою для ухвалення рішень на державному, регіональному та локальному рівнях, оскільки лише комплексний підхід до управління водними басейнами може гарантувати їхнє раціональне використання та відновлення.

Використання геоінформаційних технологій (ГІС) відіграє ключову роль у цьому процесі. Завдяки можливості поєднувати просторові та аналітичні дані ГІС дозволяє створювати інтерактивні карти, моделі розподілу забруднень, відстежувати динаміку екологічних змін і визначати найбільш уразливі зони. Це забезпечує наукове підґрунтя для планування водоохоронних заходів і підвищує обґрунтованість управлінських рішень. Крім того, геоінформаційні системи сприяють підвищенню прозорості управління, оскільки дозволяють відкрито представляти екологічні дані для громадськості, науковців і органів влади.

Як зазначено у статті [5], ефективне управління водними ресурсами неможливе без системного збору, класифікації та стандартизації даних про стан водних масивів. Ці відомості є ключовою основою для формування планів управління водними басейнами та розробки заходів із покращення якості води відповідно до вимог Водної рамкової директиви ЄС [6] та Водного кодексу України [7]. Планування управління передбачає кілька взаємопов'язаних етапів: оцінку екологічного стану водних масивів, ідентифікацію джерел забруднення, розробку та реалізацію природоохоронних заходів, а також регулярний моніторинг результатів їх виконання [8].

Особливе значення має застосування автоматизованих систем моніторингу, оснащених сучасними сенсорами та засобами передачі даних. Такі системи забезпечують збір інформації в режимі реального часу про фізико-хімічні параметри води, зокрема температуру, рівень рН, концентрацію розчиненого кисню, електропровідність і ступінь забруднення органічними чи неорганічними речовинами. Паралельно проводяться лабораторні аналізи, які

уточнюють хімічний склад води, визначаючи наявність важких металів, пестицидів, нафтопродуктів, токсинів та інших небезпечних компонентів.

Поєднання автоматизованого моніторингу з лабораторними дослідженнями забезпечує створення надійної бази даних, необхідної для розробки екологічних стратегій, моделювання динаміки водних процесів та прогнозування можливих наслідків антропогенного впливу. У перспективі такі системи можуть стати складовою інтелектуальної екологічної інфраструктури, здатної автоматично аналізувати тенденції, формувати звіти й рекомендації для оптимізації управління водними ресурсами.

1.2 Огляд аналогів

Одним із аналогів є «Європейська система WISE (Water Information System for Europe)» на (рис. 1.2).

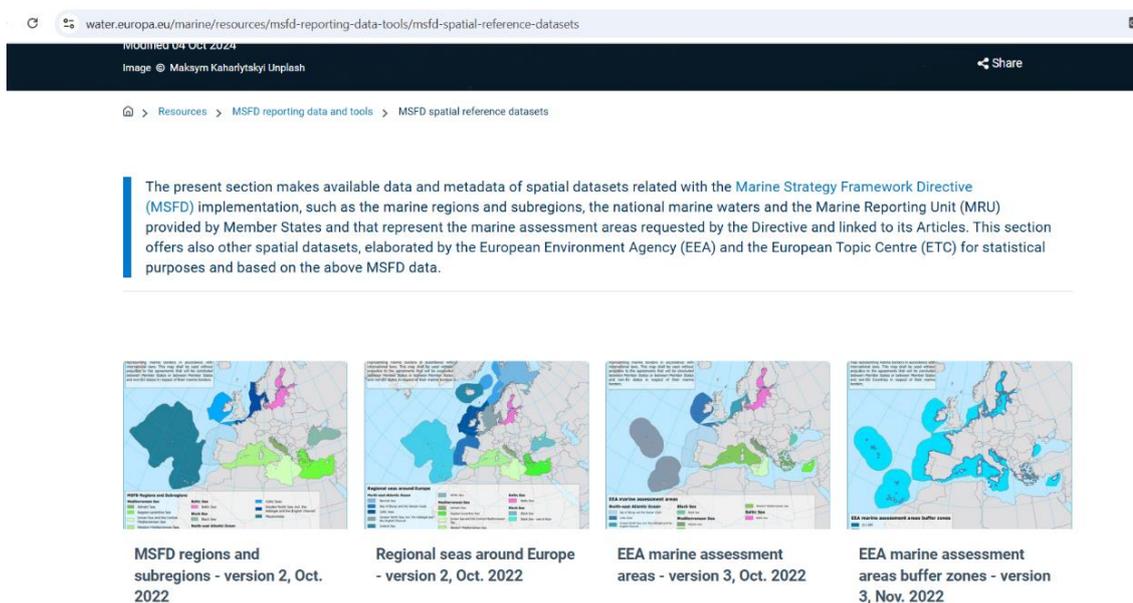


Рисунок 1.2 – Веб-ресурс «Європейська система WISE (Water Information System for Europe)»

Європейська система WISE (Water Information System for Europe) — це масштабна інтегрована інформаційна платформа, створена для збору,

обробки, узагальнення та аналізу даних про водні ресурси на території країн Європейського Союзу. Її метою є забезпечення єдиного доступу до достовірної інформації про стан водних об'єктів, водокористування та водну політику ЄС. Система функціонує як результат співпраці між Європейською комісією, Європейським агентством з навколишнього середовища (ЕЕА), Євростатом та Об'єднаним дослідницьким центром (JRC), що дозволяє об'єднати зусилля різних установ для формування комплексного підходу до управління водними ресурсами.

У межах WISE акумулюються відомості про широке коло параметрів: якість поверхневих та підземних вод, рівень забруднення, стан водних екосистем, гідрологічні характеристики, а також інформація про правові та адміністративні заходи, спрямовані на охорону і відновлення водних об'єктів. Окрему увагу приділено даним щодо реалізації Рамкової водної директиви ЄС (2000/60/EC), яка визначає основні принципи сталого управління водними ресурсами. Таким чином, WISE слугує не лише аналітичним інструментом, а й важливим елементом моніторингу виконання екологічного законодавства в країнах-членах ЄС [9].

Водночас варто підкреслити, що система WISE орієнтована насамперед на стратегічний рівень управління — тобто забезпечення загальноєвропейського бачення стану водних ресурсів та підтримку прийняття рішень на рівні політики ЄС. Її дані агреговані за великими басейнами або адміністративними одиницями, тому не призначені для точного локального моніторингу окремих водойм, малих річок чи ставків. Для таких завдань необхідні національні або регіональні системи моніторингу, які можуть доповнювати WISE, забезпечуючи більш детальну, оперативну і територіально прив'язану інформацію.

Іншим аналогом є веб-система «Моніторинг та екологічна оцінка водних ресурсів України» [10]. Аналогічна система до описаної вище. Джерелом даних є веб-сайт (рис. 1.3).

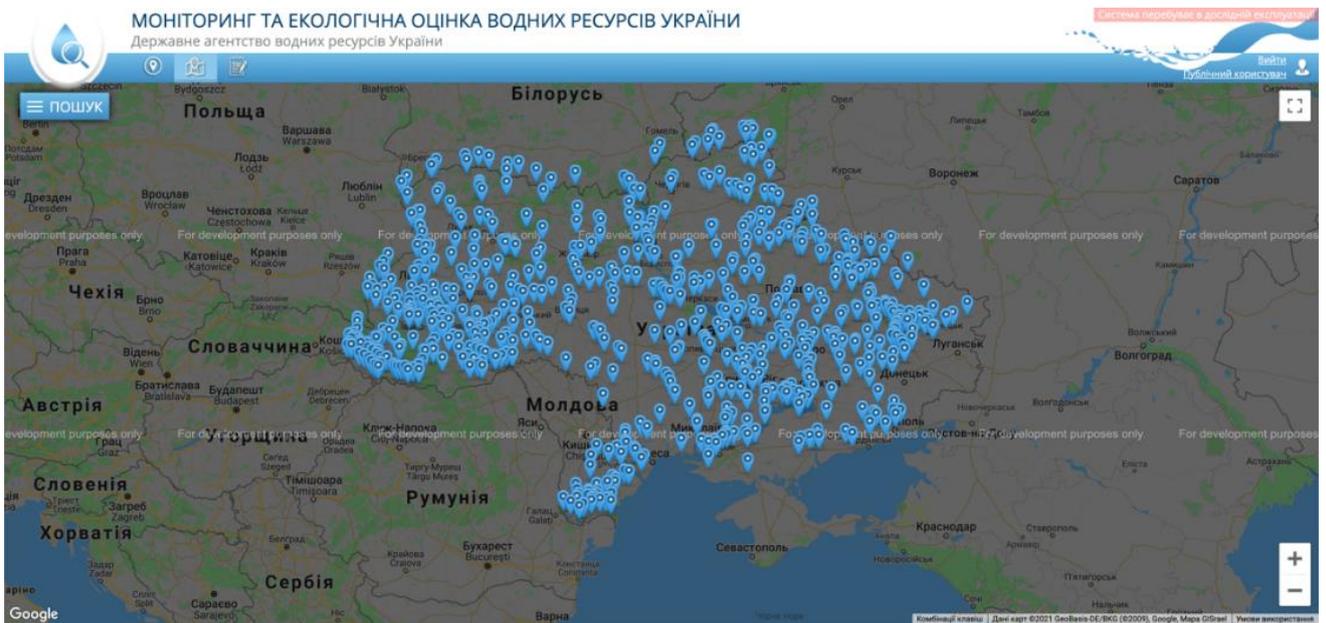


Рисунок 1.3 – Інтерфейс веб-системи «Моніторинг та екологічна оцінка водних ресурсів України»

Попри значні переваги та широкий функціонал, веб-система WISE має й певні недоліки, що обмежують її ефективність у практичному використанні. Однією з ключових проблем є нестабільна робота інтерактивної карти, яка є важливим елементом візуалізації просторових даних. Через технічні збої або неповну інтеграцію геоінформаційних шарів відображення інформації може бути частково некоректним або неповним, що ускладнює проведення просторового аналізу та візуального оцінювання стану водних ресурсів у певних регіонах.

Ще одним обмеженням є нерівномірне наповнення бази даних системи. Для окремих територій Європейського Союзу інформація представлена детально та систематично, тоді як для інших регіонів — зокрема для Вінницької області та інших частин України, які наразі лише частково інтегровані до загальноєвропейських екологічних систем, — кількість доступних відомостей є обмеженою. Це створює труднощі для проведення глибокого та об'єктивного аналізу стану водних екосистем, оскільки відсутність повних даних унеможливорює створення точних моделей чи прогнозів щодо динаміки змін водного середовища.

У зв'язку з цим виникає необхідність удосконалення системи за рахунок інтеграції сучасних інтелектуальних технологій. Одним із перспективних напрямів розвитку є впровадження штучного інтелекту (ШІ) для обробки природномовних запитів користувачів. Це дозволить значно спростити роботу з великими обсягами екологічних даних, зробивши доступ до інформації більш інтуїтивним і персоналізованим. Використання моделей машинного навчання дасть змогу автоматично виявляти закономірності у водних системах, здійснювати прогнозування екологічних процесів і підвищити точність оцінювання стану водойм навіть у регіонах із частковими або фрагментарними даними.

Таким чином, поєднання технологічної основи системи WISE з інструментами штучного інтелекту створить передумови для побудови нового покоління екологічних ІТ-рішень, здатних забезпечити більш повне, гнучке й достовірне моделювання стану поверхневих вод, а також сприяти реалізації принципів сталого управління природними ресурсами на регіональному рівні.

1.3 Сучасний стан моніторингу басейну річки Південний Буг

Моніторинг якості вод у басейні річки Південний Буг є однією з ключових складових державної екологічної політики, спрямованої на забезпечення раціонального використання водних ресурсів, збереження екосистем та охорону здоров'я населення. У сучасних умовах зростання антропогенного навантаження, кліматичних змін та інтенсифікації господарської діяльності питання комплексного оцінювання стану водних масивів набуває особливої актуальності. Саме тому у 2025–2030 роках реалізується План управління річковим басейном Південного Бугу, розроблений та затверджений Державним агентством водних ресурсів України [11].

Зазначений документ є стратегічною основою для організації системного управління водними ресурсами в межах басейну та містить

детальний перелік природоохоронних заходів, спрямованих на покращення екологічного стану поверхневих і підземних вод. У ньому наведено карти масивів поверхневих вод, характеристику їх гідрологічних та екологічних параметрів, а також результати стратегічної екологічної оцінки, яка враховує можливі ризики та наслідки впровадження запланованих заходів.

Згідно з Проєктом ПУРБ, у басейні Південного Бугу ідентифіковано кілька категорій масивів поверхневих вод, серед яких річкові масиви, штучні або істотно змінені водойми, а також перехідні води. Для класифікації річкових масивів застосовано європейський підхід відповідно до стандартів ABRD ЄС, що передбачає типологізацію з урахуванням природного походження водойм, кліматичних умов, географічного розташування та екологічного регіону. У результаті в межах басейну було визначено 15 типів річкових масивів вод, що відрізняються за гідрологічними та геоморфологічними характеристиками [12].

Водозбір басейну охоплює водотоки різного масштабу — від малих річок до великих і дуже великих водних об'єктів. При цьому враховуються площа водозбору, швидкість течії, глибина, структура русла, а також геологічні особливості території, представлені як вапняковими, так і силікатними породами. Таке різноманіття природних умов обумовлює складну динаміку гідрологічного режиму та потребує диференційованого підходу до організації моніторингу й управління.

Басейн Південного Бугу характеризується значною зарегульованістю, що проявляється у великій кількості гідротехнічних споруд, водосховищ і ставків. Сукупний об'єм штучних водойм у маловодні роки перевищує природний стік річки, що істотно впливає на сезонний розподіл вод, процеси самоочищення та якісні показники води. Додатковим негативним чинником є високий рівень ерозійного стоку, зумовлений особливостями рельєфу та інтенсивним сільськогосподарським використанням земель. Значні площі орних угідь сприяють надходженню у водойми зважених речовин, поживних елементів та агрохімікатів, що погіршує екологічний стан водних об'єктів.

У Плані управління басейном передбачено систематичну оцінку хімічного та екологічного стану поверхневих вод шляхом контролю концентрацій пріоритетних забруднювальних речовин (рис. 1.4).

Показник	Відмінна	Добра	Задовільна	Погана	Дуже погана
NH ₄ (амоній), мг/л	≤0.2	0.21–0.5	0.51–1.0	1.01–2.0	>2.0
NO ₂ (нітри), мг/л	≤0.01	0.011–0.03	0.031–0.1	0.101–0.5	>0.5
NO ₃ (нітрати), мг/л	≤1	1–10	11–25	26–50	>50
PO ₄ (фосфати), мг/л	≤0.05	0.051–0.1	0.11–0.2	0.21–0.5	>0.5
БСК ₅ , мг O ₂ /л	≤3	3.1–5	5.1–7	7.1–10	>10
Завислі речовини, мг/л	≤10	11–25	26–50	51–100	>100
O ₂ (розчинений кисень), % насичення	>90%	75–90%	60–75%	40–60%	<40%
Cl ⁻ (хлориди), мг/л	≤25	26–50	51–150	151–300	>300
SO ₄ ²⁻ (сульфати), мг/л	≤50	51–100	101–200	201–500	>500

Рисунок 1.4 – Експертна таблиця класифікації якості води

Представлена класифікаційна таблиця об’єднує основні хімічні та екологічні показники, які використовують для оцінки якості поверхневих вод. Оскільки в офіційних джерелах України немає єдиної універсальної шкали, що поділяє води на категорії “відмінна”, “добра”, “задовільна”, “погана” та “дуже погана” для всіх показників одночасно, діапазони значень сформовані на основі європейських екологічних систем оцінки (зокрема Water Framework Directive), рибогосподарських нормативів та санітарних норм, а також даних моніторингу річкових систем України. Таким чином, запропонована таблиця є узагальненою, науково обґрунтованою та придатною для застосування в аналітичних роботах і звітах [13].

Категорії “відмінна”, “добра”, “задовільна”, “погана” та “дуже погана” відображають ступінь екологічної небезпеки та відхилення показників від природних або нормативних значень:

- Відмінна — відповідає природному стану води, мінімальне антропогенне навантаження;
- Добра — незначні відхилення, що не впливають на екосистему.
- Задовільна — помірне забруднення, окремі показники наближаються до граничних значень;
- Погана — суттєві порушення, екосистема працює з перевантаженням;
- Дуже погана — критичний стан, значні перевищення більшості норм.

Концентрація амонію є важливим індикатором забруднення води органічними та побутовими стоками. Низькі значення характерні для чистих річок, тоді як підвищення концентрацій свідчить про активні процеси розкладу органічної речовини, скиди стічних вод або недостатній рівень самоочищення водного об'єкта. Значення понад 1 мг/л вказують на погіршення екологічного стану, а перевищення понад 2 мг/л свідчить про суттєве антропогенне навантаження.

Нітрити є проміжною формою азоту між амонієм і нітратами, тому вони швидко окиснюються за наявності кисню. Їх присутність в підвищених концентраціях означає порушення природних біохімічних процесів. Навіть невеликі перевищення нітритів становлять екологічну небезпеку, тому діапазони їх оцінки мають низькі порогові значення. Показник понад 0.1 мг/л вказує на несприятливий стан річки.

Нітрати є кінцевим продуктом перетворення азоту в природних водах і менш токсичні, ніж нітрити чи амоній. Проте їх підвищена концентрація свідчить про надмірне надходження добрив або побутових стоків. Значення до 10 мг/л вважаються характерними для чистих та малозабруднених річок. Показники понад 50 мг/л є ознакою інтенсивного сільськогосподарського впливу та порушення стану водойми.

Фосфати є однією з ключових речовин, що викликають евтрофікацію — надмірний ріст водоростей та “цвітіння” води. Навіть незначні підвищення

концентрації (більше 0.1 мг/л) можуть спричинити деградацію водної екосистеми. Води з концентрацією фосфатів понад 0.5 мг/л належать до категорії "дуже погана якість".

БСК5 характеризує кількість кисню, необхідного мікроорганізмам для розкладу органічних речовин у воді. Чим вища величина БСК5, тим більше органічного забруднення міститься у воді. Значення до 3 мг O₂/л характерні для чистих вод, тоді як показники вище 7 свідчать про суттєву деградацію водних масивів.

Цей показник демонструє кількість частинок, що знижують прозорість води. Їх надмір містить як природні (ерозійні процеси), так і антропогенні джерела (скиди підприємств, міські стоки). Високі концентрації негативно впливають на рибу та інші водні організми, зменшують доступ світла та прискорюють замулення.

Кисень є основним фактором життєздатності водних організмів. Високе насичення (>90%) характерне для швидкотечних, чистих річок. Зниження до рівня нижче 60% свідчить про погіршення умов для риби та безхребетних, а рівень нижче 40% є критичним і часто призводить до заморних явищ.

Підвищені концентрації хлоридів часто вказують на вплив побутових стоків або промислових відходів. Невеликі значення (до 50 мг/л) є нормальними, але понад 150 мг/л свідчать про суттєве антропогенне навантаження, а перевищення понад 300 мг/л характерні для вод з дуже поганою якістю.

Сульфати можуть надходити у воду природним шляхом (розчинення гірських порід), однак їх підвищений вміст часто пов'язаний із промисловими стоками. Концентрації до 100 мг/л є типовими для чистих вод, тоді як рівень понад 200 мг/л вказує на значне забруднення.

Особлива увага приділяється речовинам, що становлять загрозу для здоров'я людини та біоти, відповідно до вимог Водної рамкової директиви ЄС. Це забезпечує гармонізацію національної системи моніторингу з європейськими стандартами та сприяє підвищенню її достовірності.

У 2025 році оператором басейнової ради Південного Бугу заплановано проведення комплексного аналізу якісного стану масивів поверхневих вод, зокрема малих річок, які є найбільш вразливими до локального забруднення. Паралельно здійснюється системний огляд потенційних загроз, серед яких зміни гідрохімічного складу, скиди недостатньо очищених стічних вод, накопичення біогенних речовин та органічних забруднювачів.

Басейнова рада Південного Бугу формує план роботи, що охоплює питання організації моніторингу, контролю за виконанням природоохоронних заходів та періодичну оцінку екологічного стану водних об'єктів. Контрольні пункти спостережень розміщуються по всій території басейну та охоплюють ключові ділянки гідрографічної мережі, що дозволяє отримувати репрезентативні дані про стан вод у просторі та часі.

Стратегічною метою управління водними ресурсами басейну Південного Бугу визначено досягнення та підтримання «доброго» екологічного статусу поверхневих і підземних вод відповідно до європейських екологічних вимог. Реалізація цієї мети передбачає подолання низки проблем, серед яких надмірна зарегульованість, інтенсивне ерозійне навантаження, недостатній рівень впровадження сучасних методів гідрохімічного аналізу та необхідність інтеграції інноваційних інформаційних технологій у систему моніторингу [14].

Сучасний стан моніторингу басейну Південного Бугу характеризується переходом від фрагментарного спостереження до комплексної, системної моделі управління, що поєднує екологічні, гідрологічні та інформаційно-аналітичні підходи. Це створює передумови для підвищення ефективності прогнозування стану водних масивів та забезпечення сталого розвитку регіону.

1.4 Особливості прогнозування рівня забрудненості

Прогнозування рівня забрудненості водних масивів є ключовим етапом системи управління водними ресурсами, оскільки дозволяє своєчасно виявляти потенційні екологічні ризики та приймати ефективні управлінські рішення. Для басейну річки Південний Буг прогнозування здійснюється на основі комплексного аналізу спостережних даних, включаючи хімічні, фізичні та біологічні показники якості води, гідрологічні характеристики, а також інформацію про антропогенне навантаження на водні об'єкти.

Особливістю прогнозування є необхідність інтегрувати різноманітні джерела даних. Це включає результати лабораторного контролю, автоматичні датчики стану води, супутникові спостереження, а також відкриті дані про господарську діяльність у регіоні. Таке поєднання дозволяє моделювати зміну концентрацій забруднювачів у просторі та часі, оцінювати тенденції та можливі екстремальні ситуації, наприклад, перевищення нормативних рівнів забруднення після сильних опадів або скидних вод.

Для підвищення точності прогнозування застосовуються сучасні інформаційні технології: геоінформаційні системи (ГІС) для просторового аналізу, моделі гідрохімічної динаміки, а також методи машинного навчання та штучного інтелекту. Вони дозволяють будувати адаптивні моделі, які враховують сезонні та довгострокові коливання гідрологічного режиму, взаємозв'язки між різними забруднювачами та ймовірність виникнення синергетичного ефекту при одночасному впливі декількох факторів.

Ще однією специфікою прогнозування є необхідність враховувати локальні особливості басейну, такі як зарегульованість річки, наявність водосховищ, ерозійне навантаження та концентрацію сільськогосподарської діяльності у водозборі. Ці фактори істотно впливають на швидкість розсіювання забруднювачів і формування їхніх концентрацій у різних ділянках річки.

Впровадження адаптивного парсингу даних моніторингу та великих мовних моделей дозволяє автоматизувати збір і обробку інформації, що надходить з різних джерел. Це забезпечує швидке виявлення аномалій, формування прогнозних сценаріїв та оперативне планування заходів щодо мінімізації негативного впливу забруднення на екосистеми та населення.

Прогнозування рівня забрудненості водних масивів басейну Південного Бугу ґрунтується на інтеграції великих обсягів даних, використанні сучасних ІТ-рішень та врахуванні регіональних особливостей, що дозволяє підвищити точність оцінки екологічного стану та ефективність прийняття управлінських рішень.

1.5 Висновки

У розділі проведено огляд басейну річки Південний Буг, його природних характеристик, водогосподарського значення та джерел забруднення. Проаналізовано європейські та українські системи моніторингу вод, оцінено їх переваги й обмеження, а також сучасний стан моніторингу та якість води за основними показниками. Охарактеризовано методи прогнозування забрудненості, включно з ГІС, математичним моделюванням і ШІ. Результати дали цілісне уявлення про стан водних ресурсів і можливі шляхи вдосконалення моніторингу.

2 РОЗРОБЛЕННЯ ТА ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ АНАЛІЗУ ДАНИХ МОНІТОРИНГУ ЯКОСТІ ПОВЕРХНЕВИХ ВОД

2.1.1 GeoPandas

Інтеграція геопросторових даних здійснюється за допомогою бібліотек тому для вирішення поставленого завдання було обрано GeoPandas, Shapely та Folium, які дозволяють об'єднувати аналітичну інформацію з просторовими координатами, формувати карти поширення забруднень, відображати межі басейнів, водозборів і точок спостережень. Завдяки цьому система дає можливість візуалізувати просторово-часову динаміку стану водних об'єктів та оцінювати вплив антропогенних факторів (рис. 2.2).

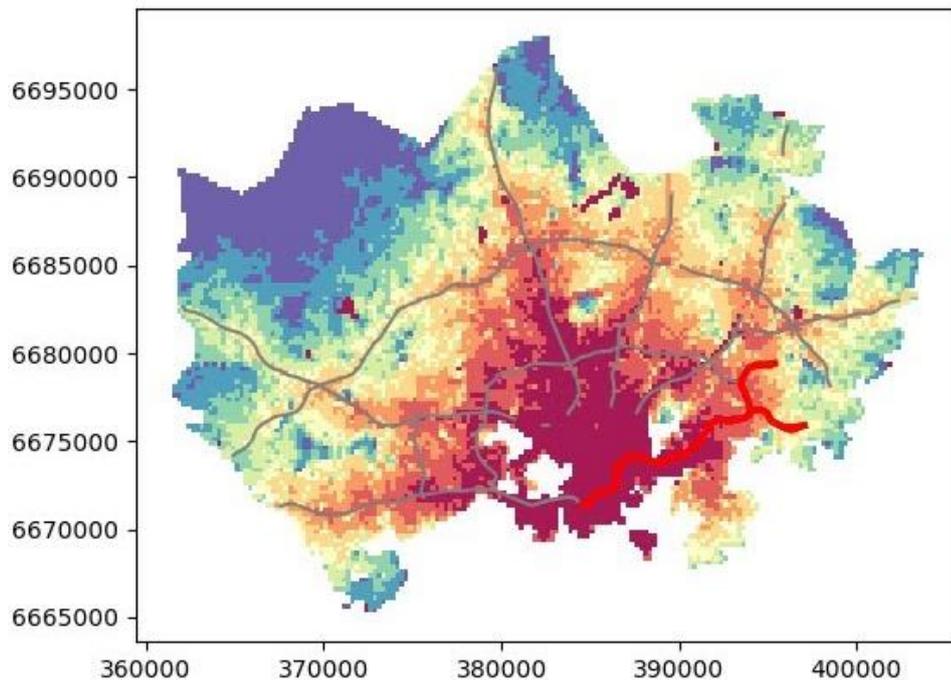


Рисунок 2.2 – Побудова шейп-файлу за допомогою GeoPandas

Бібліотека GeoPandas є одним із найважливіших інструментів для роботи з просторовими даними у Python.

Вона ґрунтується на можливостях Pandas, але розширює їх геометричними операціями, що дозволяє обробляти шейп-файли, геодані у форматах GeoJSON, KML, GPKG та інших.

GeoPandas забезпечує можливість завантаження, об'єднання, фільтрації та візуалізації просторових об'єктів, що є критично важливим для аналізу екологічної ситуації в басейні Південного Бугу.

Наприклад, за допомогою цієї бібліотеки можна відобразити межі басейнів, розташування контрольних пунктів спостережень, масиви поверхневих вод, а також інтегрувати їх з даними моніторингу.

GeoPandas дозволяє легко визначати пересічення геометричних об'єктів, знаходити відстані між точками, виконувати буферизацію та інші просторові операції, що значно полегшує аналітичні задачі.

Бібліотека Shapely доповнює GeoPandas і є низькорівневим інструментом, який відповідає за геометричні обчислення. Вона дозволяє працювати з геометричними типами — точками, лініями, полігонами — та виконувати над ними складні операції просторового аналізу: визначення площ, периметрів, побудову буферних зон, об'єднання, різницю, перетин та інші.

Завдяки Shapely стає можливим аналіз географічного впливу забруднень, наприклад, визначення територій, які знаходяться в зоні впливу певних джерел забруднення або гідротехнічних споруд.

Це дає змогу моделювати потенційне поширення забруднювальних речовин у межах русла або водозбірної площі.

2.1 Вибір оптимальних інформаційних технологій

Для реалізації інформаційної технології аналізу стану поверхневих вод басейну річки Південний Буг було обрано комплекс сучасних інструментів і

технологій, які забезпечують повний цикл роботи з екологічними даними — від збору та попередньої обробки до інтеграції, аналізу, моделювання й візуалізації результатів (рис. 2.1).

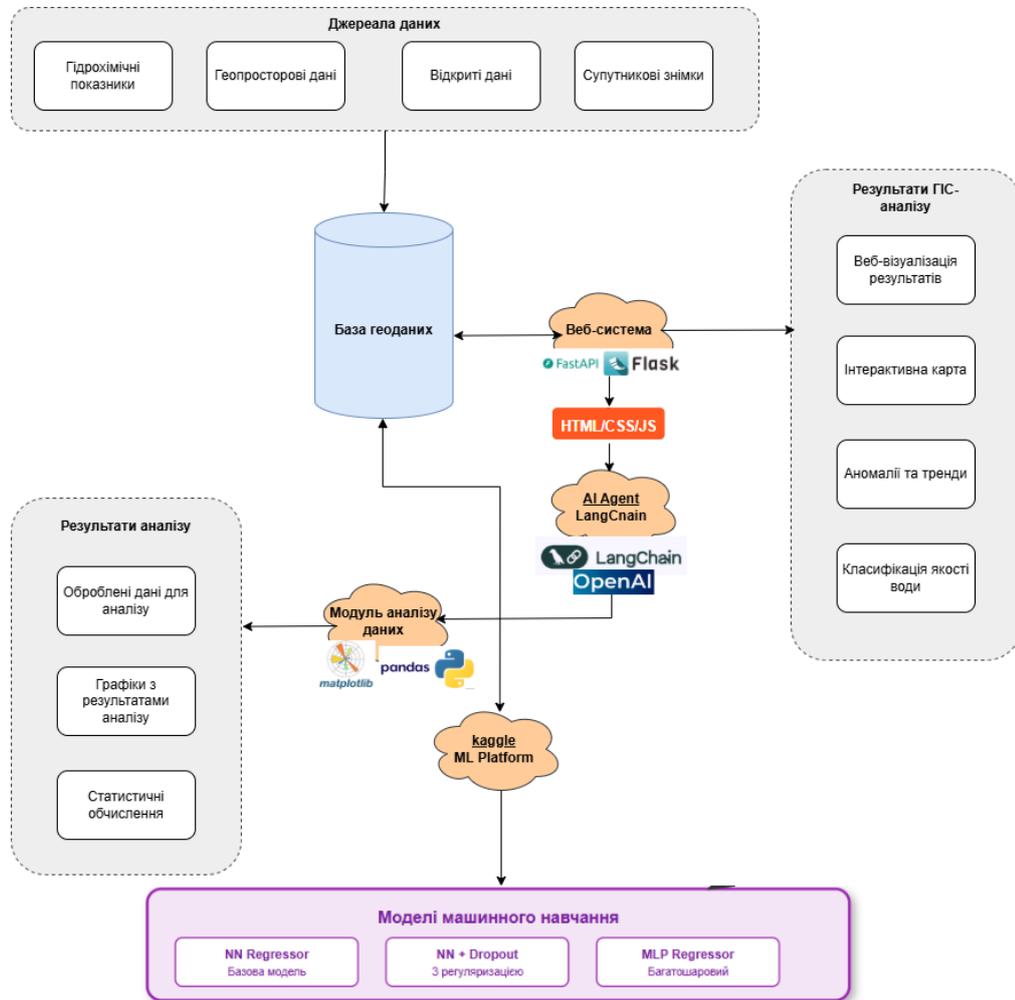


Рисунок 2.1 – Обрані оптимальні інструменти і технології

Такий підхід дозволяє створити гнучку, масштабовану та надійну систему, орієнтовану на підтримку прийняття рішень у сфері управління водними ресурсами.

Зокрема, для обробки великих обсягів моніторингових даних, що надходять із сенсорних систем та лабораторних вимірювань, застосовано модулі аналізу даних на основі Python. Використання бібліотек Pandas, NumPy та SciPy забезпечує ефективну роботу з табличними, часовими та числовими

форматами, що є важливою складовою при аналізі динаміки показників якості води.

Одним із ключових завдань побудови системи аналізу стало забезпечення якісної обробки великих обсягів моніторингових даних, які надходять із різних джерел. Це можуть бути як автоматизовані сенсорні комплекси, що фіксують показники у режимі реального часу (температура води, рівень рН, розчинений кисень, електропровідність, мутність, концентрація амонію, нітратів та інших іонів), так і результати лабораторних аналізів, які містять точні хімічні вимірювання пріоритетних забруднювальних речовин. Для ефективної роботи з такими даними застосовано модулі аналізу на основі мови програмування Python — одного з найбільш поширених та гнучких інструментів у сфері екологічного моделювання та наукових обчислень.

Використання бібліотек Pandas, NumPy та SciPy забезпечує можливість обробляти табличні структури, часові ряди та числові масиви, що є вкрай важливим при аналізі динаміки показників якості води. Pandas дозволяє легко поєднувати різні джерела даних, виконувати фільтрацію, нормалізацію, корекцію пропусків та аномалій, тоді як NumPy забезпечує високошвидкісні математичні операції з багатовимірними масивами. SciPy, у свою чергу, розширює можливості чисельного аналізу і надає інструменти для статистичної обробки, інтерполяції, оптимізації, що має важливе значення при моделюванні процесів, пов'язаних із транспортом та трансформацією забруднювальних речовин у водному середовищі.

Застосування Python робить систему не лише продуктивною, а й адаптивною до нових вимог. У разі розширення переліку показників або зміни структури моніторингової мережі, інструментарій може бути швидко модифікований, що забезпечує довгострокову актуальність розробленої технології. Крім того, відкритість екосистеми Python сприяє інтеграції з іншими платформами, зокрема з геоінформаційними системами, хмарними сервісами та веб-інтерфейсами для представлення екологічної інформації.

Важливою складовою створюваної інформаційної технології є інтеграція отриманих даних у просторовий контекст. Оцінка стану поверхневих вод неможлива без врахування географічного положення, структури водозбору, гідрологічного режиму та характеристик прилеглих територій. Саме тому для просторової обробки даних застосовано геоінформаційні технології (ГІС), які дозволяють об'єднувати аналітичні розрахунки з просторовими шарами, створювати інтерактивні карти, проводити зональний аналіз, визначати джерела забруднення та відстежувати екологічні тенденції на різних ділянках річкового басейну.

Для картографічної візуалізації було обрано інструменти на основі QGIS та веб-технологій Leaflet/Mapbox, які забезпечують зручний доступ до просторових даних як для користувачів, так і для розробників системи. Їхнє застосування дозволяє формувати шари концентрацій забруднювачів, відображати контрольні пункти спостережень, виконувати порівняння стану вод у різні роки та будувати аналітичні поверхні на основі інтерполяції. Візуальні карти слугують не лише для наочного представлення результатів, але й для прийняття управлінських рішень, оскільки дозволяють одразу виявляти проблемні ділянки та оцінювати масштаби впливу певних джерел забруднення.

Крім того, важливою задачею є стандартизація даних, оскільки моніторингові показники можуть походити з різних систем та мати відмінні формати, частоту оновлення, одиниці вимірювання та рівень точності. Для усунення цих розбіжностей застосовуються засоби попередньої обробки, нормалізації та уніфікації, що дозволяє створити єдиний структурований набір даних, придатний для порівняння та аналітичних операцій. Особлива увага приділяється обробці пропущених або некоректних значень, що є характерною проблемою екологічних спостережень, оскільки сенсори можуть виходити з ладу, дані можуть бути спотворені погодними умовами чи технічними збоями.

Після уніфікації дані проходять етап статистичного аналізу, під час якого визначаються середні значення, мінімальні та максимальні концентрації,

сезонні коливання, тенденції та аномалії. Такий підхід дозволяє не лише описати поточний стан водних об'єктів, але й виявити довгострокові зміни, що мають важливе значення для оцінки екологічної ситуації в басейні річки Південний Буг. Наприклад, можна визначити зростання рівня біогенних речовин у певні роки, зміни в сезонності процесів евтрофікації або зміну показників розчиненого кисню у зв'язку з температурними процесами.

У рамках інформаційної технології передбачено можливість моделювання сценаріїв розвитку екологічної ситуації. Це включає прогнозування поширення забруднювальних речовин у разі аварійних скидів, оцінку впливу кліматичних змін, моделювання наслідків тривалих періодів маловоддя або інтенсивних опадів. Для таких задач застосовуються математичні моделі гідродинаміки та гідрохімічних процесів, які в поєднанні з машинним навчанням дозволяють створювати адаптивні прогнозні системи.

Сучасні методи машинного навчання забезпечують додаткові можливості при роботі з великими масивами моніторингових даних. Зокрема, алгоритми класифікації та кластеризації дозволяють автоматично визначати типові режими якості води, виявляти нетипові ситуації та групувати подібні за характеристиками ділянки річкового русла. Моделі регресії застосовуються для прогнозування концентрацій конкретних забруднювачів, у той час як нейронні мережі можуть враховувати складні нелінійні взаємозв'язки між показниками, що зумовлені природними та антропогенними факторами.

Уся система візуалізації та аналізу працює у взаємодії з базою даних, у якій зберігаються результати обчислень, просторові об'єкти, історичні дані моніторингу та метадані вимірювань. Це дозволяє створити цілісну аналітичну платформу, яка може використовуватися органами державного управління, екологічними службами, науковими установами та іншими користувачами, що працюють з екологічною інформацією.

Реалізація інформаційної технології аналізу стану поверхневих вод басейну річки Південний Буг базується на сучасному інструментарії обробки, моделювання та візуалізації даних, що забезпечує високу точність,

масштабованість та можливість подальшого розвитку. Обраний підхід дозволяє суттєво підвищити ефективність екологічного моніторингу, забезпечити інтеграцію різних джерел інформації та створити основу для прийняття науково обґрунтованих рішень у сфері управління водними ресурсами.

Порівняння отриманих результатів дало змогу визначити та відібрати найбільш ефективні моделі прогнозування, здатні забезпечити високу точність оцінювання стану водних об'єктів на основі наявних моніторингових даних. У процесі дослідження проводився аналіз різних підходів машинного навчання та нейромережових архітектур, що дало можливість оцінити їхню продуктивність, швидкість збіжності, стійкість до шумів і здатність до узагальнення даних. Ретельне порівняння моделей дозволило виявити ті алгоритми, які найкраще відображають нелінійні закономірності у динаміці екологічних показників і найбільш адекватно реагують на зміну вхідних параметрів.

Завдяки такому підходу вдалося підвищити точність моделювання та достовірність прогнозів, що є критично важливим для систем екологічного моніторингу, де навіть незначні похибки можуть призвести до неправильного тлумачення стану водного середовища. Використання оптимізованих моделей дозволяє своєчасно виявляти негативні тенденції, прогнозувати можливі сценарії деградації якості води та формувати рекомендації для прийняття управлінських рішень.

Отже, впровадження найефективніших моделей на основі результатів порівняльного аналізу забезпечує надійність і наукову обґрунтованість роботи інформаційної системи, роблячи її дієвим інструментом для моніторингу, аналізу та прогнозування стану поверхневих вод. Це створює основу для подальшого розвитку екологічних інформаційних технологій і сприяє підвищенню ефективності управління водними ресурсами в регіональному та національному масштабах [15].

2.1.2 FastAPI та Flask

Для забезпечення вебдоступу та взаємодії користувача з системою реалізовано інтерфейс на основі HTML5, CSS3 та JavaScript (ES6+), який забезпечує зручну навігацію, адаптивний дизайн і швидке завантаження сторінок. Серверну частину побудовано на платформі Python з використанням FastAPI та Flask, що дозволяють створювати високопродуктивні асинхронні API та мікросервіси (рис. 2.3).

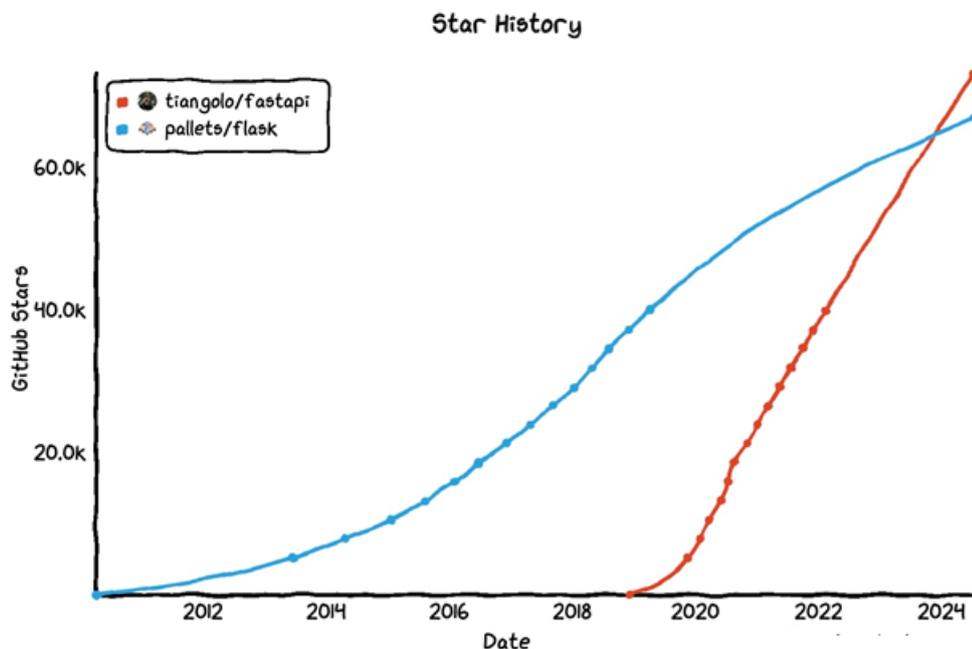


Рисунок 2.3 – FastAPI проти Flask порівняння для веб-розробки на Python

Серед двох фреймворків FastAPI сьогодні використовується частіше для нових проєктів, оскільки він має кращу продуктивність, підтримує асинхронне програмування «з коробки» та автоматично генерує документацію для API (Swagger/OpenAPI). Flask залишається популярним для простих або вже існуючих проєктів, де важлива гнучкість та широка екосистема розширень.

FastAPI, своєю чергою, використано для створення високопродуктивних асинхронних API, здатних обробляти великий обсяг запитів у реальному часі. Це є критично важливим для екологічних систем, що працюють із потоковими

даними або надають доступ до великих масивів моніторингової інформації. Завдяки FastAPI система отримує можливість швидко реагувати на запити, забезпечує низьку затримку при передачі результатів аналізу та підтримує масштабованість у випадку зростання навантаження.

2.1.3 LangChain

Для вирішення поставленого завдання було запропоновано, системи інтегровано елементи штучного інтелекту — зокрема, OpenAI API та бібліотеку LangChain, які забезпечують обробку природномовних запитів користувачів, інтелектуальний пошук інформації та генерацію аналітичних звітів. Це робить систему більш інтерактивною та здатною до адаптивної взаємодії з користувачем (рис. 2.4).

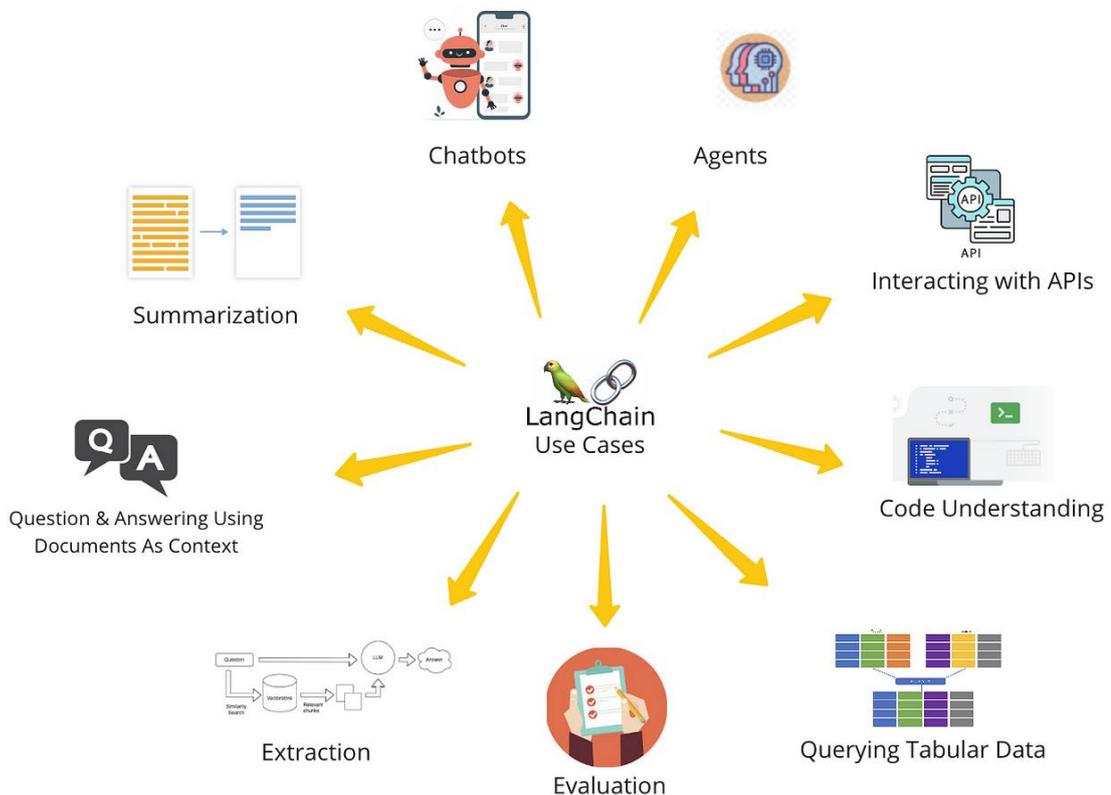


Рисунок 2.4 – Можливості LangChain

Окрім обробки запитів, система забезпечує інтелектуальний пошук інформації, що значно підвищує точність та швидкість отримання необхідних даних. На відміну від традиційних методів пошуку, які часто обмежуються простим співпадінням ключових слів, використання інтелектуальних алгоритмів дозволяє враховувати семантичні зв'язки між поняттями, контекст запиту та історію попередніх взаємодій користувача з системою. Це дає змогу швидко знаходити релевантну інформацію навіть у великих масивах даних і забезпечує більш комплексне розуміння предметної області, що особливо важливо у випадках аналітичної роботи чи підготовки звітів.

Для підвищення аналітичних можливостей застосовано бібліотеку LangChain, яка дозволяє ефективно об'єднувати різні компоненти штучного інтелекту та створювати складні логічні ланцюжки обробки інформації. Це означає, що система здатна не лише збирати та аналізувати дані, а й генерувати структуровані аналітичні звіти, підсумовуючи великі обсяги інформації у зручній для користувача формі. Звіти можуть містити узагальнені висновки, графічні візуалізації, таблиці та ключові рекомендації, що дозволяє користувачу швидко орієнтуватися у результатах та приймати обґрунтовані рішення.

Завдяки інтеграції цих технологій система стає більш інтерактивною та адаптивною. Вона може реагувати на поведінку користувача, підлаштовуватися під його стиль спілкування та пропонувати індивідуалізовані підказки або рекомендації. Це створює ефект «живого» асистента, який не просто надає інформацію, а й активно допомагає у її обробці та використанні. Така інтерактивність підвищує зручність роботи із системою, робить процес навчання чи аналізу даних більш ефективним і дозволяє користувачу зосередитися на важливих аспектах завдання, не витрачаючи час на рутинну обробку інформації. скористатись інтернет середовищем Kaggle.

2.1.4 Kaggle

Для вирішення поставленого завдання було запропоновано, скористатись інтернет середовищем Kaggle.

Серед основних переваг Kaggle слід виділити можливість швидкого доступу до численних відкритих датасетів, що значно спрощує підготовку даних для навчання моделей та тестування алгоритмів. Платформа містить багатий каталог даних різного типу та складності, від текстових і числових до зображень і аудіо, що дозволяє реалізовувати як прості експерименти, так і складні дослідницькі проекти.

Також Kaggle надає зручне середовище для написання та виконання коду на Python та R без необхідності встановлення додаткових бібліотек чи налаштування локальної системи. У цьому середовищі можна відразу інтегрувати популярні бібліотеки для аналізу даних та машинного навчання, а також використовувати сучасні інструменти для обробки природної мови чи комп'ютерного зору.

Крім того, Kaggle дозволяє легко організовувати експерименти та відслідковувати результати, зберігати версії моделей і порівнювати їхню продуктивність. Це особливо важливо при роботі з штучним інтелектом, коли оцінка точності та ефективності алгоритмів є ключовою для прийняття рішень щодо оптимізації системи.

Завдяки використанню Kaggle система отримує можливість оперативного тестування різних методів аналізу та моделювання, що пришвидшує процес розробки і дозволяє ефективно перевіряти гіпотези.

Це робить платформу цінним інструментом для дослідників і розробників, які прагнуть інтегрувати передові технології у свої проекти та забезпечити високу якість результатів.

На рисунку 2.5, зображено датасет, на основі якого було виконано дану роботу.

```

from sklearn.metrics import mean_squared_error as mse

import warnings
warnings.simplefilter('ignore')

measurement = pd.read_csv('../input/sdsddj/measurement.csv')
measurement.head()

```

	ID_Station	Date	NH4	BSK5	Suspended	O2	NO3	NO2	SO4	PO4	Cl
0	1	4/17/2000	0.035	1.30	NaN	12.98	4.70	0.038	403.2	0.143	146.1
1	1	7/4/2000	0.031	1.13	10.4	9.89	0.24	0.022	416.0	0.060	149.0
2	1	12/4/2000	0.042	1.20	NaN	9.63	2.02	0.006	384.0	0.275	154.5
3	1	2/6/2001	0.062	2.88	1.6	11.77	4.30	0.004	364.8	0.380	160.1
4	1	4/3/2001	NaN	2.50	15.6	11.40	11.40	0.023	397.0	0.197	146.0

```

measurement.columns

Index(['ID_Station', 'Date', 'NH4', 'BSK5', 'Suspended', 'O2', 'NO3', 'NO2',
      'SO4', 'PO4', 'Cl'],
      dtype='object')

```

Рисунок 2.5 – Датасет у середовищі Kaggle

У межах завдання з підбору оптимальних ІТ-рішень і моделей машинного навчання платформа Kaggle забезпечує зручне середовище для експериментів із даними та навчання моделей. Було протестовано кілька моделей на основі нейронних мереж, зокрема:

- NN Regressor — базова модель нейронної мережі для регресії;
- NN Regressor with Dropout — варіант із використанням Dropout-регуляризації для зменшення перенавчання;
- MLP Regressor — модель багатошарового перцептрону, що забезпечує гнучке наближення складних залежностей між вхідними та вихідними параметрами.

Щоб визначити найбільш ефективну для прогнозування та аналізу даних. Перша модель, NN Regressor, представляє собою базову нейронну мережу для регресії, яка дозволяє апроксимувати залежності між вхідними та вихідними параметрами. Вона забезпечує простий і зрозумілий підхід до прогнозування, проте може бути схильною до перенавчання при роботі зі складними або великими наборами даних.

Другим підходом стала модель NN Regressor with Dropout, яка є модифікацією базової нейронної мережі з використанням Dropout-регуляризації. Цей метод дозволяє зменшити ризик перенавчання, випадковим чином відключаючи частину нейронів під час навчання, що сприяє підвищенню здатності моделі до узагальнення та покращує її стійкість до шуму в даних. Такий підхід особливо корисний при роботі зі складними багатовимірними даними, де звичайні мережі можуть демонструвати нестабільні результати.

Ще одним тестованим варіантом стала модель MLP Regressor, що базується на багатошаровому перцептроні. Ця архітектура забезпечує гнучке наближення складних залежностей між вхідними та вихідними параметрами завдяки використанню кількох прихованих шарів та нелінійних функцій активації. MLP Regressor дозволяє адаптувати модель до специфіки даних і досягати точності прогнозування, що робить її ефективним інструментом для задач регресії у різних сферах застосування.

Використання Kaggle для тестування цих моделей забезпечує швидке порівняння їх продуктивності, дозволяє аналізувати результати та робити висновки щодо оптимальної архітектури та налаштувань моделі для конкретного завдання. Такий підхід дозволяє підвищити ефективність процесу розробки, забезпечує наочність результатів і сприяє прийняттю обґрунтованих рішень при виборі моделі машинного навчання.

2.2 Аналіз геопросторових зв'язків

Для забезпечення наочного подання результатів дослідження було застосовано середовище Google Colab, у якому було побудовано відповідні графіки. Вони дали змогу не лише подати інформацію у зручному форматі, а й поглибити аналіз водозбірних територій та їх перетину з населеними пунктами.

Візуалізація сприяла кращому розумінню просторових взаємозв'язків між водозбірними басейнами та населеними територіями, а також дала можливість оцінити вплив цих басейнів на окремі аспекти розвитку місцевих громад. Такий підхід підвищує якість аналізу та допомагає враховувати фактори, що формують розвиток регіону.

Для створення графічних матеріалів було використано Python разом із популярними бібліотеками Matplotlib, Pandas та Seaborn. На першому етапі було імпортовано необхідні інструменти та завантажено набір даних для подальшої обробки (рис. 2.6).

```
[ ] import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

#Завантаження та огляд даних
df = pd.read_csv('catchment_Intersect9_TabletoExcel.csv')
df
```

	FID	FID_catchment	SEM9	SEM326	FID_Метарічки	SEM9.1	SEM15	SEM150	SEM152	SEM327	...	type_1	codename_1	adminua_1	adminru_1	adminen_1	koatuu_1	nameua_ne
0	1	155	UA_M5.4_0254		48	Без назви (с. Зарічне)	МЕНЕЕ 20 М			0	...	Області, автономії та ради	Межі областей та автономни	Вінницька	Вінницкая	Vinnytska	500000000	Гуральн
1	2	155	UA_M5.4_0254		48	Без назви (с. Зарічне)	МЕНЕЕ 20 М			0	...	Області, автономії та ради	Межі областей та автономни	Вінницька	Вінницкая	Vinnytska	500000000	Шурі Копілівськ
2	3	169	UA_M5.4_0272		49	Самець (р.Сільниця)	МЕНЕЕ 20 М	277		0	...	Області, автономії та ради	Межі областей та автономни	Вінницька	Вінницкая	Vinnytska	500000000	Кришин
3	4	169	UA_M5.4_0272		49	Самець (р.Сільниця)	МЕНЕЕ 20 М	277		0	...	Області, автономії та ради	Межі областей та автономни	Вінницька	Вінницкая	Vinnytska	500000000	Тарасівк
...	Самець	МЕНЕЕ	Області, автономії та ради	Межі областей та автономни

Рисунок 2.6 – Завантаження датасету

Для визначення кількості перетинів річок із населеними пунктами різних категорій, що кодуються змінною codename, доцільно використати інструменти аналізу даних мовою Python. На першому етапі дані завантажуються та опрацьовуються за допомогою бібліотеки Pandas, яка забезпечує зручну роботу з табличними структурами, фільтрацію записів і групування за потрібними ознаками. Для цього застосовуються бібліотеки Matplotlib або Seaborn, які дозволяють побудувати інформативні та наочні візуалізації, налаштувати підписи осей, масштабування та стиль відображення даних (рис. 2.7).



Рисунок 2.7 – Графік за кількістю населених пунктів за типами

Стовпець "population" містить числові значення, що мають вагоме аналітичне значення та відкривають можливості для детальнішого вивчення демографічних особливостей досліджуваних територій. На основі цих даних можна аналізувати, як саме розподіляється чисельність населення між різними типами населених пунктів — селами, містечками, містами та селищами міського типу.

Створення різних видів графіків, які відображають цей розподіл, дає змогу наочно порівнювати категорії, виявляти тенденції й помічати відхилення. Використовуючи дані зі стовпця "population", можна підсумувати кількість населення для кожної групи населених пунктів, що дозволяє визначити їхній внесок у загальну демографічну структуру регіону.

Такий підхід дає змогу виявити можливі проблеми, зокрема надмірну щільність населення або недостатнє освоєння окремих територій, що може

стати підґрунтям для формування ефективної регіональної політики. Водночас подібний аналіз є важливою складовою стратегічного планування, орієнтованого на сталий розвиток, раціональний розподіл ресурсів і покращення якості життя населення.

Отже, стовпець "population" — це не просто числові дані, а цінне джерело інформації, яке допомагає оптимізувати управління територіями та підтримує розробку стратегій довгострокового розвитку регіонів (рис. 2.8).



Рисунок 2.8 – Графік за кількістю населених пунктів за типами.

На побудованому графіку добре простежується нерівномірність розподілу населення між різними категоріями населених пунктів. Зокрема, селища (с-ще) характеризуються найнижчими показниками чисельності населення, тоді як у містах зосереджена найбільша кількість жителів — 2 849 360 осіб. Це свідчить про високу концентрацію населення в міських агломераціях порівняно з іншими типами поселень.

Для аналізу розподілу даних відповідно до семантики SEM326, що містить код геоприв'язки в ArcGIS, можна застосувати графічну візуалізацію, яка відображатиме кількість записів для кожного коду. Такий підхід дозволяє краще побачити, як дані розподіляються між різними географічними зонами чи регіонами (рис. 2.9).

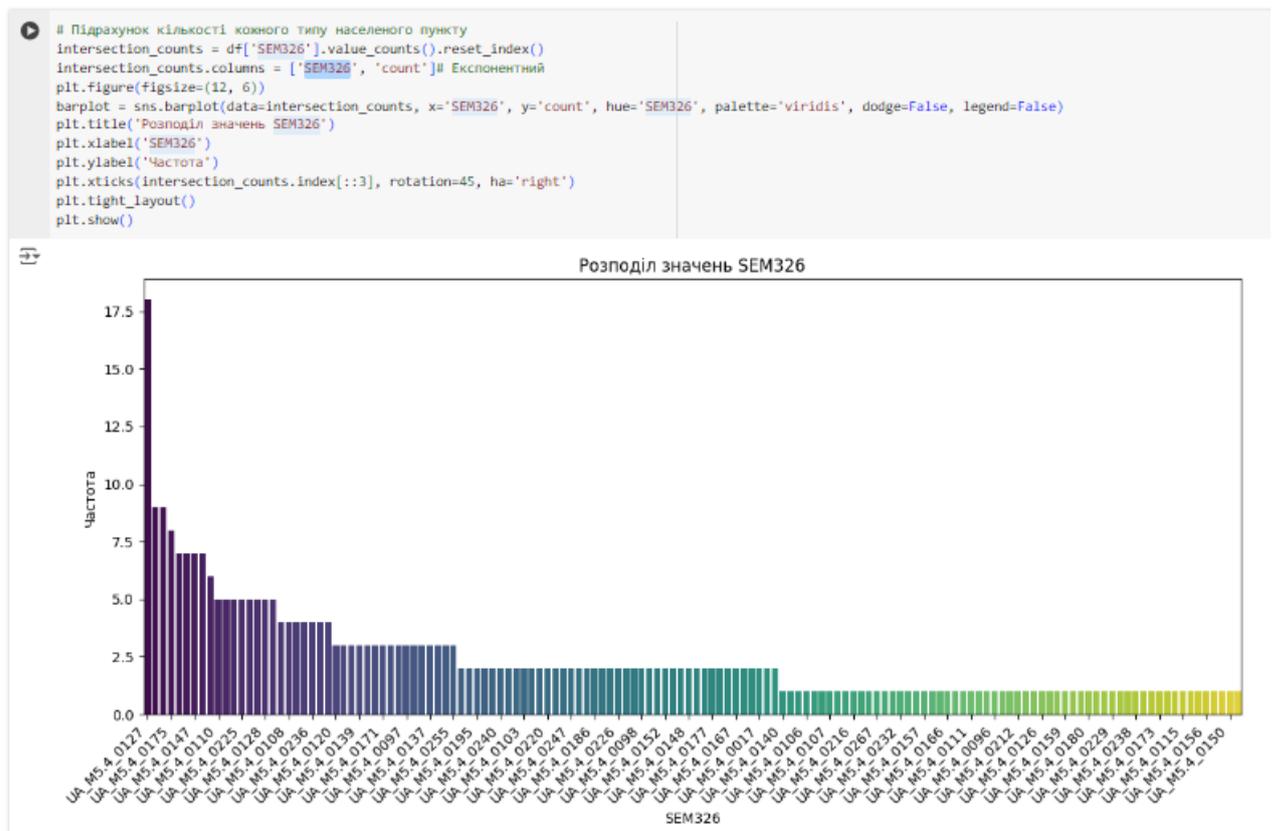


Рисунок 2.9 – Графік розподілу даних за географічними областями або регіонами.

Наступний графік відображає розподіл даних за кодом геоприв'язки та типами населених пунктів, включно зі смт (селищами міського типу). Така

візуалізація дозволить побачити, як саме різні типи поселень пов'язані з певними географічними зонами та допоможе виявити можливі закономірності між ними.

Для цього було створено окремий DataFrame, скопіювавши стовпці SEM326 та type, що містить класифікацію населених пунктів. Далі дані буде згруповано за кодом геоприв'язки та типом населеного пункту за допомогою функції groupby, після чого виконаємо підрахунок кількості записів у кожній групі. На завершальному етапі побудуємо стовпчикову діаграму, яка відобразить отриманий розподіл.

Такий підхід дає змогу чіткіше зрозуміти, як різні типи населених пунктів розміщуються у просторі, та визначити потенційні залежності або тенденції у їх географічному поширенні (рис. 2.10).



Рисунок 2.10 – Графік розподілу даних геоприв'язки по СМТ.

Графік, що відображає розподіл кількості перетинів водозбірних площ із масивами вод залежно від геокоду SEM326 та типу населеного пункту, дає змогу наочно порівняти, як часто такі перетини трапляються у різних категоріях поселень. Це дозволяє оцінити, як водні ресурси розподіляються з урахуванням географічної прив'язки та типології населених пунктів, а також

зрозуміти можливий вплив цих чинників на формування водозборів і управління ними.

Для візуалізації цього розподілу буде застосовано графік `histplot`, який побудує гістограму частоти перетинів водозбірних площ за типами населених пунктів у прив'язці до SEM326.

Такий формат графіка забезпечує швидке виявлення тенденцій, закономірностей і потенційних взаємозв'язків у структурі даних (рис. 2.11).

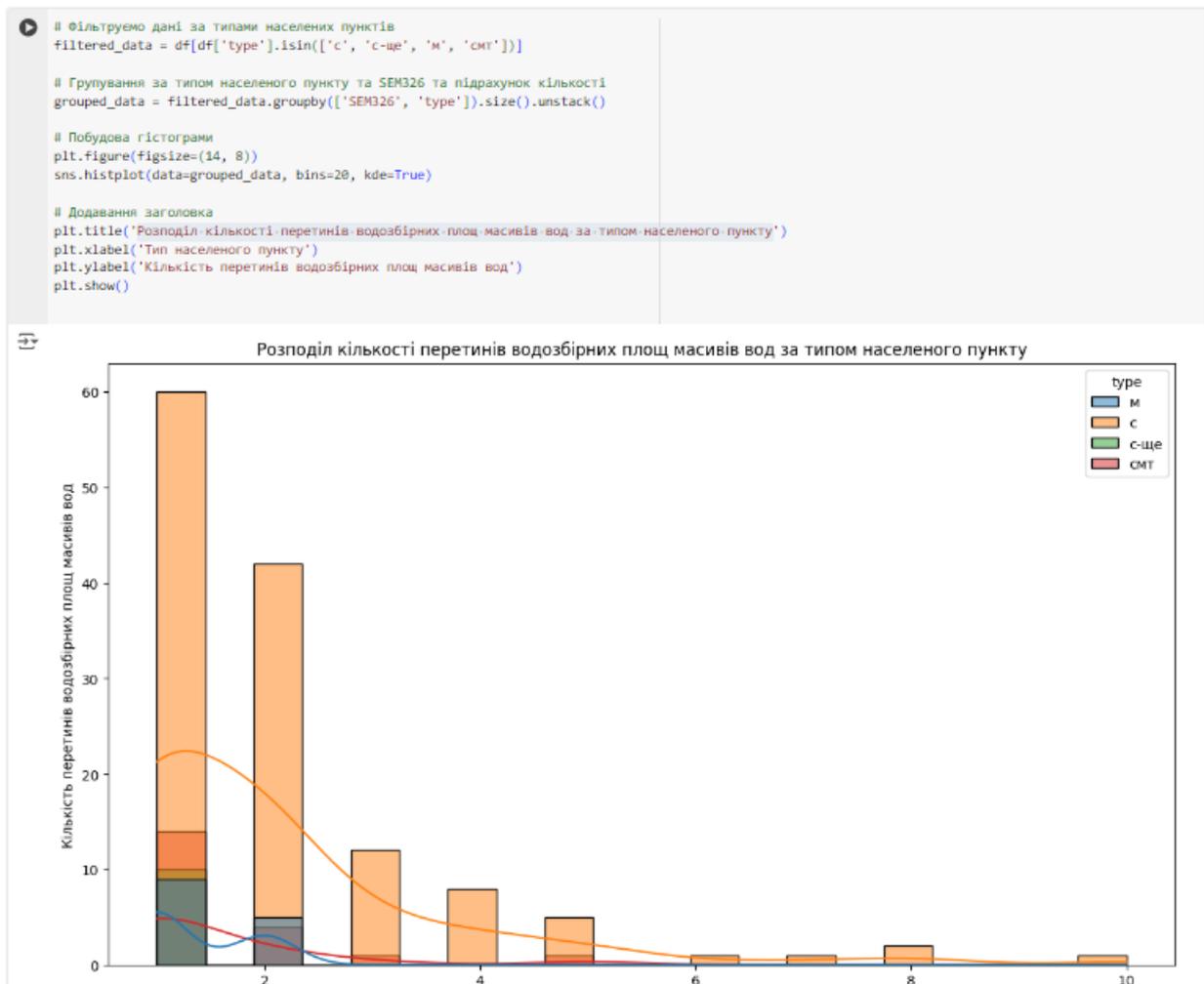


Рисунок 2.11 – Графік розподілу даних за геоприв'язкою та типом населеного пункту

Наступний графік, який відобразатиме розподіл даних за кодом геоприв'язки та типом населеного пункту, доповнений підрахованою кількістю поселень.

Ці значення будуть відображені над стовпцями, а використання різного кольору допоможе краще виділити кожен групу. Така візуалізація дозволяє детальніше проаналізувати, як тип населеного пункту пов'язаний із певною географічною зоною, та виявити потенційні закономірності.

Для цього застосовуємо функцію `isin()`, яка дає змогу розподілити дані на чотири категорії: 'с', 'с-ще', 'м', 'сmt'.

Після цього виконуємо групування за кодом геоприв'язки SEM326 та типом поселення `type`, а кількість населених пунктів у кожній комбінації визначаємо за допомогою методу `size()` (рис. 2.12).

```

# Фільтруємо дані за типами населених пунктів
filtered_data = df[df['type'].isin(['с', 'с-ще', 'м', 'сmt'])]

# Групування за типом населеного пункту та SEM326 та підрахунок кількості
grouped_data = filtered_data.groupby(['SEM326', 'type']).size().unstack(fill_value=0)

# Побудова стовпчастої діаграми
fig, ax = plt.subplots(figsize=(14, 8))

# Побудова стовпчиків з різними кольорами для кожного типу населеного пункту
colors = ['skyblue', 'lightgreen', 'salmon', 'orange']
bars = grouped_data.plot(kind='bar', stacked=True, ax=ax, color=colors, edgecolor='black')

# Додавання текстових надписів над кожним стовпчиком
for container in ax.containers:
    ax.bar_label(container)

# Додавання заголовка та підписів до осей
plt.title('Розподіл кількості перетинів водозбірних площ масивів вод за типом населеного пункту')
plt.xlabel('SEM326')
plt.ylabel('Кількість перетинів водозбірних площ масивів вод')

# Додавання легенди
plt.legend(title='Тип населеного пункту', labels=['с', 'с-ще', 'м', 'сmt'])

# Встановлення міток осі X
xticks = range(len(grouped_data.index))
ax.set_xticks(xticks)
ax.set_xticklabels(grouped_data.index, rotation=45, ha='right')

# Вибірково відображення міток осі X
for label in ax.get_xticklabels()[::2]:
    label.set_visible(True)
for label in ax.get_xticklabels()[1::2]:
    label.set_visible(False)

# Відображення графіку
plt.tight_layout()
plt.show()

```

Рисунок 2.12 – Фрагмент коду графіку.

кількість населених пунктів різних категорій змінюється залежно від регіону, що відображає різноманітність географічних умов та характеристик територій.

Крім того, візуалізація допомогла виявити потенційні закономірності та взаємозв'язки, наприклад, кількість поселень у різних географічних зонах або співвідношення між типами населених пунктів.

Загалом, графічне подання даних сприяє глибшому розумінню просторового розподілу та особливостей населених пунктів, що є корисним для подальшого аналізу, планування управління територіями та прийняття рішень у суміжних галузях, таких як географія та ресурсне планування.

На основі таблиці водозбірних площ була створена додаткова таблиця, яка містить загальну кількість перетинів із населеними пунктами, а також окремо за типами поселень: сільські ради (с-ще), села (с), міста (м) та селища міського типу (сmt).

Після формування цієї таблиці її можна зберегти у відповідному форматі за допомогою функції `to_excel()`. Далі у середовищі ArcGIS виконується операція Join на водозбірних площах масивів.

У вікні, що з'явиться, необхідно вказати шлях до створеного файлу та зв'язати поля SEM326 і code, щоб програма коректно розпізнавала геоприв'язку. Це дозволяє ArcGIS правильно ідентифікувати дані для подальшого використання та аналізу.

Після завершення підготовки даних можна створити тематичну карту. Для цього у властивостях шару потрібно перейти до вкладки Symbology, де налаштовується тип символу, кольори, прозорість, масштаб та інші параметри відображення.

Після налаштування всіх необхідних параметрів користувач має змогу переглянути попередній вигляд карти, оцінити розташування та відображення об'єктів, а також переконатися у коректності візуалізації. Після цього внесені зміни можна застосувати, зберігши оновлену карту у відповідному форматі для подальшого використання, аналізу або інтеграції з іншими даними та системами. Такий підхід забезпечує контроль на всіх етапах підготовки

картографічного матеріалу та дозволяє отримати якісний і наочний результат. (рис. 2.14).

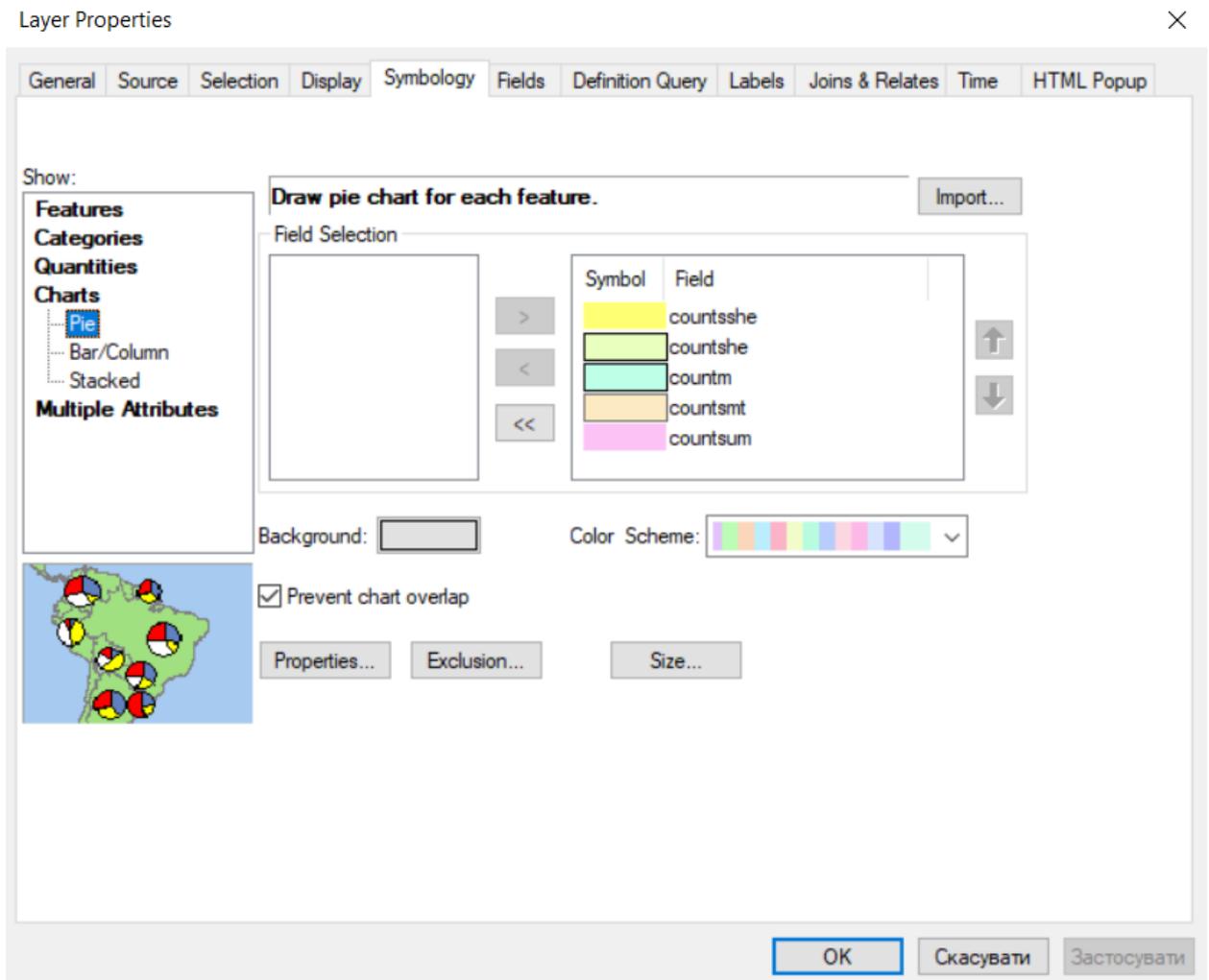


Рисунок 2.14 Вікно налаштування тематичної карти.

Для наочної візуалізації числових даних ми побудуємо кругові діаграми у вкладці Pie. Для цього слід обрати всі стовпці, які містять числові значення, та підтвердити вибір кнопкою ОК. В результаті програма відобразить розподіл значень кожного стовпця у вигляді окремих кругових діаграм.

Такий підхід дозволяє швидко оцінити пропорційне співвідношення різних категорій у даних, виявити домінуючі та менш представлені значення, а також наочно порівнювати їх між собою. Крім того, кругові діаграми полегшують сприйняття великих обсягів інформації і стають корисним інструментом для підготовки звітів чи презентацій (рис. 2.15).

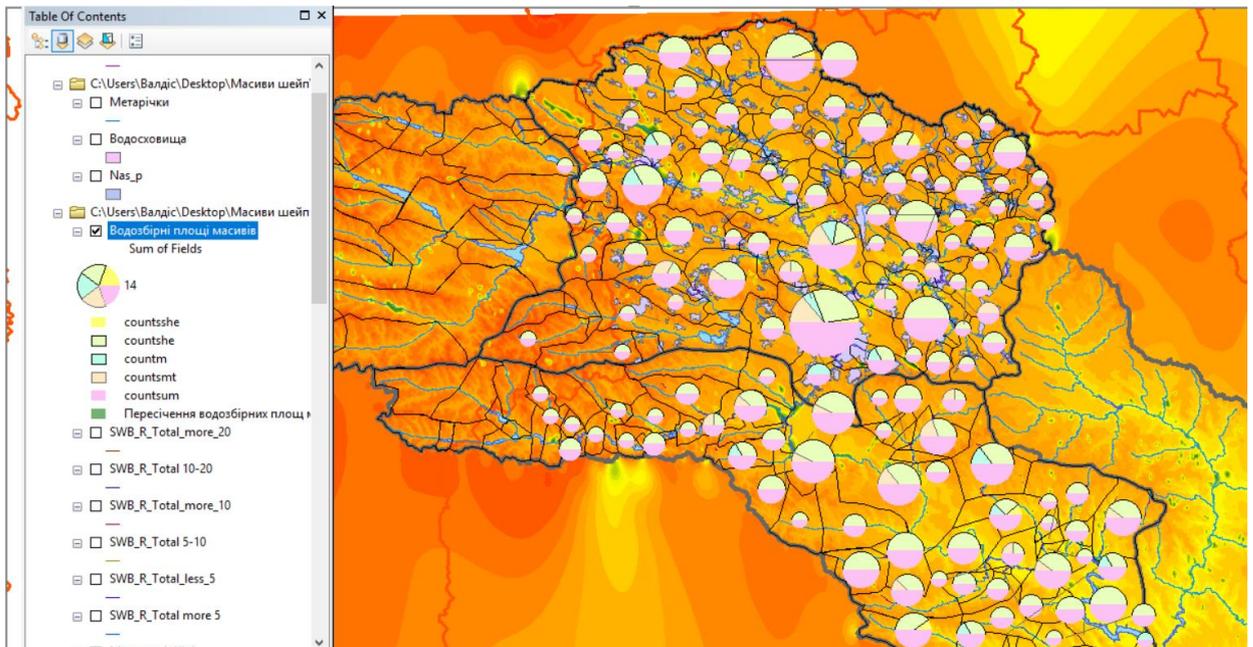


Рисунок 2.15 Тематична карта населених пунктів.

В результаті проведеної роботи ми отримуємо тематичну карту, на якій використані кругові діаграми для відображення кількості перетинів річок з різними типами населених пунктів. Кожен сегмент діаграми позначений певним кольором, що дозволяє швидко та наочно оцінити розподіл перетинів між категоріями поселень.

Рожевий колір представляє загальну кількість перетинів і займає приблизно половину круга, підкреслюючи найбільший внесок у загальну статистику.

Жовтий сегмент відповідає селищам, світло-зелений — селам, світло-оранжевий — селищам міського типу, а блакитний сегмент відображає міста. Така візуалізація допомагає швидко визначити, які типи населених пунктів найчастіше перетинаються з водними масивами, а які займають меншу частку.

За аналогічним принципом створюється тематична карта чисельності населення.

Для цього використовується стовпець “population”, що містить дані про кількість мешканців у кожному населеному пункті.

Кожна точка на карті зафарбовується залежно від чисельності населення, що дає змогу порівнювати різні території та оцінювати концентрацію населення у регіоні.

Завдяки такій візуалізації легко побачити території з високою щільністю населення та ті, де населення розподілено менш щільно, що може бути корисним для подальшого аналізу та планування управління ресурсами.

Для полегшення інтерпретації карти і зручності аналізу в ArcGIS було додано легенду, яка пояснює значення кольорових індикаторів.

Процес додавання легенди включав кілька етапів:

1. Перехід у режим компоновки: на верхній панелі обрано «Вид» (View), далі «Компонування» (Layout View);

2. Вибір інструменту додавання легенди: на панелі інструментів знайдено і вибрано «Додати легенду» (Add Legend), що відкриває майстер створення легенди;

3. Налаштування легенди: у майстрі обрано шари, які потрібно включити в легенду, зокрема ті, що містять дані про чисельність населення;

4. Форматування: встановлено стиль, шрифти, кольори, рамки та інші параметри, щоб легенда була зрозумілою та інформативною;

5. Розміщення легенди: легенду розміщено у зручному місці на карті з можливістю регулювати розмір та положення;

6. Завершення компоновки: перевірено зовнішній вигляд легенди на карті та внесено необхідні корективи.

Такий підхід забезпечує підвищення наочності та інформативності карти, оскільки використання кольорових індикаторів дає змогу користувачам без додаткових пояснень інтерпретувати зміст візуалізації. Кожен колір однозначно відповідає певній категорії або діапазону чисельності населення, що полегшує сприйняття просторової інформації. У результаті користувач може швидко порівняти населені пункти між собою та оперативно оцінити рівень концентрації населення на різних ділянках карти (рис. 2.16).

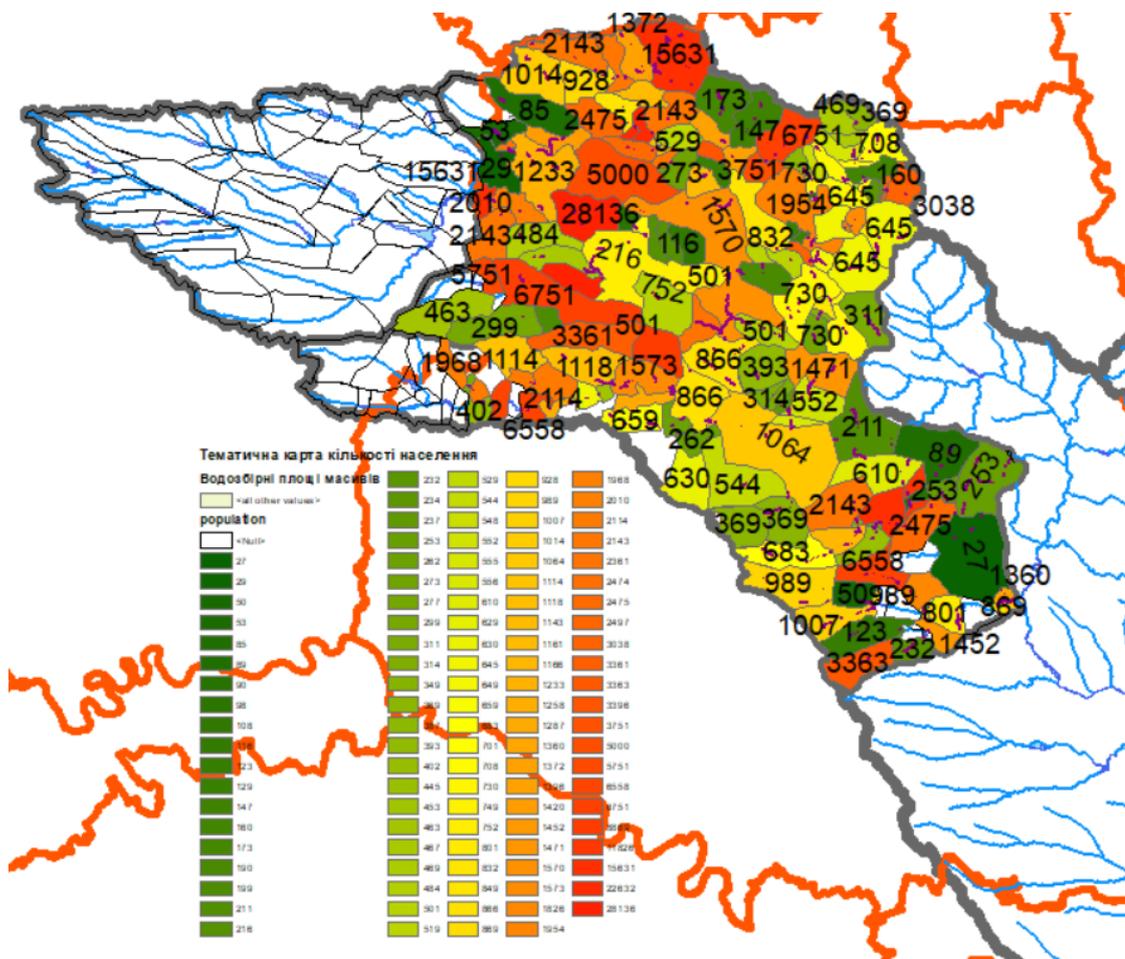


Рисунок 2.16 Тематична карта кількості населення.

Спочатку населені пункти з найменшою кількістю мешканців виділяються салатовим кольором. Цей відтінок обрано для позначення територій з низькою щільністю населення, що дозволяє швидко ідентифікувати області з невеликою чисельністю жителів. У міру збільшення кількості населення колір поступово змінюється до насиченого оранжевого, який символізує найвищу щільність населення та допомагає миттєво виділити райони з великою кількістю мешканців.

Такий спосіб візуалізації забезпечує швидке та ефективне оцінювання розподілу перетинів річок із різними типами населених пунктів за допомогою кольорового кодування. Графічне відображення даних у вигляді тематичної карти та кругових діаграм створює зручний інструмент для аналізу, де різні кольори відповідають різним категоріям поселень. Це дозволяє легко виявляти закономірності та тенденції у взаємодії річок із селищами, селами та містами,

що може стати у нагоді при плануванні управління водними ресурсами та розвитку інфраструктури населених пунктів.

2.3 Автоматизований збір, обробка та структуризація даних моніторингу з просторовою прив'язкою

Спочатку потрібно зайти на сайт «Моніторинг та екологічна оцінка водних ресурсів України», перейшовши у вкладку Звітність та обравши необхідні фільтри: період (дату «з» і «до»), басейн Південний Буг та всі доступні показники завантажити дані.

Отриманий звіт міститиме детальну інформацію про якість води на основі даних із постів спостереження.

Зокрема, буде представлено концентрації забруднюючих речовин (наприклад, нітратів, фосфатів, важких металів), рівень розчиненого кисню, показники жорсткості води, мінералізації, рН, температуру та інші важливі гідрохімічні параметри.

Такий звіт дозволить оцінити екологічний стан водойм, порівняти результати з нормативами якості води та виявити потенційно проблемні ділянки для подальшого моніторингу або вжиття коригувальних заходів (рис. 2.17).

	A	B	C	D	E	F	G	H	I	J	K	L
1	Дані моніторингу (за районом річкового басейну або суббасейну)											
2	з 01.03.2000 до 11.03.2025											
3	Район											
4	1. Пост: Балка Ташлик, 6 км, Ташлицьке вдсх., став-охолоджувач ПУ АЕС (міст через Ташлицьке вдсх., південь м. Південно-Українськ)											
5	Значення	Показник										
6		Амоній-іони, мг	Біохімічне спож	Завислі (с	Кисень рс	Нітрат-іон	Нітрит-іон	Сульфат-іс	Фосфат-іс	Хлорид-іони, мг/дм3		
7	Дата											
8	17.04.2000	0,035	1,3	0	12,98	4,7	0,038	403,2	0,143	146,1		
9	04.07.2000	0,031	1,13	10,4	9,89	0,24	0,022	416	0,06	149		
10	04.12.2000	0,042	1,2	0	9,63	2,02	0,006	384	0,275	154,5		
11	06.02.2001	0,062	2,88	1,6	11,77	4,3	0,004	364,8	0,38	160,1		
12	03.04.2001	0	2,5	15,6	11,4	11,4	0,023	397	0,197	146		
13	09.07.2001	0,05	1,98	21	12,97	0,51	0,022	384	0,002	154,5		
14	08.10.2001	0,106	1,04	0	9,63	0,86	0,007	364,8	0,061	148,9		
15	09.01.2002	0	1,77	12	12,04	3,78	0,032	317,2	0,315	168,4		

Рисунок 2.17 – Початкові дані

Отримані дані містять ключову інформацію про пости спостереження, включаючи назву, географічне розташування, дату відбору проб та значення показників якості води (температура, рН, вміст розчиненого кисню, забруднювальні речовини тощо).

Ці дані є основою для аналізу динаміки зміни екологічного стану водойм, виявлення потенційно небезпечних зон та своєчасного реагування на негативні зміни (рис. 2.18).

```

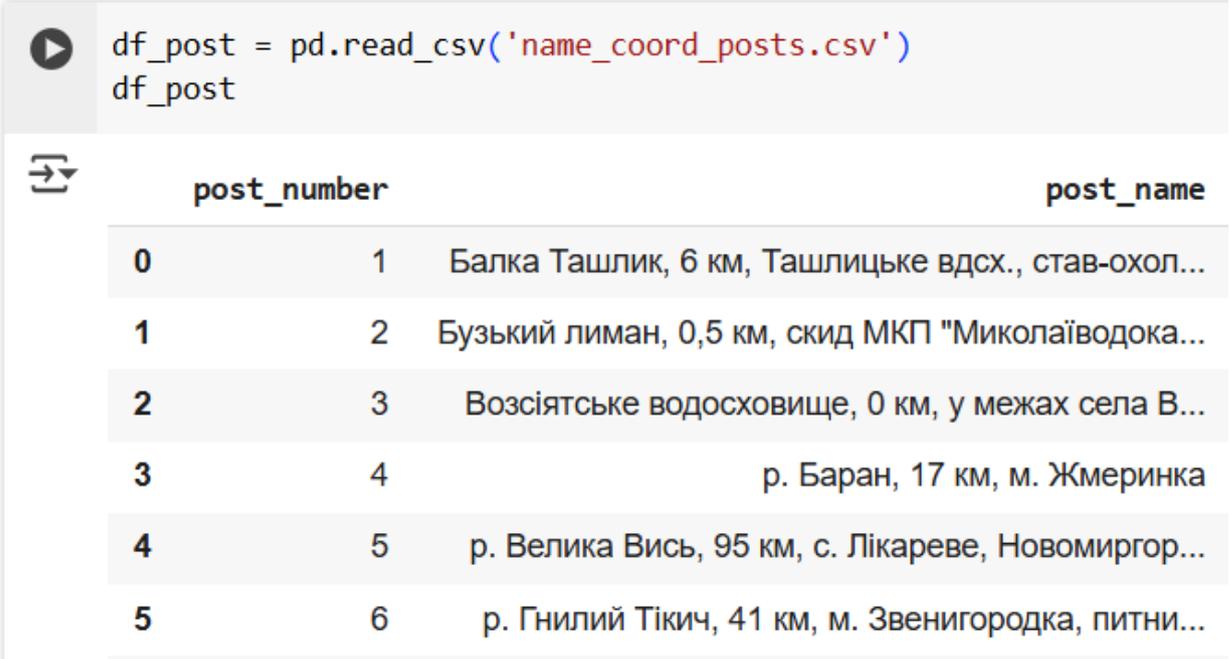
for line in lines:
    line = line.replace('""', '').strip('')
    post_match = re.match(r'(\d+)\. \s*Пост:\s*(.?.*?);*$', line)
    if post_match:
        post_number = int(post_match.group(1))
        post_name = post_match.group(2).strip() # Прибираємо зайві пробіли
        current_post = {'post_number': post_number, 'post_name': post_name}
        monitoring_posts.append(current_post)
        in_data_block = False
        headers_found = False
        continue
    if not headers_found and 'Показник' in line:
        headers_found = True
        continue
    if headers_found and not in_data_block and len(line) > 0 and ';' in line:
        indicators = [ind.strip() for ind in line.rstrip(';').split(';') if ind.strip()]
        in_data_block = True
        continue
    if in_data_block and 'Дата' in line:
        continue
    if in_data_block and len(line) > 0 and ';' in line:
        values = line.rstrip(';').split(';')
        if len(values) > 0 and values[0].strip() and re.match(r'\d{2}\.\d{2}\.\d{4}', values[0]):
            date = values[0].strip()
            row_data = {'post_number': post_number, 'date': date}
            for i, value in enumerate(values[1:], 1):
                if i <= len(indicators) and value.strip():
                    try:
                        row_data[indicators[i-1]] = float(value.replace(',', '.'))
                    except ValueError:
                        row_data[indicators[i-1]] = None
            measurements.append(row_data)

```

Рисунок 2.18 – Частина коду для парсингу даних

Спочатку текстовий файл розбивається на окремі рядки, після чого з використанням регулярних виразів визначаються записи, що містять інформацію про пости спостереження — їхні номери та назви. Далі алгоритм ідентифікує заголовки показників якості води, після чого обробляє дані вимірювань. Рядки з такими даними розділяються за допомогою символу «;», з кожного рядка фіксується дата, а числові значення конвертуються у формат

«float». Якщо значення не піддається конвертації, воно позначається як відсутнє (NaN). У результаті формуються два DataFrame: перший — з інформацією про пости спостереження, другий — з очищеними вимірюваннями, які надалі використовуються для аналітики та візуалізації (рис. 2.19).



```
df_post = pd.read_csv('name_coord_posts.csv')
df_post
```

	post_number	post_name
0	1	Балка Ташлик, 6 км, Ташлицьке вдсх., став-охол...
1	2	Бузький лиман, 0,5 км, скид МКП "Миколаїводока...
2	3	Возсіятське водосховище, 0 км, у межах села В...
3	4	р. Баран, 17 км, м. Жмеринка
4	5	р. Велика Вись, 95 км, с. Лікареве, Новомиргор...
5	6	р. Гнилий Тікич, 41 км, м. Звенигородка, питни...

Рисунок 2.19 – Назви постів

Другий файл, *measurement*, містить результати моніторингу якості води. У поданому наборі даних зібрано результати систематичних вимірювань концентрацій основних показників якості води, зокрема амонію, розчиненого кисню, нітратів, фосфатів та інших хімічних і фізико-хімічних параметрів. Для кожного пункту спостереження, ідентифікованого за допомогою показника *ID_Station*, наведено значення відповідних характеристик у конкретні моменти часу, що задаються полем *Date*. Така структура даних дає змогу аналізувати часову динаміку змін стану водного середовища, а також порівнювати показники між різними локаціями, оцінюючи просторові закономірності та тенденції у якості поверхневих вод (рис. 2.20).

```
df_data = pd.read_csv('measurement.csv')
df_data
```

	ID_Station	Date	NH4	BSK5	Suspended	O2	N03	N02	S04	P04	C1
0	1	4/17/2000	0.035	1.30	NaN	12.98	4.70	0.038	403.2	0.143	146.1
1	1	7/4/2000	0.031	1.13	10.4	9.89	0.24	0.022	416.0	0.060	149.0
2	1	12/4/2000	0.042	1.20	NaN	9.63	2.02	0.006	384.0	0.275	154.5
3	1	2/6/2001	0.062	2.88	1.6	11.77	4.30	0.004	364.8	0.380	160.1
4	1	4/3/2001	NaN	2.50	15.6	11.40	11.40	0.023	397.0	0.197	146.0

Рисунок 2.20 – Результати вимірювань якості води

Стовпці **post_number** та **ID_Station** вказують на унікальний ідентифікатор поста для моніторингу, де **post_number** є порядковим номером, а **ID_Station** — це унікальний код станції або поста, що забезпечує зв'язок між конкретними даними і постом для моніторингу. Таким чином, **ID_Station** дозволяє однозначно ідентифікувати кожен станцію або пост на основі даних у системі моніторингу водних ресурсів.

Щоб відобразити пости на карті, було додано два нові стовпці — **lat** (широта) і **lon** (довгота). Ці стовпці містять координати для кожного поста моніторингу, що дозволяє точно розташувати пости на карті.

2.4 Інтерактивна аналітика даних моніторингу з використанням ШІ

Для додавання постів моніторингу було використано файл `name_coord_posts.csv`, з якого зчитувалися координати кожного посту (рис. 2.21).

Для кожного з них створювалася геометрична точка (`Point`) із відповідними координатами. На основі цих даних формувався `GeoDataFrame` за допомогою бібліотеки `GeoPandas`.

Пости відображалися на інтерактивній карті у вигляді круглих точок, колір яких змінювався від світло-червоного до насиченого в залежності від концентрації шкідливих речовин.

```

80
81 # API endpoints with /api prefix
82 @app.get("/api/posts", response_model=List[Dict[str, Any]])
83 async def get_posts():
84     """Get list of all monitoring posts with their coordinates"""
85     result = posts_df[['ID_Station', 'Name_post', 'Location', 'Location_details', 'lat', 'lon']].rename(
86         columns={'ID_Station': 'id', 'Name_post': 'name'}).to_dict('records')
87     return result
88

```

Рисунок 2.21 – Фрагмент коду додавання постів на карту

Додатково, з файлу `measurement.csv` були зчитані дані про якість води. Для кожного посту було додано дані про рівень забрудненості на основі результатів вимірювань шкідливих речовин, що відображає рівень забруднення (рис. 2.22). Біля кожного маркера на карті додавалась анотація з назвою посту та інформацією про вміст шкідливих речовин.

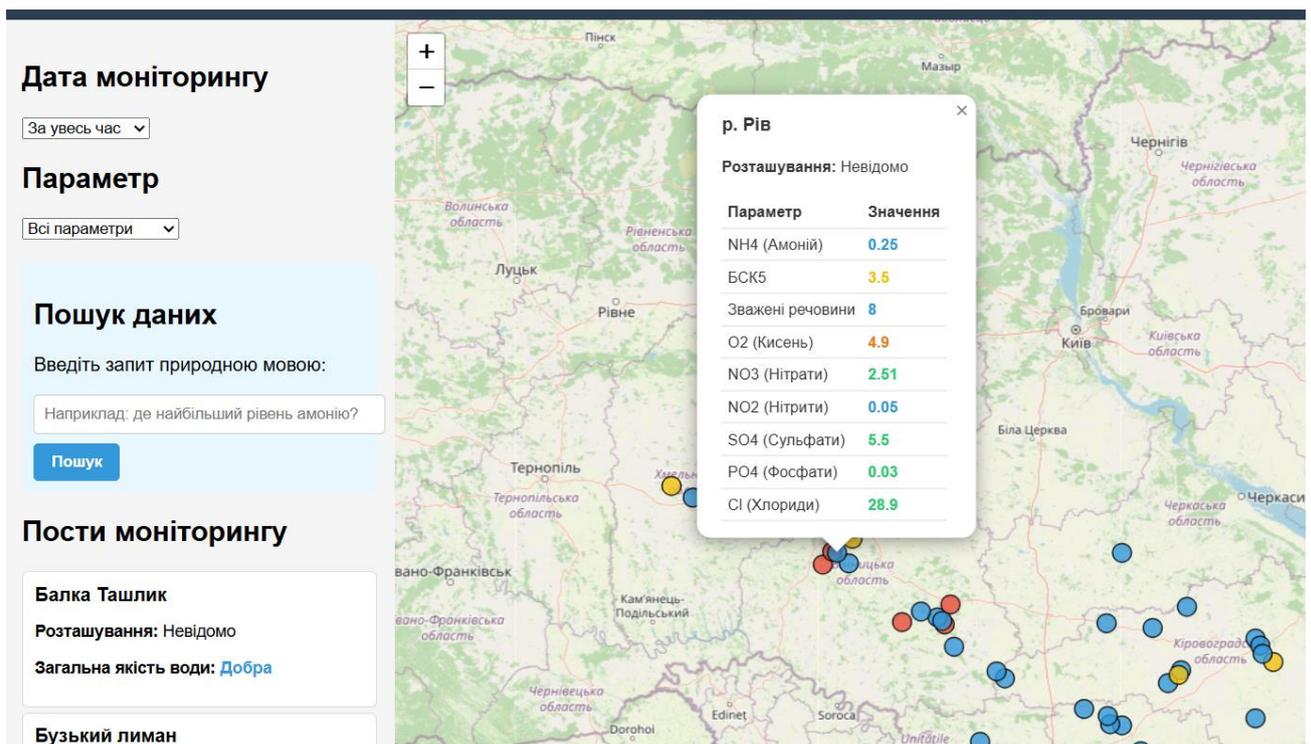


Рисунок 2.22 – Фрагмент скріншоту сайту системи моніторингу якості води

Для зручності користувача створена функція, яка виводить детальну інформацію про пости у вигляді таблиць: з ID постів, їх назвами та координатами. Це дозволяє оперативно аналізувати стан поверхневих вод у межах басейну річки Південний Буг.

Для більш зручного аналізу до інтерактивної карти були додані додаткові точки, що відображають розташування підприємств, які потенційно можуть бути джерелами забруднення. Ці об'єкти позначені окремими маркерами, що дає змогу візуально зіставити їхнє розташування з постами моніторингу та оцінити можливий вплив господарської діяльності на якість води в різних ділянках басейну (рис. 2.23).

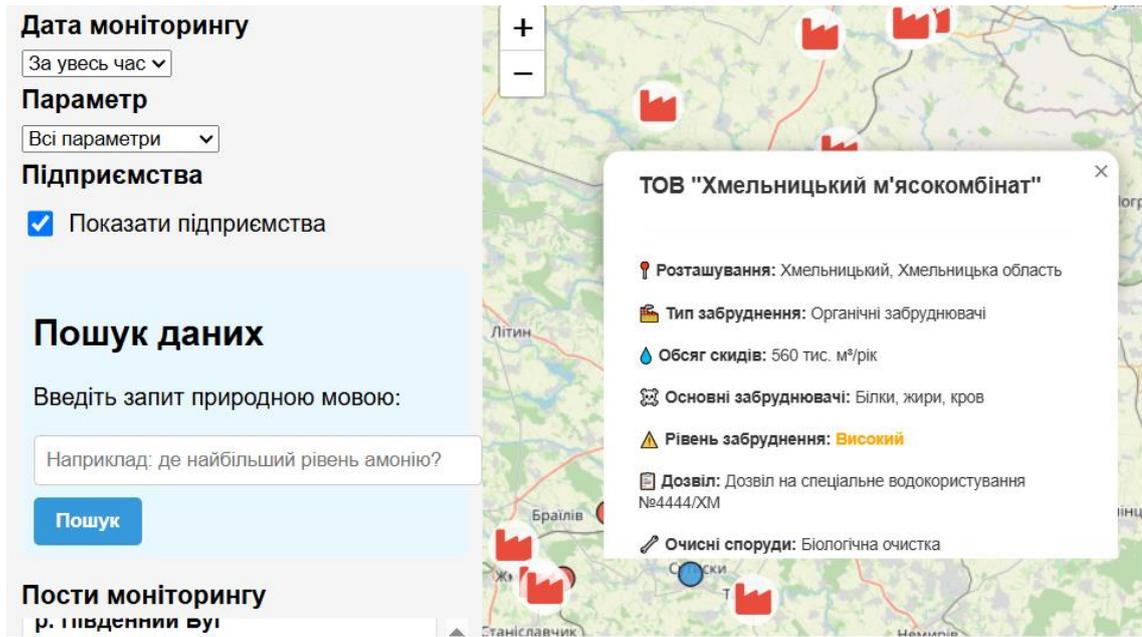


Рисунок 2.23 – Відображення підприємств забруднювачів

Для кожного поста реалізовано функцію перегляду інтерактивного графіка, який відображає зміни рівнів забрудненості за всіма показниками якості води впродовж усього періоду спостережень.

Такий підхід дозволяє користувачам здійснювати комплексний аналіз даних, оцінюючи загальні тенденції коливань показників якості води. Він сприяє своєчасному виявленню пікових і критичних значень параметрів, а також дає можливість простежити сезонні коливання й довгострокові зміни стану водного об'єкта, що є важливим для моніторингу, прогнозування та прийняття обґрунтованих управлінських рішень (рис. 2.24).

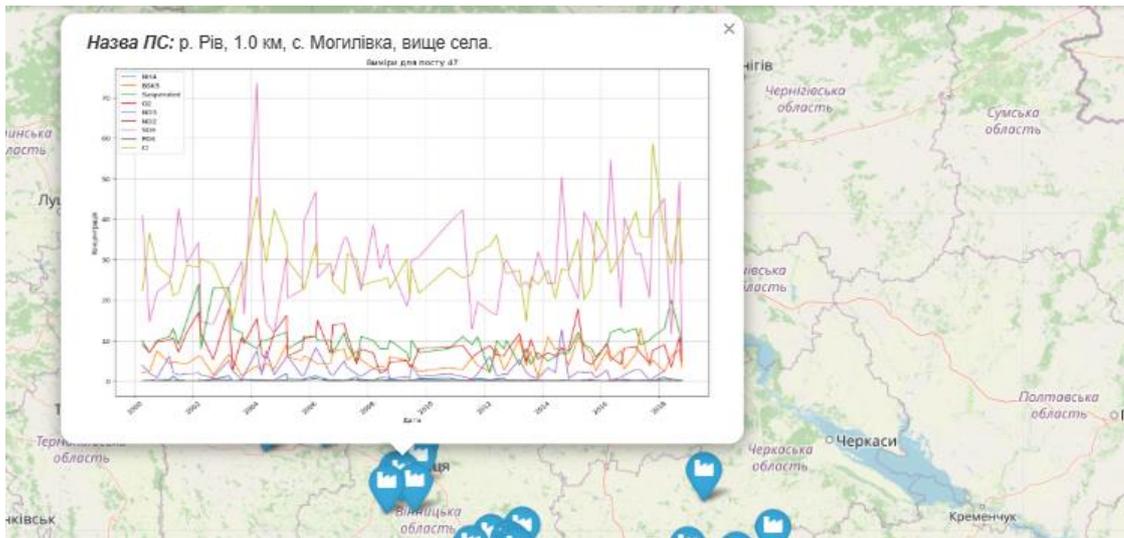


Рисунок 2.24 – Графік шкідливих вмісту речовин продовж всього часу

Цей графік допомагає візуалізувати зміни концентрації шкідливих речовин з часом, що дозволяє простежити тенденції погіршення або покращення стану води на конкретному посту. Такий аналіз сприяє більш глибокому розумінню екологічної ситуації.

Підхід до обробки даних спостережень, що використовує можливості великих мовних моделей (LLM) на базі GPT-4, бібліотеки Langchain і pandas. Це дозволяє автоматизувати аналіз великих масивів даних, спрощуючи процес отримання аналітичних висновків з неструктурованих або напівструктурованих джерел (рис. 2.25).

Запитання: Який самий південний пункт спостереження?

Відповідь: Самий південний пункт спостереження - "Бузький лиман".

Запитання: Яка протяжність ділянки спостереження

Відповідь: Максимальна протяжність ділянки спостереження становить 773 кілометри.

Запитання: Який найбільший рівень BSK5 спостерігався і коли, на якому посту?

Відповідь: Найвищий рівень BSK5, який спостерігався, становив 50.9. Це відбулося на посту з ID 41 24 листопада 2003 року.

Запитання: В якому місяці спостерігається мінімальний і максимальний рівень BSK5?

Відповідь: Мінімальний рівень BSK5 спостерігається у січні, а максимальний - у березні.

Рисунок 2.25 – Приклад роботи аналітичної системи на основі природної мови

Програмний модуль здійснює обробку двох джерел даних у форматі Excel: перше містить інформацію про координати та назви пунктів спостереження, друге — результати екологічних вимірювань показників якості води. На основі об'єднання та аналізу цих даних формується інтелектуальний агент, який забезпечує інтерактивну взаємодію з користувачем шляхом обробки природномовних запитів. Такий підхід дає змогу отримувати аналітичні результати без безпосереднього звернення до структури даних. Функціональну схему роботи відповідного ШІ-агента наведено на (рис. 2.26).

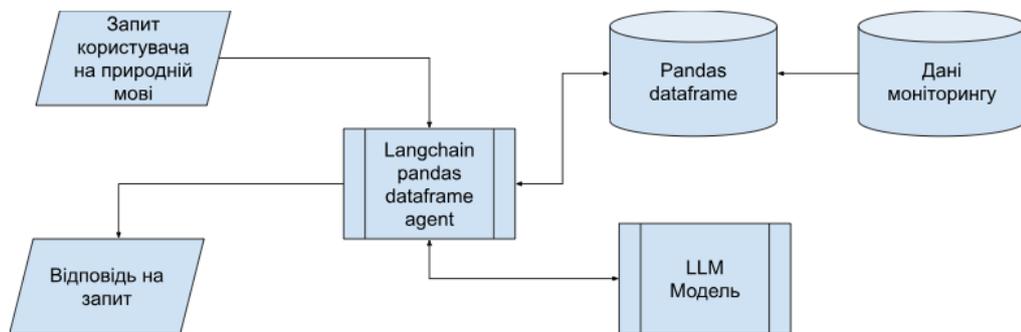


Рисунок 2.26 – Функціональна схема роботи ШІ-агента

За допомогою бібліотеки Pandas дані моніторингу зчитуються із файлів Excel та перетворюються у датафрейми. За допомогою бібліотеки Langchain створюється ШІ агент на основі цих датафреймів та великої мовної моделі GPT-4. Запит користувача надходить у вигляді опису природною мовою. LLM аналізує запит користувача та одержує структуру даних датафреймів від бібліотеки Langchain. LLM приймає рішення про те які розрахунки необхідно здійснити щоб одержати необхідні дані, ці розрахунки здійснюються безпосередньо на комп'ютері де запущено програму, без необхідності передавати великі об'єми даних в LLM. LLM одержує результати розрахунків і формує фінальну відповідь на запит користувача у вигляді природної мови.

Особливістю такого підходу є повна абстракція від необхідності володіння знаннями у сфері програмування та статистичного аналізу.

Користувачеві достатньо сформулювати запит у природній, звичній для нього мові, наприклад українською, після чого інтелектуальний агент автоматично інтерпретує зміст запиту, виконує необхідні обчислення із застосуванням бібліотеки pandas та повертає коректну й релевантну відповідь у зручній для сприйняття формі. (рис. 2.27 – 2.28).

```
# Load environment variables

load_dotenv()

# Load Excel files into pandas DataFrames

posts_df = pd.read_excel("data/name_coord_posts.xlsx")
measurements_df = pd.read_excel("data/measurement.xlsx")

# Create the agent with both dataframes

agent = create_pandas_dataframe_agent(

    ChatOpenAI(temperature=0, model="gpt-4"),

    [posts_df, measurements_df],

    verbose=False,

    agent_type=AgentType.OPENAI_FUNCTIONS,

    allow_dangerous_code=True

)
```

Рисунок 2.27 – Фрагмент коду ШІ-агента

Шаблони пошуку - система розпізнає такі типи запитів:

- Максимальні значення: ['максимум', 'найбільш', 'найвищ', 'найгірш']
- Мінімальні значення: ['мінімум', 'найменш', 'найнижч']
- Порівняння "більше": ['більше', 'вище', 'понад', 'перевищує']
- Порівняння "менше": ['менше', 'нижче', 'до', 'не перевищує']
- Якість води: ['найкращ', 'найчистіш', 'найякісніш']
- Забруднення: ['найгірш', 'найбрудніш', 'забруднен']

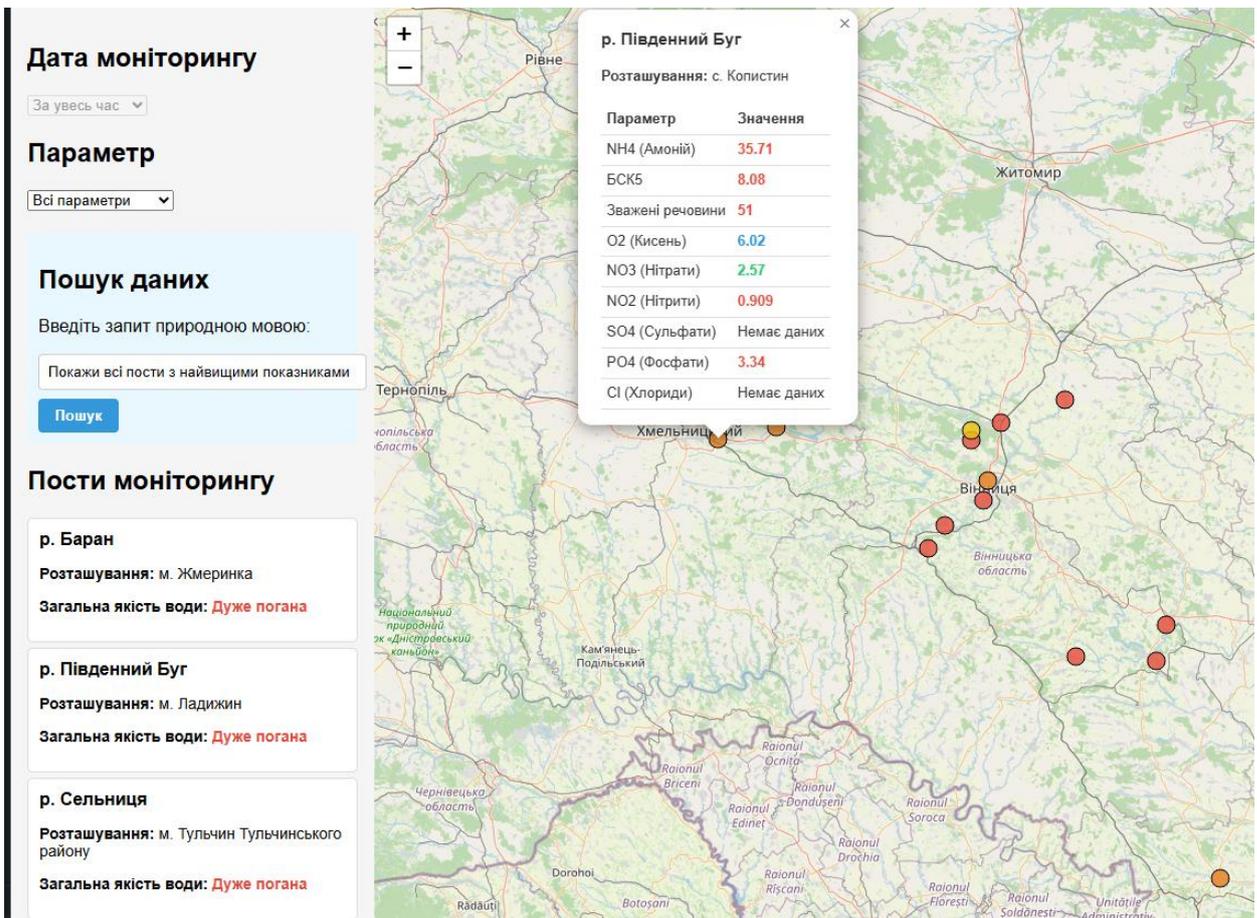


Рисунок 2.28 – Приклад запити ШІ-агента

Таким чином, ця робота демонструє ефективну інтеграцію LLM-моделі з pandas на основі бібліотеки Langchain, як середовища для високопродуктивної обробки даних [16]. Вона ілюструє перспективний напрям розвитку аналітичних систем, де ключову роль відіграє не написання коду або здійснення розрахунків, а лише формування опису даних або розрахунків які потрібні користувачу.

2.5 Розвідувальний аналіз і підготовка даних для нейронної мережі

Для виконання наступного завдання було застосовано підхід, що ґрунтується на використанні сучасних методів аналізу даних із опорою на тематичний ноутбук платформи Kaggle, у якому наведено приклад реалізації відповідної моделі та алгоритму. Обраний ноутбук використовувався як методичний орієнтир для побудови структури програмного коду, вибору

архітектури нейронної мережі та налаштування основних параметрів процесу навчання моделі [17].

У межах розвідувального аналізу даних було виконано побудову часових графіків, що відображають зміну концентрацій нітратів (NO_3) та розчиненого кисню (O_2) у водному середовищі протягом досліджуваного періоду. Така форма подання інформації дозволяє наочно простежити динаміку основних гідрохімічних показників, оцінити характер їх змін у часі та виявити можливі періодичні коливання. Зокрема, аналіз графіків дає змогу ідентифікувати сезонні закономірності, пов'язані з природними процесами, кліматичними умовами або антропогенним впливом. Крім того, подібний підхід сприяє ранньому виявленню потенційних екологічних змін і аномалій у стані водного середовища басейну річки Південний Буг, що представлено на (рис. 2.29).

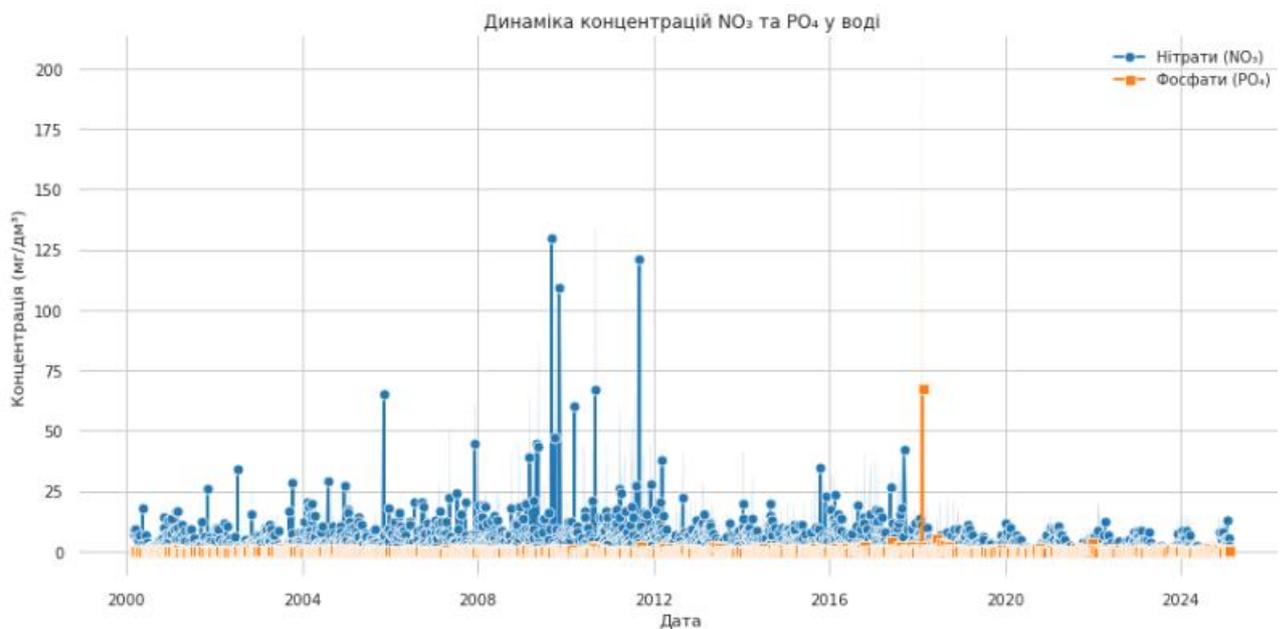


Рисунок 2.29 – Концентрація нітратів (NO_3) та розчиненого кисню (O_2) у воді

Теплова карта наочно відображає інтенсивність і напрямок взаємозв'язків між усіма числовими параметрами набору даних. Зокрема, вона дозволяє простежити, яким чином пов'язані між собою концентрації нітратів (NO_3) з показником БСК₅, рівнем розчиненого кисню (O_2) та іншими

характеристиками якості води. Такий інструмент є ефективним для виявлення прихованих закономірностей і взаємних залежностей у даних, що представлено на (рис. 2.30).

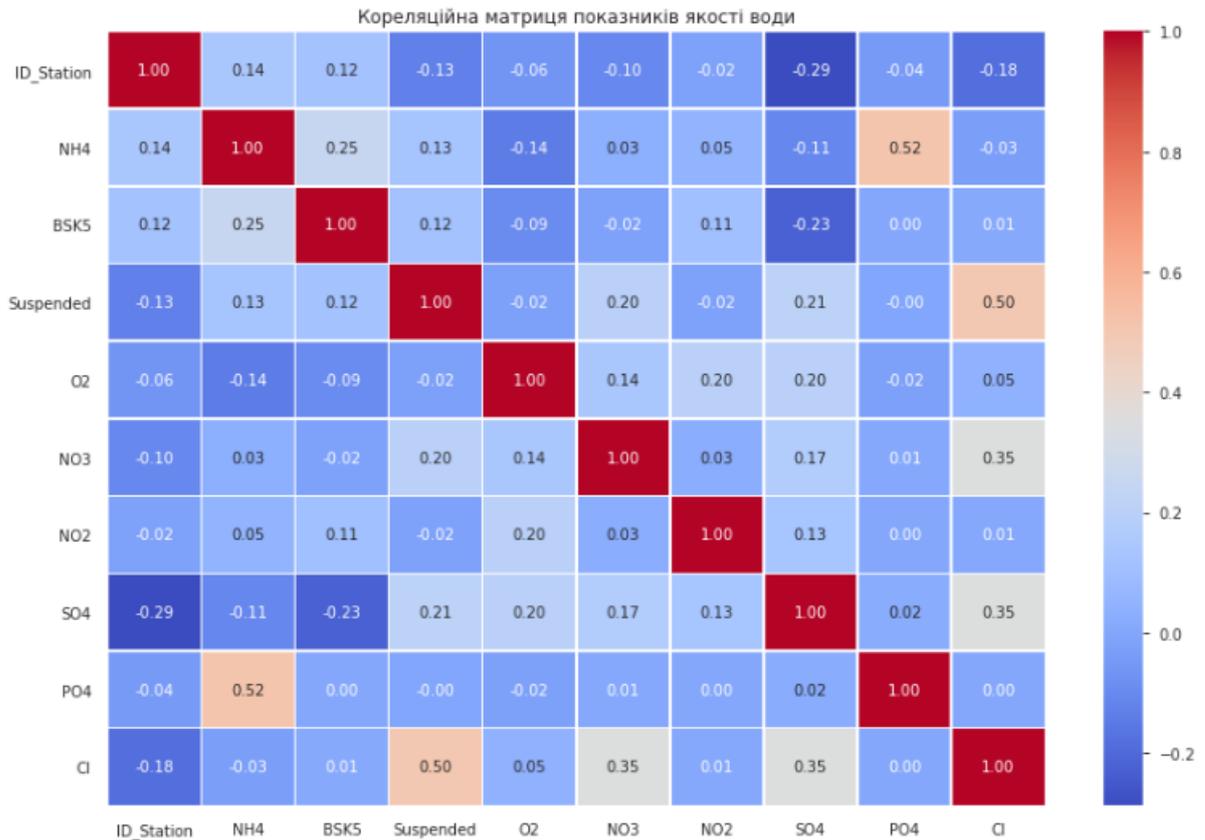


Рисунок 2.30 – Кореляційна матриця показників якості води

Побудований графік відображає середні значення показників BSK5 та Suspended для кожної станції спостереження (ID_Station), що містяться в датасеті measurement. Така форма узагальнення даних дає змогу зменшити вплив випадкових коливань окремих вимірювань і зосередитися на характерних рівнях органічного забруднення та концентрації завислих речовин для кожної точки моніторингу. Порівняння середніх значень між станціями дозволяє наочно оцінити просторові відмінності якості води, виявити ділянки з підвищеним антропогенним навантаженням або несприятливим екологічним станом. Такий візуальний аналіз сприяє швидкій ідентифікації потенційно проблемних зон у басейні річки Південний Буг та

може слугувати підґрунтям для подальшого детальнішого дослідження й прийняття управлінських рішень, що проілюстровано на (рис. 2.31).

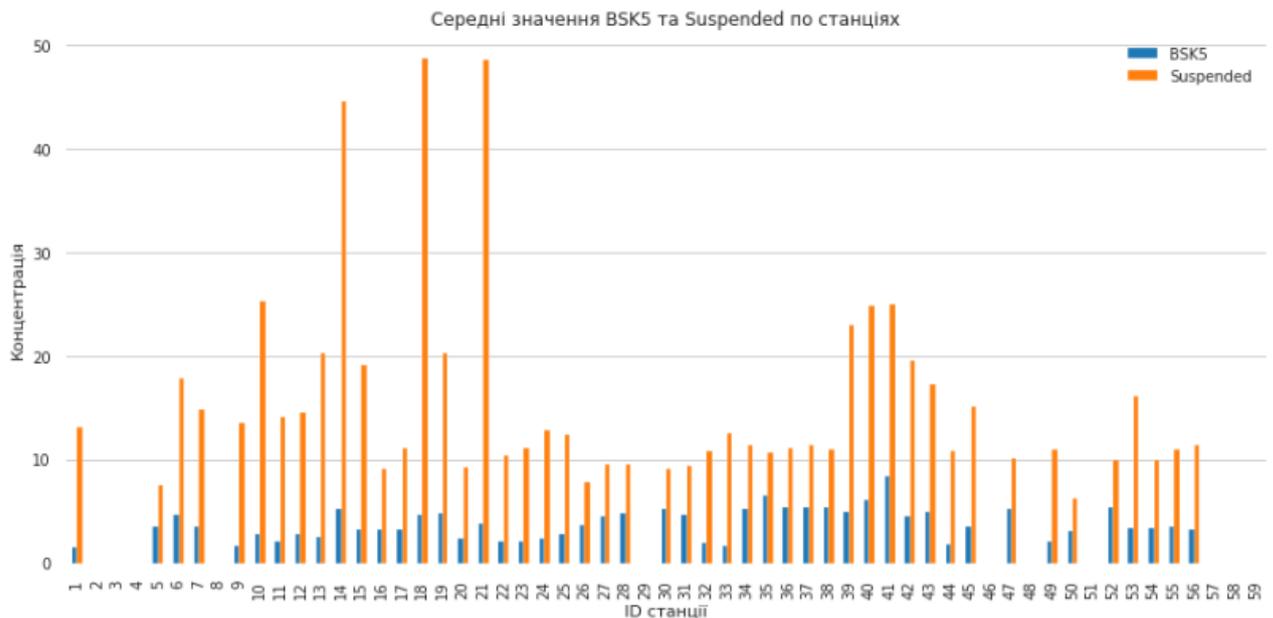


Рисунок 2.31– Середні значення параметрів BSK5 та Suspended по станціях

Графіки демонструють розподіл концентрацій ключових хімічних показників у поверхневих водах басейну річки Південний Буг. Для кожного параметра побудовано гістограми, які відображають кількість вимірювань у різних інтервалах значень. Щоб уникнути перекручування через надзвичайно великі або аномальні значення, враховано лише 95% найменших за величиною даних, тобто верхні 5% викидів відсічено.

Це дає змогу побачити типові межі концентрацій і характер розподілу, що є важливим для оцінки екологічного стану водойми та виявлення потенційних забруднень. Різні кольори відображають різні показники, що дозволяє порівнювати їхні концентрації та варіабельність у межах одного графіка на малюнку (рис. 2.32).

Такий аналіз допомагає швидко виявити характерні діапазони концентрацій речовин і можливі аномалії у воді. Це сприяє ефективному моніторингу якості води та прийняттю своєчасних управлінських рішень для збереження екосистеми.

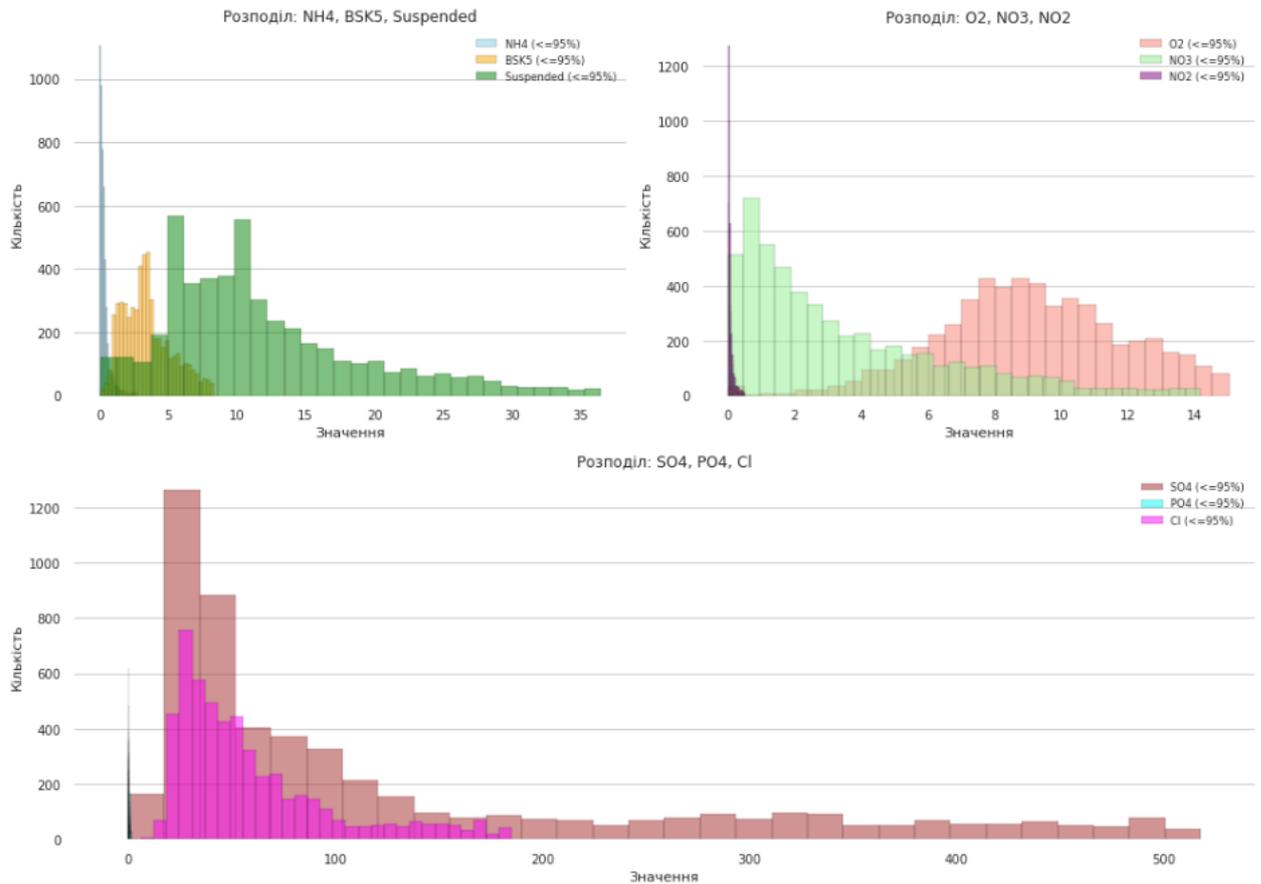


Рисунок 2.32 – Розподіл концентрацій ключових хімічних показників

На графіках представлено сезонну динаміку основних показників якості води для трьох вибраних спостережних станцій — з ідентифікаторами 37, 38 та 39. Що дає змогу порівняти зміни у концентраціях речовин протягом року без візуального перевантаження. Відображені показники включають: розчинений кисень (O_2), нітрати (NO_3), аміак (NH_4) та біохімічне споживання кисню (БСК5). Ці параметри є одними з ключових для оцінки екологічного стану водного середовища, оскільки відображають ступінь забруднення, рівень біологічної активності та кисневий режим водойми.

Аналіз здійснювався шляхом обчислення середніх значень показників по кожному місяцю. Це дозволяє виявити загальні сезонні тенденції: зростання концентрацій певних речовин у весняно-літній період або зниження в зимовий час. Вибрані станції мають найбільшу кількість повних вимірювань без пропущених значень серед усіх доступних пунктів моніторингу. Це забезпечує вищу достовірність візуалізованих даних і дозволяє зробити більш

обґрунтовані висновки щодо зміни якості води протягом року на мальюнку (рис. 2.33).

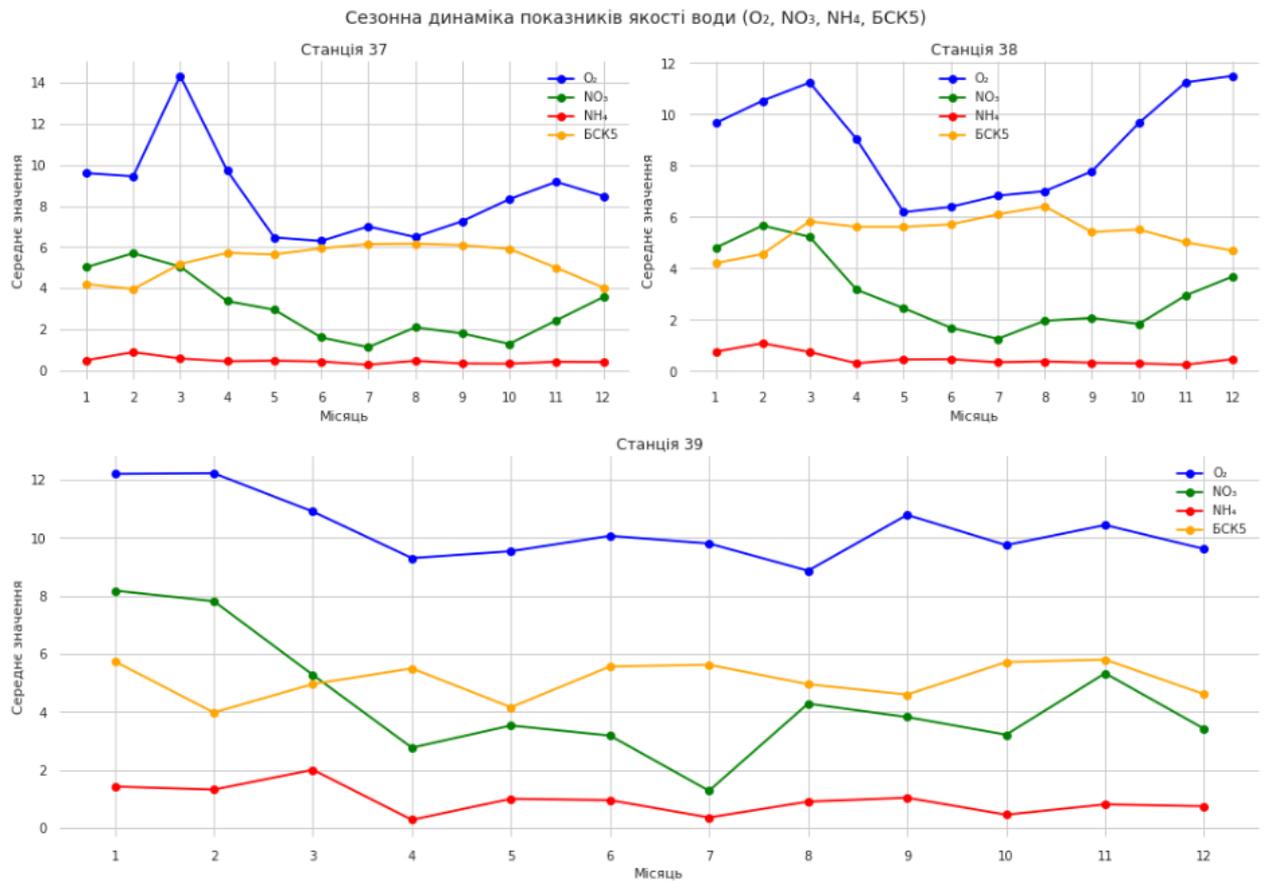


Рисунок 2.33 – Сезона динаміка показників якості води

Для підготовки даних для нейронної мережі спочатку було виконано перетворення дат у формат `datetime` та додано окремий стовпець із зазначенням місяця для кожного вимірювання. Далі було обрано моніторингові пости з ID від 32 до 39, оскільки вони містили найбільшу кількість даних із мінімальною кількістю пропусків.

Після цього виконується аналіз повноти даних. Серед усіх постів обираються ті, що мають найбільшу кількість вимірювань і найменше пропущених значень. Це дозволяє зосередитись на найбільш інформативних даних та підвищити якість навчання моделі. Для кожної обраної станції дані були відсортовані, після чого побудовано зведену таблицю, де кожен стовпець відповідає окремому посту, а значення — це рівні концентрації нітратів (NO₃).

Кількість вимірювань для всіх постів було вирівняно шляхом доповнення відсутніх значень NaN до максимальної кількості спостережень серед усіх станцій (рис. 2.34).

```
# Фільтрація станцій від 31 до 37
stations = list(range(32, 40))
filtered = measurement[measurement["ID_Station"].isin(stations)].copy()

# Створення порожнього DataFrame для результатів
max_measurements = filtered.groupby("ID_Station").size().max()
pivot_table = pd.DataFrame()

# Обробка кожної станції
for station in stations:
    # Фільтрація даних для конкретної станції
    station_data = filtered[filtered["ID_Station"] == station].copy()

    # Сортування за останнім стовпцем
    station_data = station_data.sort_values(station_data.columns[-1])

    # Створення стовпців для NH4
    station_nh4 = station_data["NO3"].tolist()
```

Рисунок 2.34 – Частина коду для обрання цільових ознак

У результаті обробки формується датафрейм, де кожен рядок відповідає конкретному моніторинговому посту у певну дату, а стовпці містять значення різних показників якості води. Така структура дозволяє використовувати її як вхід для моделі машинного навчання.

На наступному етапі підготовки дані було розділено на тренувальну та тестову вибірки у співвідношенні 50 на 50. Для цього використано функцію `train_test_split` з фіксованим параметром `random_state=0`, що забезпечує відтворюваність результатів.

Для зменшення впливу викидів у даних було проведено фільтрацію за допомогою кватильних значень. Зокрема, для кожного з числових стовпців (окрім дати) обчислено 10-й (Q10) і 90-й (Q90) перцентилі. Далі з таблиці були залишені лише ті значення, які потрапляють у межі між Q10 і Q90, тобто середні 80% даних. Це дозволяє зменшити вплив екстремальних значень і

покращити якість навчання нейронної мережі (рис. 2.35). Отриманий результат збережено в оновлену таблицю `pivot_table ()`.

```
In [5]: # Обчислення Q10 і Q90 для кожного стовпця (крім дати)
q10 = pivot_table.iloc[:, 1:].quantile(0.10)
q90 = pivot_table.iloc[:, 1:].quantile(0.90)

# Фільтрація даних: залишаємо лише значення між Q10 і Q90
filtered_data = pivot_table[
    (pivot_table.iloc[:, 1:] >= q10) & (pivot_table.iloc[:, 1:] <= q90)
]
pivot_table = filtered_data
```

Рисунок 2.35 – Частина коду для обробки даних

Із тестової вибірки було вилучено стовпець `target`, оскільки він не потрібен під час передбачення на нових (невідомих) даних.

Для тренувальної вибірки, навпаки, було виділено ознаки (всі стовпці, крім `target`) у змінну `X_train`, а значення, які потрібно передбачити — у змінну `y_train`. Ці дані далі використовуються для навчання нейронної мережі.

Для подальшої підготовки даних до навчання нейронної мережі було виконано відбір ознак. Із тестової вибірки видалено всі стовпці, окрім двох необхідних, а також залишено цільову ознаку.

Це дозволяє зменшити розмірність вхідних даних і зосередитись лише на ключових змінних, що впливають на результати моделювання.

Для ефективного навчання дані тренувальної вибірки було додатково розділено на навчальну та валідаційну підмножини у співвідношенні 70% до 30%, що забезпечує контроль якості моделі під час тренування.

Після завершення підготовки та розподілу даних можна приступати до моделювання нейронної мережі, що дозволить оцінити якість прогнозування на основі відібраних ознак і налаштованих параметрів.

2.6 Висновки

Розроблена інформаційна технологія дозволяє ефективно збирати, обробляти та аналізувати моніторингові дані якості поверхневих вод з просторовою прив'язкою. Геопросторовий аналіз та інтеграція з ГІС забезпечують візуалізацію даних на інтерактивній та тематичних картах, що дозволяє оцінювати вплив антропогенних факторів та просторові закономірності. Використання сучасних ІТ-інструментів і моделей машинного навчання забезпечує точність прогнозування та підтримку прийняття рішень у сфері управління водними ресурсами.

3 ПОБУДОВА ТА ОЦІНЮВАННЯ МОДЕЛЕЙ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПРОГНОЗУВАННЯ ЯКОСТІ ПОВЕРХНЕВИХ ВОД

3.1 Ідентифікація оптимальних параметрів нейронної мережі

Процес ідентифікації оптимальних параметрів нейронної мережі для задачі прогнозування якості поверхневих вод. Основна мета — оцінити ефективність різних моделей і вибрати найкращу за показниками точності та середньоквадратичної помилки (рис. 3.1).

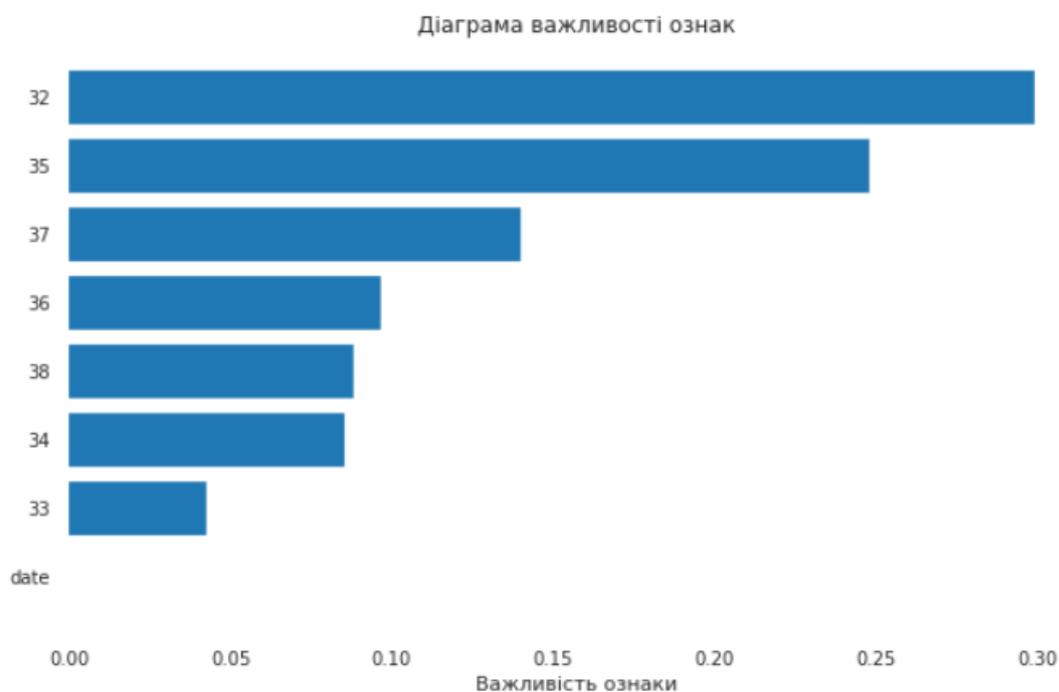


Рисунок 3.1 – Діаграма важливості ознак

Для оцінки точності прогнозів використовується коефіцієнт детермінації (R^2), який реалізовано у функції асс. Створено датафрейм result, в якому будуть зберігатися результати навчання трьох моделей: базова нейронна мережа (NN Regressor), нейронна мережа з регуляризацією Dropout (NN Regressor with Dropout) та багатошаровий перцептрон (MLP Regressor) [18]. У цьому датафреймі передбачено зберігання як коефіцієнта точності на тренувальній і валідаційній вибірках, так і середньоквадратичної помилки для

порівняння моделей (рис. 3.2). Це дозволить обрати оптимальну конфігурацію нейронної мережі для подальшого використання у прогнозуванні.

```
In [557]: # Creation the dataframe with the resulting score of all models
result = pd.DataFrame({'model' : ['NN Regressor', 'NN Regressor with Dropout', 'MLP Regressor'],
                       'train_mse': 0, 'valid_mse': 0, 'test_mse': 0})

result
```

Out[557]:

	model	train_mse	valid_mse	test_mse
0	NN Regressor	0	0	0
1	NN Regressor with Dropout	0	0	0
2	MLP Regressor	0	0	0

Рисунок 3.2 – Частина коду датафрейму для зберігання результатів

Наступним кроком була реалізація функції `build_nn()`, яка відповідає за створення та повернення моделі нейронної мережі типу `Sequential`. У межах цього підходу всі шари додаються послідовно, що робить структуру мережі зрозумілою та логічно впорядкованою.

Архітектура мережі складається з трьох повнозв'язних шарів (`Dense`) із кількістю нейронів 128, 64 та 16 відповідно. Для кожного шару застосовано активаційну функцію `ReLU`, яка забезпечує достатню виразність моделі, але водночас не перенавчається на невеликих вибірках. Така спрощена конфігурація є більш стійкою та ефективною для тренування на малих даних. Додатково на всі ці шари накладено `L2`-регуляризацію з коефіцієнтом, що допомагає запобігти перенавчанню, зменшуючи надмірну складність моделі.

Після перших трьох `Dense`-шарів застосовується `BatchNormalization`, яка стабілізує навчання та прискорює збіжність, а також шари `Dropout` із коефіцієнтами 0.3, 0.25 та 0.1. `Dropout` вимикає частину нейронів під час навчання, що дає змогу мережі узагальнювати дані краще та не «запам'ятовувати» їх.

Завершується модель одним вихідним шаром з одним нейроном та лінійною активацією. Такий підхід є оптимальним для задач регресії, де потрібно прогнозувати неперервні числові значення (рис. 3.3).

```

def build_nn():
    model = Sequential()
    model.add(Dense(1024, activation='relu', kernel_regularizer=l2(1e-4), input_shape=(train.shape[1],)))
    model.add(BatchNormalization())
    model.add(Dropout(0.3))

    model.add(Dense(512, activation='relu', kernel_regularizer=l2(1e-4)))
    model.add(BatchNormalization())
    model.add(Dropout(0.25))

    model.add(Dense(128, activation='relu', kernel_regularizer=l2(1e-4)))
    model.add(BatchNormalization())
    model.add(Dropout(0.1))

    model.add(Dense(32, activation='relu', kernel_regularizer=l2(1e-4)))

    model.add(Dense(1, activation='linear'))
    model.compile(loss='mse', optimizer=Adam(learning_rate=0.00003), metrics=['mae', 'mse'])
    return model

```

Рисунок 3.3 – Частина коду налаштування моделі NN Regressor

Модель компілюється з функцією втрат середньоквадратичної помилки (mse) і оптимізатором Adam із дуже малою швидкістю навчання (learning_rate=0.00003). Додатково вказуються метрики mae та mse [19].

Після побудови мережі виконується її тренування на тренувальній вибірці (train, target_train) з валідацією на окремій підмножині (valid, target_valid). Навчання виконується протягом максимум 1500 епох із розміром батчу 64. Для запобігання перенавчанню використано дві колбек-функції:

- EarlyStopping, що припиняє тренування при відсутності покращення протягом 250 епох, повертаючи найкращі ваги;
- ReduceLROnPlateau, яка зменшує швидкість навчання у 2 рази, якщо значення функції втрат на валідації не покращується протягом 50 епох.

У наступній моделі до архітектури додано шар Dropout, який допомагає запобігти перенавчанню, випадково "відключаючи" частину нейронів під час тренування. Модель складається з трьох шарів: два прихованих із функцією активації ReLU, один вихідний — з лінійною активацією. Для оптимізації використовується алгоритм Adam, а функцією втрат — середньоквадратична

помилка (MSE). Параметри навчання залишилися аналогічними, але механізм ReduceLRonPlateau налаштовано більш чутливо (`patience=3`, `factor=0.05`) на (рис. 3.4).

```
In [38]: %%time
def build_nn2():

    # Initializing the NN with 3 layers including 2 hidden layers and Dropout
    model = Sequential()
    model.add(Dense(units=8, activation='relu', input_shape=(train.shape[1],)))
    model.add(Dropout(0.2))
    model.add(Dense(units=6, activation='relu'))
    model.add(Dense(units=1, activation='linear'))
    model.compile(loss='mse', optimizer='adam', metrics=['mse'])

    learning_rate_reduction = ReduceLRonPlateau(monitor='val_mse',
                                                patience=3,
                                                verbose=1,
                                                factor=0.05,
                                                min_lr=0.0001)

    return model
```

Рисунок 3.4 – Частина коду для моделі NN Regressor with Dropout

Модель NN Regressor with Dropout показала точність на тренувальних даних 0.55, що свідчить про покращення результату порівняно з базовою моделлю без Dropout.

Це вказує на ефективність використання регуляризації для зменшення перенавчання. Час виконання навчання склав близько 9 секунд, що є прийнятним з огляду на обсяг і складність моделі [20].

Під час навчання моделі MLP Regressor було використано метод перебору параметрів GridSearchCV, що дозволило автоматично визначити оптимальні гіперпараметри (рис. 3.5).

Найкращі результати були досягнуті при конфігурації: один прихований шар з 50 нейронами, функція активації ReLU і Tanh, оптимізатор — Adam, адаптивна стратегія зміни швидкості навчання та максимальна кількість ітерацій — 500.

```

%%time
# MLPRegressor
mlp = MLPRegressor()
param_grid = {'hidden_layer_sizes': [(50, 30), (30, 20)],
              'solver': ['adam'],
              'activation': ['relu', 'tanh'],
              'learning_rate': ['adaptive'],
              'alpha': [0.0001, 0.0005],
              'max_iter': [500]
              }

# Training model
mlp_cv = GridSearchCV(mlp, param_grid=param_grid, cv=5, verbose=False)
mlp_cv.fit(train, target_train)
print(mlp_cv.best_params_)

```

Рисунок 3.5 – Частина коду для налаштування моделі MLPRegressor

Такий підхід дозволив досягти точності на тренувальній вибірці — 0.7, що є показником добре навченої моделі. Навчання тривало лише 2.82 секунди, що свідчить не лише про високу ефективність моделі, а й про її придатність для задач з обмеженими обчислювальними ресурсами або необхідністю швидкого прогнозування. Отримані результати підтверджують, що класична модель MLP може бути потужним інструментом навіть у порівнянні з більш складними нейронними мережами, за умови правильного налаштування гіперпараметрів.

3.2 Підготовка даних для нейронної мережі

Вибрана Після завершення етапу навчання моделей проведено прогнозування вихідної змінної на тренувальній, валідаційній та тестовій вибірках. Для цього були використані всі три архітектури: проста нейронна мережа (NN), нейронна мережа з регуляризацією Dropout (NN з Dropout) та багат шарова перцептронна модель (MLP Regressor) [21].

На графіках результатів відображено фактичні значення цільової змінної, прогнозні значення моделей, а також лінію максимально допустимого

значення (MAV). Це дозволяє візуально оцінити точність прогнозу та ступінь відхилення кожної моделі.

На графіку (рис. 3.6) зображено результати прогнозування моделей на тренувальних даних. Видно, що всі три моделі досить точно повторюють тенденцію реальних значень, проте деякі відхилення все ж присутні.

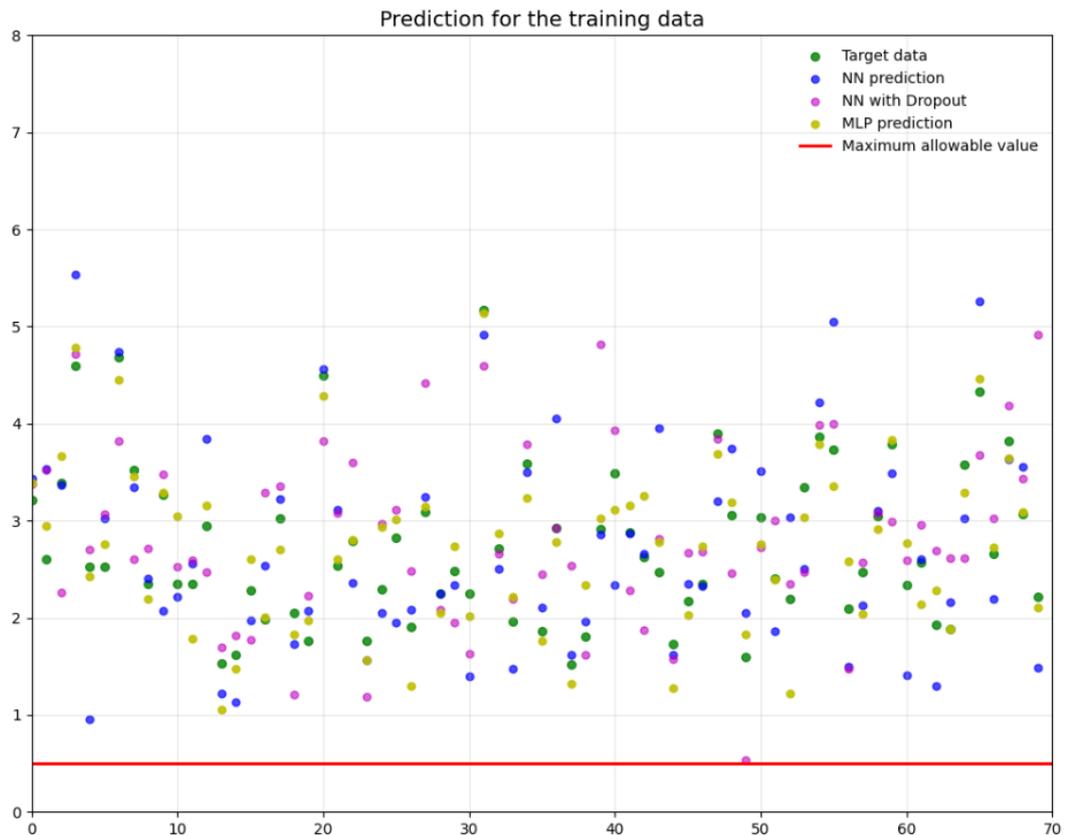


Рисунок 3.6 – Графік прогнозування тренувальних даних

MLP Regressor (жовтий) продемонструвала найменшу помилку безпосередньо на тренувальних даних ($MSE = 0.54$), проте значне погіршення результатів на тестовому наборі вказує на те, що модель схильна до перенавчання і просто "запам'ятала" вихідні приклади.

Модель NN Regressor (синій) забезпечила найбільш збалансований рівень точності: маючи показник помилки на тренуванні $MSE = 0.58$, вона досягла найкращого результату на тестових даних ($MSE = 1.85$), що свідчить про високу стабільність та відмінну здатність моделі узагальнювати нові дані.

Найменш ефективною виявилася NN Regressor with Dropout (рожевий), яка продемонструвала найвищі значення помилки (найнижчу точність) як на тренувальному, так і на тестовому наборах, що підтверджує гіпотезу про надмірний вплив регуляризації та недонавчання моделі.

Візуалізація передбачень на валідаційних даних на (рис. 3.7).

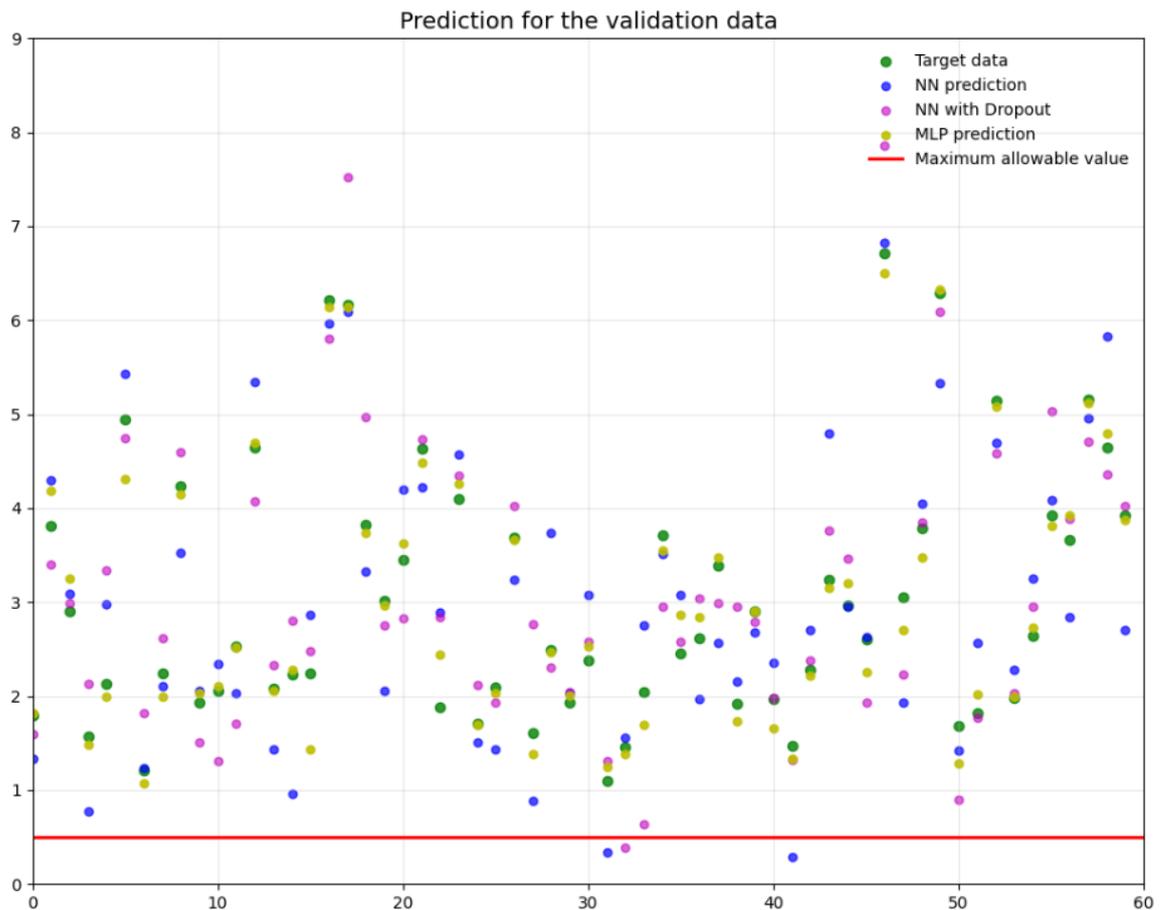


Рисунок 3.7 – Графік прогнозування валідаційних даних

Графік передбачень на валідаційному наборі демонструє відмінності в точності між трьома моделями:

- NN Regressor (синій) демонструє найкращу відповідність реальним значенням та стабільність прогнозування. Показники помилки на валідації ($MSE = 1.92$) та тесті ($MSE = 1.85$) є найнижчими серед усіх моделей, що підтверджує її високу здатність до узагальнення та надійність на нових даних.

- MLP Regressor (жовтий), незважаючи на найменшу помилку на тренувальному етапі ($MSE = 0.54$), показує значне погіршення точності на валідаційних даних ($MSE = 2.55$). Такий розрив між результатами навчання та валідації є чіткою ознакою перенавчання (*overfitting*): модель занадто підлаштувалася під тренувальну вибірку, втративши гнучкість.

- NN Regressor з Dropout (фіолетовий) демонструє найнижчу точність (найбільшу помилку) серед усіх архітектур. Високі значення помилок на валідації ($MSE = 2.81$) та тесті ($MSE = 2.92$) свідчать про те, що застосування Dropout призвело до недонавчання моделі, погіршивши її здатність відтворювати складні залежності у даних.

На графіку також присутня червона горизонтальна лінія, що позначає максимально припустиме значення ($MAV = 0.5$). Усі моделі прогнозують значення вище цього порогу, що відповідає реальному розподілу даних у цій вибірці. MAV тут виконує функцію умовного індикатора.

На графіку для тестового набору представлено передбачення трьох моделей без реальних цільових значень (оскільки вони не використовуються на етапі тестування) на (рис. 3.8).

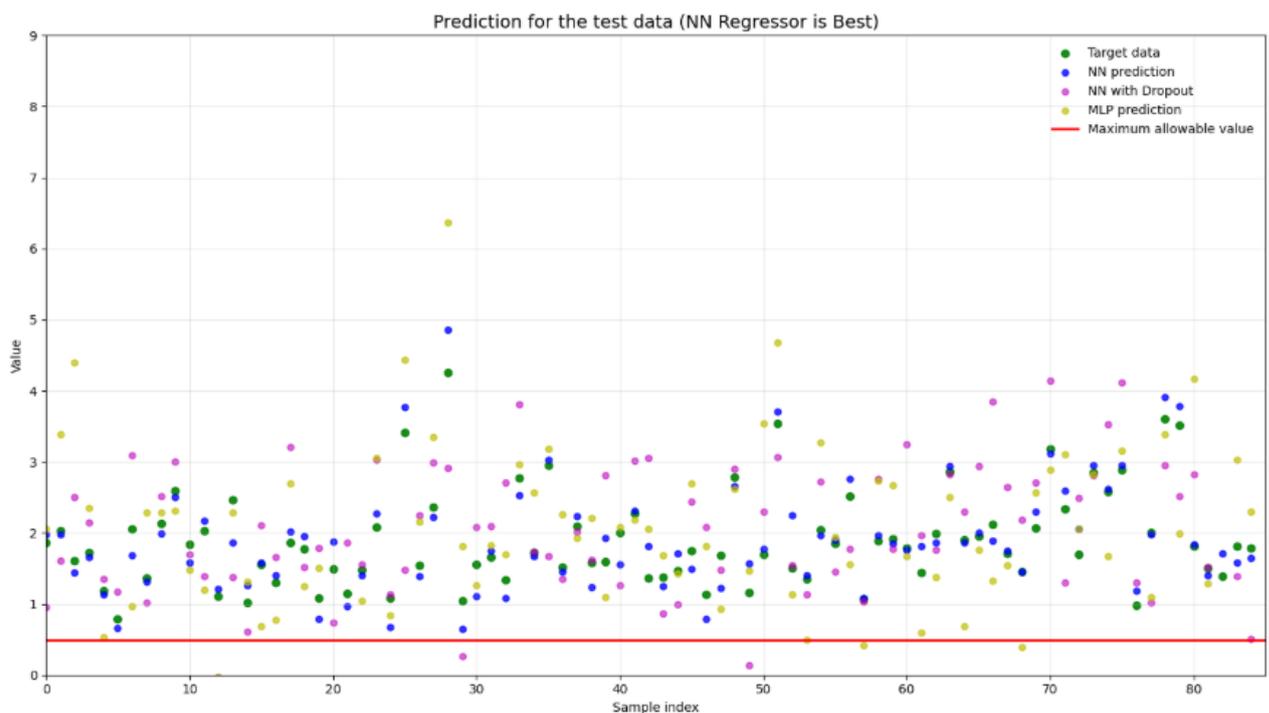


Рисунок 3.8 – Графік прогнозування тестових даних

Можна зробити такі висновки:

- NN Regressor (сині точки) демонструє найкращі результати серед усіх моделей. Передбачення моделі щільно групуються навколо реальних значень, максимально точно відтворюючи їхню динаміку. Модель показує збалансовані метрики: $MSE = 0.58$ на тренуванні та $MSE = 1.92$ на валідації, а найважливіше — найнижчу помилку на тестових даних ($MSE = 1.85$). Це свідчить про високу здатність до узагальнення та ефективність моделі на нових даних.

- MLP Regressor (жовті точки), попри низьку помилку на тренувальному етапі ($MSE = 0.54$), на тестовій вибірці показує значно гірший результат ($MSE = 2.45$). На графіку видно значний розкид точок та їхнє відхилення від цільових значень. Така розбіжність між тренувальними та тестовими показниками свідчить про перенавчання (overfitting): модель занадто "запам'ятала" тренувальні дані, втративши гнучкість для коректного прогнозування нових значень.

- NN Regressor з Dropout (фіолетові точки) демонструє найскромніші результати та характеризується найбільшою похибкою ($MSE = 2.92$ на тесті). Хаотичний характер передбачень та їх значне відхилення від реальних показників вказують на те, що застосування додаткової регуляризації (Dropout) у даній конфігурації виявилось надмірним, що призвело до недонавчання моделі. Червона лінія MAV (0.5) позначає поріг допустимих значень, і всі моделі прогнозують значення вище цього рівня. Червона лінія MAV (0.5) показує умовну межу допустимих значень.

Таким чином, NN Regressor визначено найефективнішою моделлю в цьому дослідженні, оскільки вона досягла значення середньоквадратичної помилки на тестових даних 1.85. Модель продемонструвала найкращу узагальнюючу здатність та забезпечила найвищу точність прогнозування саме на тестовому наборі даних, що є критичним критерієм для реального застосування (рис. 3.9).

model	train_mse	valid_mse	test_mse
NN Regressor	0.58	1.92	1.85
MLP Regressor	0.54	2.55	2.45
NN Regressor with Drop	0.62	2.81	2.92

Рисунок 3.9 – Таблиця навчання моделей

Натомість NN Regressor з Dropout показує найгірші результати на валідаційному наборі: середньоквадратична помилка — 2.92. Це свідчить про те, що додавання шару Dropout не покращило, а навпаки погіршило здатність моделі до узагальнення. Імовірно, через надмірну регуляризацію або невдало підібрані параметри навчання, модель стала занадто обережною — її прогнозовані значення занижені, а дисперсія зменшена, що призвело до недонавчання та втрати точності.

3.3 Висновки

У розділі проведено оцінку трьох моделей нейронних мереж для прогнозування якості поверхневих вод: NN Regressor, NN Regressor з Dropout та MLP Regressor. Найкращі результати показала базова NN Regressor, яка забезпечила найточніші передбачення із середньоквадратичною помилкою на тестових даних $MSE = 1.85$. Натомість моделі MLP Regressor ($MSE = 2.45$) та NN Regressor з Dropout ($MSE = 2.92$) продемонстрували гіршу ефективність через ознаки перенавчання або недонавчання відповідно. Отримані результати підтверджують доцільність використання саме NN Regressor у системах моніторингу та прогнозування якості води.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Оцінювання комерційного потенціалу розробки

Метою проведення комерційного та технологічного аудиту є оцінювання комерційного потенціалу інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Бу, а саме комерційного потенціалу програми, яка вибирає кращі місця для створення нових пунктів моніторингу.

Для проведення технологічного аудиту було залучено 3-х незалежних експертів Вінницького національного технічного університету кафедри системного аналізу та інформаційних технологій: к.т.н., доц. Козачко О.М., к.т.н., доц. Крижановський Є. М., к.т.н., доц. Варчук І. В. Для проведення технологічного аудиту було використано таблицю 4.1 [22] в якій за п'ятибальною шкалою використовуючи 12 критеріїв здійснено оцінку комерційного потенціалу.

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві

Продовження таблиці 4.1

Кри- терій	0	1	2	3	4
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Таблиця 4.2 – Рівні комерційного потенціалу розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0-10	Низький
11-20	Нижче середнього
21-30	Середній
31-40	Вище середнього
41-48	Високий

В таблиці 4.3 наведено результати оцінювання експертами комерційного потенціалу розробки, що відображають рівень її привабливості для практичного впровадження, ринкового використання та подальшої комерціалізації. Представлені дані характеризують узагальнену думку фахівців щодо доцільності інвестування, перспектив масштабування та конкурентоспроможності запропонованого рішення в умовах сучасного ринку інформаційних технологій.

Таблиця 4.3 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	Штельмах І.М.	Крижановський Є.М.	Варчук І.В.
	Бали, виставлені експертами:		
1	4	3	4
2	3	3	3
3	3	4	4
4	4	4	3
5	3	3	4
6	2	3	3
7	3	2	2
8	4	4	3
9	2	3	2
10	3	4	3
11	4	4	4
12	3	3	3
Сума балів	СБ ₁ =38	СБ ₂ =40	СБ ₃ =38
Середньоарифметич на сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = \frac{38 + 40 + 38}{3} = 38,7$		

Середньоарифметична сума балів, розрахована на основі висновків експертів склала 38,7 бали, що згідно таблиці 4.2 вважається, що рівень комерційного потенціалу проведених досліджень є вище середнього.

Інформаційна технологія комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг базується на даних моніторингу з використанням адаптивного парсингу, геоінформаційних технологій і великих мовних моделей для аналітичної обробки результатів. Запропоноване рішення має практичну цінність та спрямоване на підвищення ефективності моніторингу, обґрунтування управлінських рішень і вдосконалення системи охорони водних ресурсів регіону.

4.2 Прогнозування витрат на виконання науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи групуються за такими статтями: витрати на оплату праці, витрати на соціальні заходи,

матеріали, паливо та енергія для науково-виробничих цілей, витрати на службові відрядження, програмне забезпечення для наукових робіт, інші витрати, накладні витрати.

1. Основна заробітна плата кожного із дослідників Z_0 , якщо вони працюють в наукових установах бюджетної сфери визначається за формулою:

$$Z_0 = \frac{M}{T_p} * t \text{ (грн)}, \quad (4.1)$$

де M – місячний посадовий оклад конкретного розробника (інженера, дослідника, науковця тощо), грн.;

T_p – число робочих днів в місяці; приблизно $T_p \approx 21...23$ дні;

t – число робочих днів роботи дослідника.

Для розробки програмні засоби необхідно залучити програміста з посадовим окладом 9000 грн. Кількість робочих днів у місяці складає 40, а кількість робочих днів програміста складає 22. Зведемо сумарні розрахунки до таблиця 4.4.

Таблиця 4.4 – Заробітна плата дослідника в науковій установі бюджетної сфери

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату грн.
Керівник	18500	840,9	5	3864
Програмний інженер	11000	500	35	17500
Всього				22545

2. Розрахунок додаткової заробітної плати робітників

Додаткова заробітна плата Z_d всіх розробників та робітників, які приймали участь в розробці нового технічного рішення розраховується як 10 - 12 % від основної заробітної плати робітників.

На даному підприємстві додаткова заробітна плата начисляється в розмірі 10% від основної заробітної плати.

$$Z_d = (Z_o + Z_p) * \frac{H_{\text{дод}}}{100\%} \quad (4.2)$$

$$Z_d = 0,11 * 22545 = 2480 \text{ (грн).}$$

3. Нарахування на заробітну плату $H_{3П}$ дослідників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою:

$$H_{3П} = (Z_o + Z_d) * \frac{\beta}{100}, \quad (4.3)$$

де Z_o – основна заробітна плата розробників, грн.;

Z_d – додаткова заробітна плата всіх розробників та робітників, грн.;

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування, % .

Дана діяльність відноситься до бюджетної сфери, тому ставка єдиного внеску на загальнообов'язкове державне соціальне страхування буде складати 22%, тоді:

$$H_{3П} = (22545 + 2480) * \frac{22}{100} = 5505,5 \text{ (грн).}$$

4. Витрати на комплектуючі вироби до таблиця 4.4., які використовують при виготовленні одиниці продукції, розраховуються, згідно їх номенклатури, за формулою:

$$K = \sum_{i=1}^n N_i * C_i * K_i , \quad (4.5)$$

де N_i – кількість комплектуючих i -го виду, шт.;

C_i – покупна ціна комплектуючих i -го найменування, грн.;

K_i – коефіцієнт транспортних витрат (1,1...1,15).

Таблиця 4.5 – Комплектуючі, що використані на розробку

Найменування матеріалу	Ціна за одиницю, грн.	Витрачено	Вартість витраченого матеріалу, грн.
Папір	170	1	170
Олівець	10	2	20
Маркер	25	1	25
Флешка	150	1	150
Всього			365
З врахуванням коефіцієнта транспортування			389

5. Програмне забезпечення для наукової роботи включає витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення необхідного для проведення дослідження.

Для написання магістерської роботи використовувалися інтернет середовище Kaggle та ArcGIS online, які є безкоштовними.

6. Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час виконання даного етапу роботи

Дані відрахування розраховують по кожному виду обладнання, приміщенням тощо.

$$A = \frac{C * T}{T_{кор} * 12} , \quad (4.6)$$

де C – балансова вартість даного виду обладнання (приміщень), грн.;

$T_{кор}$ – час користування;

T – термін використання обладнання (приміщень), цілі місяці.

Згідно пункта 137.3.3 Податкового кодекса амортизація нараховується на основні засоби вартістю понад 2500 грн. В нашому випадку для написання магістерської роботи використовувався персональний комп'ютер вартістю 11000 грн.

$$A = \frac{11000 \cdot 1}{2 \cdot 12} = 458,33 \text{ (грн)}.$$

7. До статті «Паливо та енергія для науково-виробничих цілей» відносяться витрати на всі види палива й енергії, що безпосередньо використовуються з технологічною метою на проведення досліджень.

$$B_e = \sum_{i=1}^n \frac{W_{yt} \cdot t_i \cdot C_e \cdot K_{впi}}{\eta_i}, \quad (4.7)$$

де W_{yt} – встановлена потужність обладнання на певному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн;

$K_{впi}$ – коефіцієнт, що враховує використання потужності, $K_{впi} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

Для написання магістерської роботи використовується персональний комп'ютер для якого розрахуємо витрати на електроенергію.

$$B_e = \frac{0,3 \cdot 185 \cdot 4,32 \cdot 0,5}{0,8} = 149,85 \text{ (грн)}.$$

Витрати на службові відрядження, витрати на роботи, які виконують сторонні підприємства, установи, організації та інші витрати в нашому дослідженні не враховуються оскільки їх не було.

Накладні (загальновиробничі) витрати $B_{нзв}$ охоплюють: витрати на управління організацією, оплата службових відряджень, витрати на

утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Накладні (загальновиробничі) витрати $V_{\text{НЗВ}}$ можна прийняти як $(100\dots150)\%$ від суми основної заробітної плати розробників та робітників, які виконували дану МКНР, тобто:

$$V_{\text{НЗВ}} = (Z_o + Z_p) \cdot \frac{H_{\text{НЗВ}}}{100\%}, \quad (4.8)$$

де $H_{\text{НЗВ}}$ – норма нарахування за статтею «Інші витрати».

$$V_{\text{НЗВ}} = 22545 \cdot \frac{100}{100\%} = 22545 \text{ (грн)}.$$

Сума всіх попередніх статей витрат дає витрати, які безпосередньо стосуються даного розділу МКНР

$$V = 22545 + 2480 + 5505,5 + 389 + 458,33 + 149,85 + 22545 = 54\,072,68 \text{ (грн)}.$$

Прогнозування загальних втрат ZB на виконання та впровадження результатів виконаної МКНР здійснюється за формулою:

$$ZB = \frac{V}{\eta}, \quad (4.9)$$

де η – коефіцієнт, який характеризує стадію виконання даної НДР.

Оскільки, робота знаходиться на стадії науково-дослідних робіт, то коефіцієнт $\beta = 0,9$.

Звідси:

$$ZB = \frac{49185,6}{0,9} = 60080,75 \text{ (грн)}.$$

4.3 Розрахунок економічної ефективності науково-технічної розробки

У даному підрозділі кількісно спрогнозуємо, яку вигоду, зиск можна отримати у майбутньому від впровадження результатів виконаної наукової роботи. Розрахуємо збільшення чистого прибутку підприємства $\Delta\Pi_i$, для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, за формулою

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_0 * N * \Pi_0 * \Delta N)_i * \lambda * \rho * \left(1 - \frac{v}{100}\right), \quad (4.10)$$

де $\Delta\Pi_0$ – покращення основного оціночного показника від впровадження результатів розробки у даному році.

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки:

Π_0 – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки:

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0,8333$.

ρ – коефіцієнт, який враховує рентабельність продукту. $\rho = 0,25$;

v – ставка податку на прибуток. У 2025 році – 18%.

Припустимо, що ціна за програмний продукт зростає на 600 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року на 50 шт., протягом другого року – на 40 шт., протягом третього року на 30 шт. Реалізація продукції до впровадження розробки складала 2 шт., а її ціна до

складає 8500 грн. Розрахуємо прибуток, яке отримає підприємство протягом трьох років.

$$\begin{aligned}\Delta\Pi_1 &= [600 \cdot 2 + (8500 + 600) \cdot 50] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 111942,5 \text{ (грн)}.\end{aligned}$$

$$\begin{aligned}\Delta\Pi_2 &= [600 \cdot 2 + 9100 \cdot (50 + 40)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 201307,5 \text{ (грн)}.\end{aligned}$$

$$\begin{aligned}\Delta\Pi_3 &= [600 \cdot 2 + (9000 + 600) \cdot (50 + 40 + 30)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 268416,4 \text{ (грн)}.\end{aligned}$$

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Розрахуємо основні показники, які визначають доцільність фінансування наукової розробки певним інвестором, є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

Розрахуємо величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки.

$$PV = k_{\text{інв}} \cdot ЗВ, \quad (4.11)$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо ($k_{\text{інв}} = 2 \dots 5$).

$$PV = 2 \cdot 60080,75 = 120161,51 \text{ (грн).}$$

Розрахуємо абсолютну ефективність вкладених інвестицій E_{abc} згідно наступної формули:

$$E_{abc} = (ПП - PV), \quad (4.12)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн.;

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.13)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДЦКР, грн.;

T – період часу, протягом якою виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,2;

t – період часу (в роках).

$$ПП = \frac{111942,5}{(1 + 0,2)^1} + \frac{201307,5}{(1 + 0,2)^2} + \frac{268416,4}{(1 + 0,2)^3} = 388386,0 \text{ (грн).}$$

$$E_{abc} = (388386,0 - 120161,51) = 268224,49 \text{ (грн).}$$

Оскільки $E_{abc} > 0$, то вкладання коштів на виконання та впровадження результатів НДДКР може бути доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_B . Для цього користуються формулою:

$$E_B = \sqrt[T_{ж}]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.14)$$

де $T_{ж}$ – життєвий цикл наукової розробки, роки.

$$E_B = \sqrt[3]{1 + \frac{268224,49}{120161,51}} - 1 = 0,47 = 47\%.$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (4.15)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2025 році в Україні $d = (0,14...0,2)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05...0,1)$.

$$\tau_{min} = 0,18 + 0,5 = 0,23.$$

Так як $E_B > \tau_{min}$ то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_B}. \quad (4.16)$$

$$T_{ок} = \frac{1}{0,47} = 2,1 \text{ (роки)}.$$

Так як $T_{ок} \leq 3...5$ -ти років, то фінансування даної наукової розробки в принципі є доцільним.

4.5 Висновки

Проведено оцінку комерційного потенціалу інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг, а саме програми, яка вибирає кращі місця для створення нових пунктів моніторингу за даними громадського моніторингу, який є на вище середньому рівні.

Прогнозування витрат на виконання науково-дослідної роботи по кожній з статей витрат складе 54072,68 грн. Загальна ж величина витрат на виконання та впровадження результатів даної НДР буде складати 60080,75 грн.

Вкладені інвестиції в даний проект окупляться через 2,1 роки, приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки склала 388386,0 грн.

ВИСНОВКИ

Сучасне управління водними ресурсами потребує всебічного аналізу стану водних об'єктів, оскільки забруднення негативно впливає на екосистему та здоров'я людей. Застосування сучасних інформаційних технологій, таких як адаптивний парсинг, ГІС та великі мовні моделі, дозволяє інтегрувати дані, досліджувати геопросторові взаємозв'язки та прогнозувати зміни якості води, що підвищує ефективність управлінських рішень і зменшує антропогенний вплив на басейн річки Південний Буг.

У першому розділі здійснено аналіз басейну річки Південний Буг, оцінено стан водних ресурсів, системи моніторингу та методи прогнозування забруднення з використанням ГІС, моделювання і ШІ, що дозволяє визначити шляхи підвищення ефективності управління якістю води.

У другому розділі здійснено розробку інформаційної технології яка забезпечує збір, обробку та аналіз просторово прив'язаних даних якості вод, дозволяє візуалізувати їх на інтерактивній та тематичних картах і підвищує точність прогнозування та ефективність управління водними ресурсами.

У третьому розділі здійснено оцінювання трьох моделей нейронних мереж для прогнозування якості поверхневих вод: NN Regressor, NN Regressor з Dropout та MLP Regressor. Найкращі результати показала базова NN Regressor, яка забезпечила найточніші передбачення із середньоквадратичною помилкою на тестових даних $MSE = 1.85$. Натомість моделі MLP Regressor ($MSE = 2.45$) та NN Regressor з Dropout ($MSE = 2.92$) продемонстрували гіршу ефективність через ознаки перенавчання або недонавчання відповідно. Отримані результати підтверджують доцільність використання саме NN Regressor у системах моніторингу та прогнозування якості води.

У результаті дослідження здійснено комплексний аналіз стану басейну річки Південний Буг, оцінено сучасні методи моніторингу та прогнозування якості води з використанням ГІС, тематичних карт та нейронних мереж.

Виявлено ефективність базової NN Regressor для передбачення змін якості води та обґрунтовано доцільність інтеграції сучасних інформаційних технологій у систему управління водними ресурсами. Загалом отримані результати формують основу для підвищення точності прогнозів, оптимізації моніторингу та зменшення негативного антропогенного впливу на екосистему річкового басейну.

У четвертому розділі проведено оцінку комерційного потенціалу інформаційної технології комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг. Прогнозування витрат на виконання науково-дослідної роботи по кожній з статей витрат складе 54 072,6 грн. Загальна ж величина витрат на виконання та впровадження результатів даної НДР буде складати 60080,75 грн.

Вкладені інвестиції в даний проект окупляться через 2,1 роки, приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки склала 388386,0 грн.

За даною темою дослідження зроблено доповідь на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного технічного університету (2024-2025) [1], а також опубліковано дві фахових статті у «Вісник Вінницького політехнічного інституту» (2024-2025) [2-3].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Караваєв В. О., Крижановський Є. М., Штельмах І. М. Інформаційна технологія аналізу даних моніторингу стану поверхневих вод басейну р. Південний Буг. *LIIV Всеукраїнська науково-технічна конференція підрозділів Вінницького національного технічного університету (2024-2025)*. Вінниця, 2024-2025. URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa/2025/paper/view/23865/19666> (дата звернення 24.03.2025).
2. Крижановський Є. М., Караваєв В. О., Штельмах І. М., Войцеховська О. О. Автоматизація використання природномовних запитів для комплексного аналізу стану поверхневих вод басейну Південного Бугу // *Вісник Вінницького політехнічного інституту*. – 2025. – № 4. – С. 118–125. – DOI: <https://doi.org/10.31649/1997-9266-2025-181-4-118-125>.
3. Мокін В. Б., Бондалетов К. О., Крижановський Є. М., Караваєв В. О. Метод аугментації текстів про стан масивів вод на основі інтелектуальної прив'язки до багатозв'язних геоінформаційних систем іменованих сутностей. *Вісник Вінницького політехнічного інституту*. 2024. № 3. С. 55–65. DOI: <https://doi.org/10.31649/1997-9266-2023-168-3-55-65>.
4. Водні ресурси України : монографія / за ред. О. І. Олійника. Київ : Наукова думка, 2015. 320 с.
5. Мокін В. Б., Гораш М. А., Крижановський Є. М., Вуж Т. Є. Інформаційна інтелектуальна технологія автоматизованої геоприв'язки екологічної текстової природно-мовної інформації. *Наукові праці Вінницького національного технічного університету*. 2020. № 4. URL: <https://praci.vntu.edu.ua/index.php/praci/article/view/624>
6. Directive 2000/60/EC of the European Parliament and of the Council. EUR-Lex – Access to European Union Law. URL: https://eur-lex.europa.eu/resource.html?uri=cellar:5c835afb-2ec6-4577-bdf8-756d3d694eeb.0004.02/DOC_1&format=PDF (дата звернення: 07.06.2023).

7. Водний кодекс України : Кодекс України від 06.06.1995 р. № 213/95-ВР. Станом на 19 серп. 2022 р. URL: <https://zakon.rada.gov.ua/laws/show/213/95-вр#Text> (дата звернення: 07.06.2023).
8. Про затвердження Порядку розроблення плану управління річковим басейном : Постанова Кабінету Міністрів України від 18.05.2017 р. № 336. URL: <https://www.kmu.gov.ua/npas/249999756> (дата звернення: 04.06.2023).
9. Water Information System for Europe (WISE). URL: <https://www.eea.europa.eu/data-and-maps/data>
10. Веб-система "Моніторинг та екологічна оцінка водних ресурсів України". URL: <http://ecology.gov.ua>
11. Звіт про стратегічну екологічну оцінку проєкту Плану управління річковим басейном Південного Бугу (2025-2030) / *Міністерство захисту довкілля та природних ресурсів України, Державне агентство водних ресурсів України*. Київ, 2024. 185 с. URL: https://davr.gov.ua/fls18/pl24/Pivdennyu_Byg_SEO.pdf.
12. План управління річковим басейном Південного Бугу на 2025–2030 роки. Державне агентство водних ресурсів України. Опубліковано 19.02.2025. URL: <https://www.davr.gov.ua/plan-upravlinnya-richkovim-basejnom-pivdenного-bugu>
13. Хільчевський В. К. Агрогідрохімія : підручник. Київ : ДІА, 2021. 176 с.
14. Про затвердження Плану управління річковим басейном Південного Бугу (2025–2030 роки) : розпорядження Кабінету Міністрів України від 29.11.2024 № 1210-р. *Урядовий кур'єр*. 2024. URL: <https://zakon.rada.gov.ua/laws/show/1210-2024-%D1%80#Text>.
15. Лебідь О., Охарєв В., Федосєєнков С., Шундель О., Теличко Р., Клименков О. Геоінформаційні технології екологічного моніторингу акваторії Чорного моря після руйнування Каховської ГЕС. Екологічна безпека та природокористування. 2023. Т. 48, № 4, с. 130–144. DOI: 10.32347/2411-4049.2023.4.130-144.

16. Хільчевський В., Забокрицька М. Особливості нормативного оцінювання якості води водних об'єктів для рекреаційних цілей в Україні. Гідрологія, гідрохімія і гідроекологія. 2022. № 1 (63). С. 53–63. DOI: 10.17721/2306-5680.2022.1.4
17. Mokin V.B. Kaggle Dataset «AI-ML-DS Training. L3AT: NH4 - NN models» URL: <https://www.kaggle.com/code/vbmokin/ai-ml-ds-training-l3at-nh4-nn-models/notebook#5.-Test-prediction>
18. Wu J., Zhang H., Lee S. Enhancing surface water quality prediction in data-scarce sites based on neural network framework. Journal of Hydrology. 2025. Vol. 617. P. 123456. DOI: <https://doi.org/10.1016/j.jhydrol.2025.123456>
19. Мокін В. Б. Машинне навчання та інтелектуальний аналіз даних : навч. посіб. Київ : ВНТУ, 2024. 263 с.
20. Мокін В. Б., Дратований М. В. / *Наука про дані: машинне навчання та інтелектуальний аналіз даних : електронний навчальний посібник комбінованого (локального та мережевого) використання* [Електронний ресурс] – Вінниця : ВНТУ, 2024. – 258 с. – Режим доступу: <https://docs.vntu.edu.ua/card.php?id=8163>
21. A state-of-the-art review: *Applications of deep learning in water quality management // Journal of Environmental Management*. – 2022. – Vol. 321. – P. 115874.
22. Козловський В. О., Лесько О. Й., Кавецький В. В. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт : уклад. Вінниця : ВНТУ, 2021. 42 с.

Додаток А

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

_____ д.т.н., проф. Віталій МОКІН

«__» _____ 2025 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на магістерську кваліфікаційну роботу

«ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КОМПЛЕКСНОГО АНАЛІЗУ
ГЕОПРОСТОРОВИХ ЗВ'ЯЗКІВ ТА ПРОГНОЗУВАННЯ СТАНУ МАСИВІВ
ВОД БАСЕЙНУ РІЧКИ ПІВДЕННИЙ БУГ»

08-34.МКР.002.02.000.ТЗ

Керівник: к.т.н., ас. каф. САІТ

_____ Ігор ШТЕЛЬМАХ

«__» _____ 2025 р.

Розробив: студент гр. 2ІСТ-24м

_____ Вадим КАРАВАЄВ

«__» _____ 2025 р.

Вінниця 2025

1. Підстава для проведення робіт

Підставою для виконання роботи є наказ № __ по ВНТУ від «__» _____ 2025 р., та індивідуальне завдання на МКР, затверджене протоколом № __ засідання кафедри САІТ від «__» _____ 2025 р.

2. Джерела розробки:

- Караваєв В. О., Крижановський Є. М., Штельмах І. М. Інформаційна технологія аналізу даних моніторингу стану поверхневих вод басейну р. Південний Буг // Матеріали LIV науково-технічної конференції підрозділів ВНТУ, Вінниця, 24-27 березня 2025 р. Електрон. текст. дані. 2025. URI: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2025/paper/view/23865>.
- В. Б. Мокін, К. О. Бондалетов, Є. М. Крижановський, і В. О. Караваєв, «МЕТОД АУГМЕНТАЦІЇ ТЕКСТІВ ПРО СТАН МАСИВІВ ВОД НА ОСНОВІ ІНТЕЛЕКТУАЛЬНОЇ ПРИВ'ЯЗКИ ДО БАГАТОЗВ'ЯЗНИХ ГЕОІНФОРМАЦІЙНИХ СИСТЕМ ІМЕНОВАНИХ СУТНОСТЕЙ», Вісник ВПІ, вип. 3, с. 55–65, Черв. 2023.

3. Мета і призначення роботи:

підвищення швидкості проведення комплексного аналізу геопросторових зв'язків та збільшення точності прогнозування стану масивів басейну річки Південний Буг шляхом використання методів штучного інтелекту та геоінформаційних технологій.

4. Вихідні дані для проведення робіт:

Результати моніторингу якості поверхневих вод басейну річки Південний Буг, дані з відкритих екологічних реєстрів і офіційних джерел, геопросторові матеріали ГІС, показники гідрохімічного стану водних об'єктів, а також статистична інформація для побудови моделей аналізу та прогнозування.

5. Методи дослідження:

Дослідження існуючих інформаційних технологій, розвідувальний аналіз, розробка UML-діаграм, використання середовища Kaggle, мови програмування HTML/CSS/JS та Python.

6. Етапи роботи і терміни їх виконання:

1. Загальна характеристика поставленої задачі ____ – ____
2. Розроблення та впровадження інформаційної технології аналізу даних моніторингу якості поверхневих вод ____ – ____
3. Обробка результатів спостережень та візуалізація даних у середовищі Kaggle ____ – ____
4. Побудова та оцінювання моделей нейронних мереж для прогнозування якості поверхневих вод ____ – ____
5. Економічна частина ____ – ____
6. Оформлення матеріалів до захисту МКР ____ – ____

7. Очікувані результати та порядок реалізації:

Очікується створення та впровадження інформаційної технології, що забезпечить аналіз і прогнозування стану водних масивів басейну Південного Бугу на основі обробки моніторингових даних, моделювання та формування практичних рекомендацій для підвищення ефективності екологічного контролю.

8. Вимоги до розробленої документації

Пояснювальна записка оформлена у відповідності до вимог «Методичних вказівок до виконання магістерських кваліфікаційних робіт для студентів спеціальності 126 «Інформаційні системи та технології» (освітня програма «Інформаційні технології аналізу даних та зображень»)

9. Порядок приймання роботи

Публічний захист..... «__» _____ 2025 р.
 Початок розробки «__» _____ 2025 р.
 Граничні терміни виконання МКР «__» _____ 2025 р.

Розробив студент групи 2ІСТ-24м _____ Вадим КАРАВАЄВ

Додаток Б

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: «Інформаційна технологія комплексного аналізу геопросторових зв'язків та прогнозування стану масивів вод басейну річки Південний Буг»

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра САІТ, ФІТА, гр. 2ІСТ-24м

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 12,18 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне):

- Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту
- У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

Віталій МОКІН, зав. каф. САІТ

_____ (підпис)

Сергій ЖУКОВ, доц. каф. САІТ

_____ (підпис)

Особа, відповідальна за перевірку _____ (підпис)

Сергій ЖУКОВ

З висновком експертної комісії ознайомлений(-на)

Керівник _____ (підпис)

Ігор ШТЕЛЬМАХ, к.т.н., ас. каф. САІТ

Здобувач _____ (підпис)

Вадим КАРАВАЄВ

Додаток В
Лістинг програми

```
i # Load environment variables
load_dotenv()

# Global variables to store dataframes
posts_df = None
measurements_df = None
merged_df = None
pollution_df = None

@asynccontextmanager
async def lifespan(app: FastAPI):
    # Startup
    global posts_df, measurements_df, merged_df, pollution_df
    try:
        # Get the base directory
        base_dir = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

        # Load monitoring posts data
        posts_path = os.path.join(base_dir, "data", "name_coord_posts.csv")
        measurements_path = os.path.join(base_dir, "data", "measurement.csv")
        pollution_path = os.path.join(base_dir, "data", "pollution.csv")

        posts_df = pd.read_csv(posts_path)
        measurements_df = pd.read_csv(measurements_path)
        measurements_df['Date'] = pd.to_datetime(measurements_df['Date'],
format='%m/%d/%y')
```

```
# Load pollution data
pollution_df = pd.read_csv(pollution_path)
print(f"Loaded pollution data: {len(pollution_df)} companies")
print("Columns in pollution data:", pollution_df.columns.tolist())
print("First few rows of pollution data:")
print(pollution_df.head())

# Create a merged dataframe for easier querying
merged_df = pd.merge(
    measurements_df,
    posts_df,
    on="ID_Station",
    how="left"
)

except Exception as e:
    print(f"Error loading data files: {e}")
    raise RuntimeError(f"Failed to load data files: {e}")

yield

# Shutdown
# Clean up resources if needed

app = FastAPI(title="Water Quality Monitoring API", lifespan=lifespan)

# Add CORS middleware to allow frontend to access API
app.add_middleware(
    CORSMiddleware,
    allow_origins=["*"],
    allow_credentials=True,
    allow_methods=["*"],
```

```

@app.get("/api/measurements/{date}", response_model=List[Dict[str,
Any]])
async def get_measurements_by_date(date: str):
    """Get measurements for a specific date"""
    # If date is 'all', return all measurements
    if date == 'all':
        result = []
        # Group by station and get the latest measurement for each
        latest_measurements =
measurements_df.sort_values('Date').groupby('ID_Station').last()
        for _, row in latest_measurements.iterrows():
            post_id = row.name # ID_Station is the index
            post_info = posts_df[posts_df['ID_Station'] == post_id]

            if not post_info.empty:
                post_name = post_info.iloc[0]['Name_post']
                measurement = {
                    'id': post_id,
                    'name': post_name,
                    'date': row['Date'],
                    'NH4': float(row['NH4']) if pd.notna(row['NH4']) else None,
                    'BSK5': float(row['BSK5']) if pd.notna(row['BSK5']) else None,
                    'Suspended': float(row['Suspended']) if
pd.notna(row['Suspended']) else None,
                    'O2': float(row['O2']) if pd.notna(row['O2']) else None,
                    'NO3': float(row['NO3']) if pd.notna(row['NO3']) else None,
                    'NO2': float(row['NO2']) if pd.notna(row['NO2']) else None,
                    'SO4': float(row['SO4']) if pd.notna(row['SO4']) else None,
                    'PO4': float(row['PO4']) if pd.notna(row['PO4']) else None,
                    'Cl': float(row['Cl']) if pd.notna(row['Cl']) else None

```

```

    }
    result.append(measurement)
return result

# Convert date string to datetime object for comparison
try:
    query_date = datetime.strptime(date, '%Y-%m-%d')
except ValueError:
    raise HTTPException(status_code=400, detail="Invalid date format.
Use YYYY-MM-DD.")

# Check if date exists in the dataset
if date not in measurements_df['Date'].dt.strftime('%Y-%m-%d').values:
    raise HTTPException(status_code=404, detail=f"No measurements
found for date {date}")

# Filter measurements by date
date_measurements = measurements_df[measurements_df['Date'] ==
query_date]

# Join with posts data to get post names
result = []
for _, row in date_measurements.iterrows():
    post_id = row['ID_Station']
    post_info = posts_df[posts_df['ID_Station'] == post_id]

    if not post_info.empty:
        post_name = post_info.iloc[0]['Name_post']
        measurement = {
            'id': post_id,

```

```

        'name': post_name,
        'NH4': float(row['NH4']) if pd.notna(row['NH4']) else None,
        'BSK5': float(row['BSK5']) if pd.notna(row['BSK5']) else None,
        'Suspended': float(row['Suspended']) if pd.notna(row['Suspended'])
else None,
        'O2': float(row['O2']) if pd.notna(row['O2']) else None,
        'NO3': float(row['NO3']) if pd.notna(row['NO3']) else None,
        'NO2': float(row['NO2']) if pd.notna(row['NO2']) else None,
        'SO4': float(row['SO4']) if pd.notna(row['SO4']) else None,
        'PO4': float(row['PO4']) if pd.notna(row['PO4']) else None,
        'Cl': float(row['Cl']) if pd.notna(row['Cl']) else None
    }
    result.append(measurement)

"""Find measurements where parameter is below the specified threshold"""
if param_name not in merged_df.columns:
    return []

# Filter out NaN values and apply threshold
filtered_df = merged_df[(merged_df[param_name].notna()) &
(merged_df[param_name] < threshold)]
if filtered_df.empty:
    return []

# Prepare results
results = []
for _, row in filtered_df.iterrows():
    results.append(prepare_measurement_result(row))

return results

```

```
raise HTTPException(status_code=400, detail="Query cannot be
empty")
# Detect parameter in query
detected_param = None
for key, value in param_mapping.items():
    if key in user_query:
        detected_param = value
        print(f"Found parameter match: {key} -> {value}")
        break

print(f"Detected parameter: {detected_param}")

# Detect season in query
season = None
if 'літо' in user_query or 'влітку' in user_query:
    season = 'summer'
elif 'зима' in user_query or 'взимку' in user_query:
    season = 'winter'
elif 'весна' in user_query or 'навесні' in user_query:
    season = 'spring'
elif 'осінь' in user_query or 'восени' in user_query:
    season = 'autumn'
elif 'травень' in user_query or 'травні' in user_query:
    season = 'may'

print(f"Detected season: {season}")

# Define query patterns and corresponding actions
# Check for maximum/highest/largest pattern
```

```

if any(term in user_query for term in ['максимум', 'найбільш', 'найвищ',
'найгірш', 'найвищий', 'найбільший']):
    print("Found maximum/highest pattern")
    if detected_param == 'all':
        # For all parameters, find worst quality
        results = find_worst_quality()
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найвищими показниками")
        return results
    elif detected_param == 'O2':
        # For oxygen, highest is best
        results = find_maximum_parameter('O2', season)
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найвищим вмістом кисню")
        return results
    elif detected_param:
        # For other parameters, highest is worst
        results = find_maximum_parameter(detected_param, season)
        if not results:
            raise HTTPException(status_code=404, detail=f"Не знайдено
постів з найвищим вмістом {detected_param}")
        return results
    else:
        # General worst quality
        results = find_worst_quality()
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найвищими показниками")

```

```
return results

# Check for minimum/lowest pattern
elif any(term in user_query for term in ['мінімум', 'найменш',
'найнижч', 'найнижчий', 'найменший']):
    print("Found minimum/lowest pattern")
    if detected_param == 'all':
        # For all parameters, find best quality
        results = find_best_quality()
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найнижчими показниками")
        return results
    elif detected_param == 'O2':
        # For oxygen, lowest is worst
        results = find_minimum_parameter('O2', season)
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найнижчим вмістом кисню")
        return results
    elif detected_param:
        # For other parameters, lowest is best
        results = find_minimum_parameter(detected_param, season)
        if not results:
            raise HTTPException(status_code=404, detail=f"Не знайдено
постів з найнижчим вмістом {detected_param}")
        return results
    else:
        # Default to general best quality
        results = find_best_quality()
```

```

    if not results:
        raise HTTPException(status_code=404, detail="Не знайдено
постів з найнижчими показниками")
    return results

# If no specific pattern matched but we detected a parameter,
# return stations with highest values for that parameter
elif detected_param:
    print(f"No specific pattern matched, using detected parameter:
{detected_param}")
    if detected_param == 'all':
        results = find_worst_quality()
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найвищими показниками")
        return results
    elif detected_param == 'O2':
        # For oxygen, higher is better
        results = find_maximum_parameter(detected_param, season)
        if not results:
            raise HTTPException(status_code=404, detail="Не знайдено
постів з найвищим вмістом кисню")
        return results
    else:
        # For other contaminants, higher is worse
        results = find_maximum_parameter(detected_param, season)
        if not results:
            raise HTTPException(status_code=404, detail=f"Не знайдено
постів з найвищим вмістом {detected_param}")
        return results

```

```

# Default response - return best and worst quality stations
else:
    print("No parameter detected, returning best and worst quality
stations")
    # Combine best and worst quality results
    best_results = find_best_quality()[:3] # Top 3 best
    worst_results = find_worst_quality()[:3] # Top 3 worst
    if not best_results and not worst_results:
        raise HTTPException(status_code=404, detail="Не знайдено
постів з найкращими та найгіршими показниками")
    return best_results + worst_results

except Exception as e:
    print(f"Error in ask endpoint: {e}")
    raise HTTPException(status_code=500, detail=f"Error processing
query: {str(e)}")

@app.get("/api/companies", response_model=List[Dict[str, Any]])
async def get_companies():
    """Get list of all companies with their coordinates"""
    if pollution_df is None:
        raise HTTPException(status_code=404, detail="Pollution data not
loaded")

    result = pollution_df[['company_name', 'location', 'lat',
'lon']].to_dict('records')
    print(f"Returning {len(result)} companies")
    return result

```

ІЛЮСТРАТИВНА ЧАСТИНА

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КОМПЛЕКСНОГО АНАЛІЗУ
ГЕОПРОСТОРОВИХ ЗВ'ЯЗКІВ ТА ПРОГНОЗУВАННЯ СТАНУ МАСИВІВ
ВОД БАСЕЙНУ РІЧКИ ПІВДЕННИЙ БУГ

Нормоконтроль: к.т.н., доцент

_____ Сергій ЖУКОВ

«__» _____ 2025 р.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Дані моніторингу (за районом річкового басейну або суббасейну)											
2	з 01.03.2000 до 11.03.2025											
3	Район											
4	1. Пост: Балка Ташлик, 6 км, Ташлицьке вдсх., став-охолоджувач ПУ АЕС (міст через Ташлицьке вдсх., південь м. Південно-Українськ)											
5	Значення	Показник										
6		Амоній-іони, мг										
7	Дата	Біохімічне споживання Завислі (с Кисень рс Нітрат-іон Нітрит-іон Сульфат-іон Фосфат-іон Хлорид-іони, мг/дм ³)										
8	17.04.2000	0,035	1,3	0	12,98	4,7	0,038	403,2	0,143	146,1		
9	04.07.2000	0,031	1,13	10,4	9,89	0,24	0,022	416	0,06	149		
10	04.12.2000	0,042	1,2	0	9,63	2,02	0,006	384	0,275	154,5		
11	06.02.2001	0,062	2,88	1,6	11,77	4,3	0,004	364,8	0,38	160,1		
12	03.04.2001	0	2,5	15,6	11,4	11,4	0,023	397	0,197	146		
13	09.07.2001	0,05	1,98	21	12,97	0,51	0,022	384	0,002	154,5		
14	08.10.2001	0,106	1,04	0	9,63	0,86	0,007	364,8	0,061	148,9		
15	09.01.2002	0	1,77	12	12,04	3,78	0,032	317,2	0,315	168,4		

Рисунок Г.1 – Початкові дані

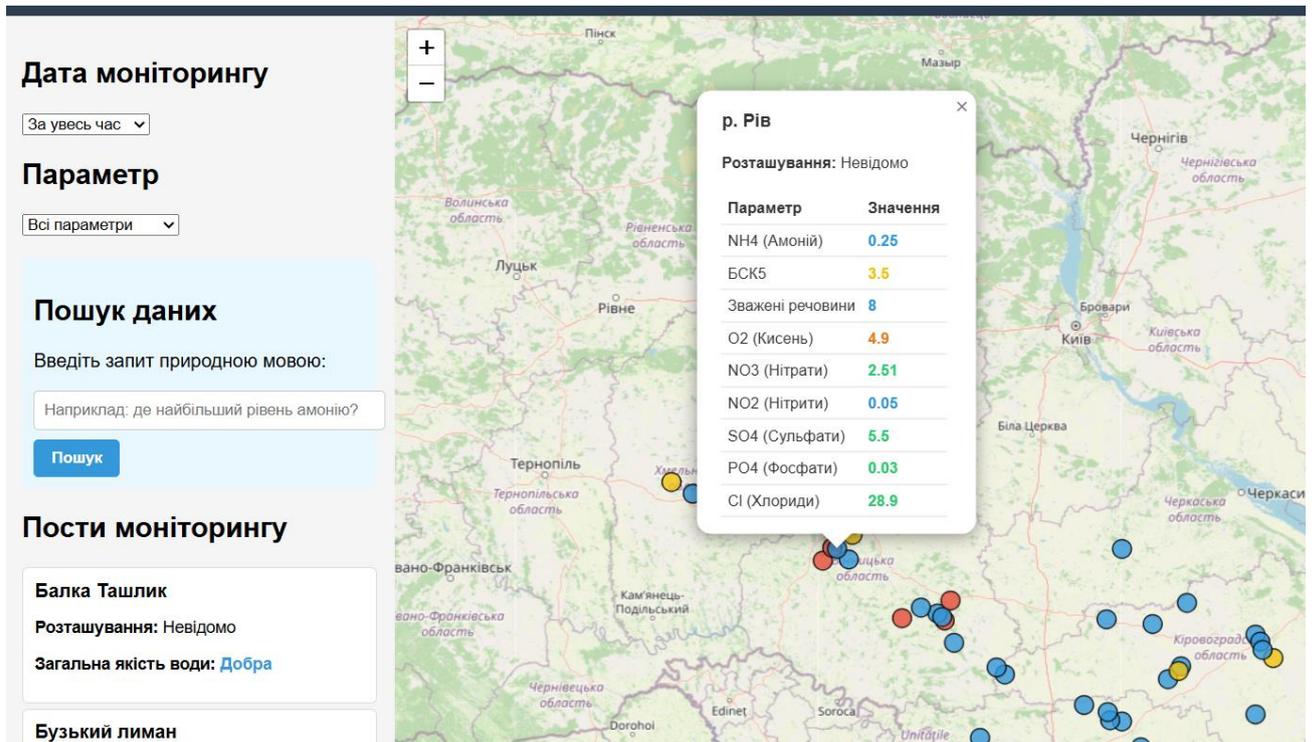


Рисунок Г.2 – Фрагмент скріншоту сайту системи моніторингу якості води

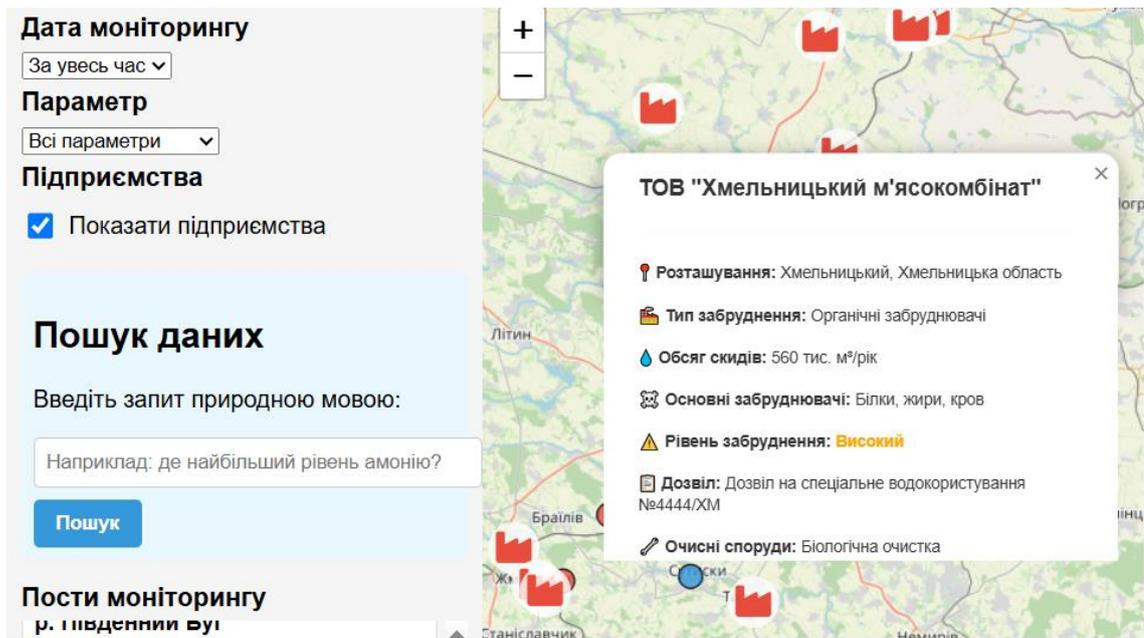


Рисунок Г.3 – Відображення підприємств забруднювачів

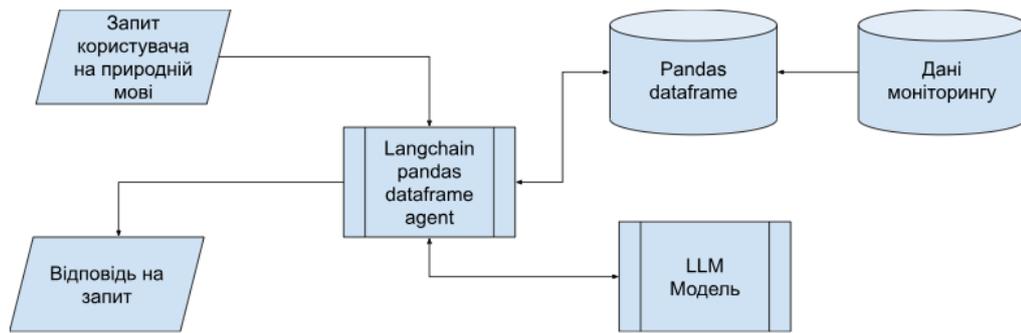


Рисунок Г.4 – Функціональна схема роботи ШІ-агента

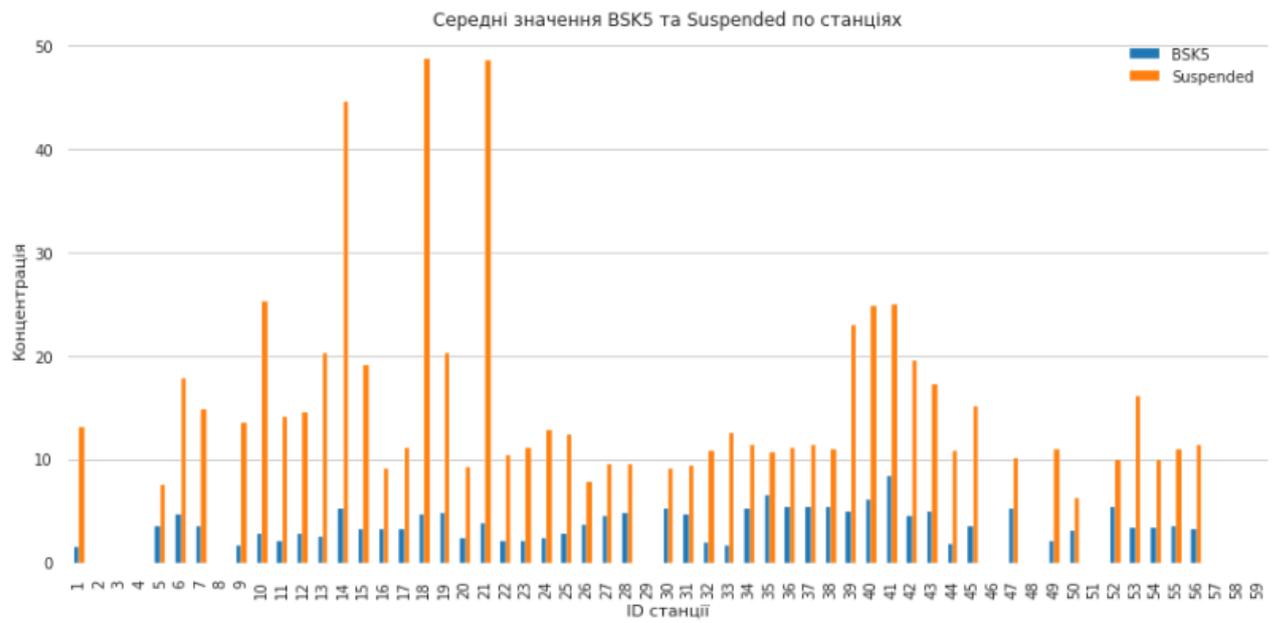


Рисунок Г.5 – Середні значення параметрів BSK5 та Suspended по станціях

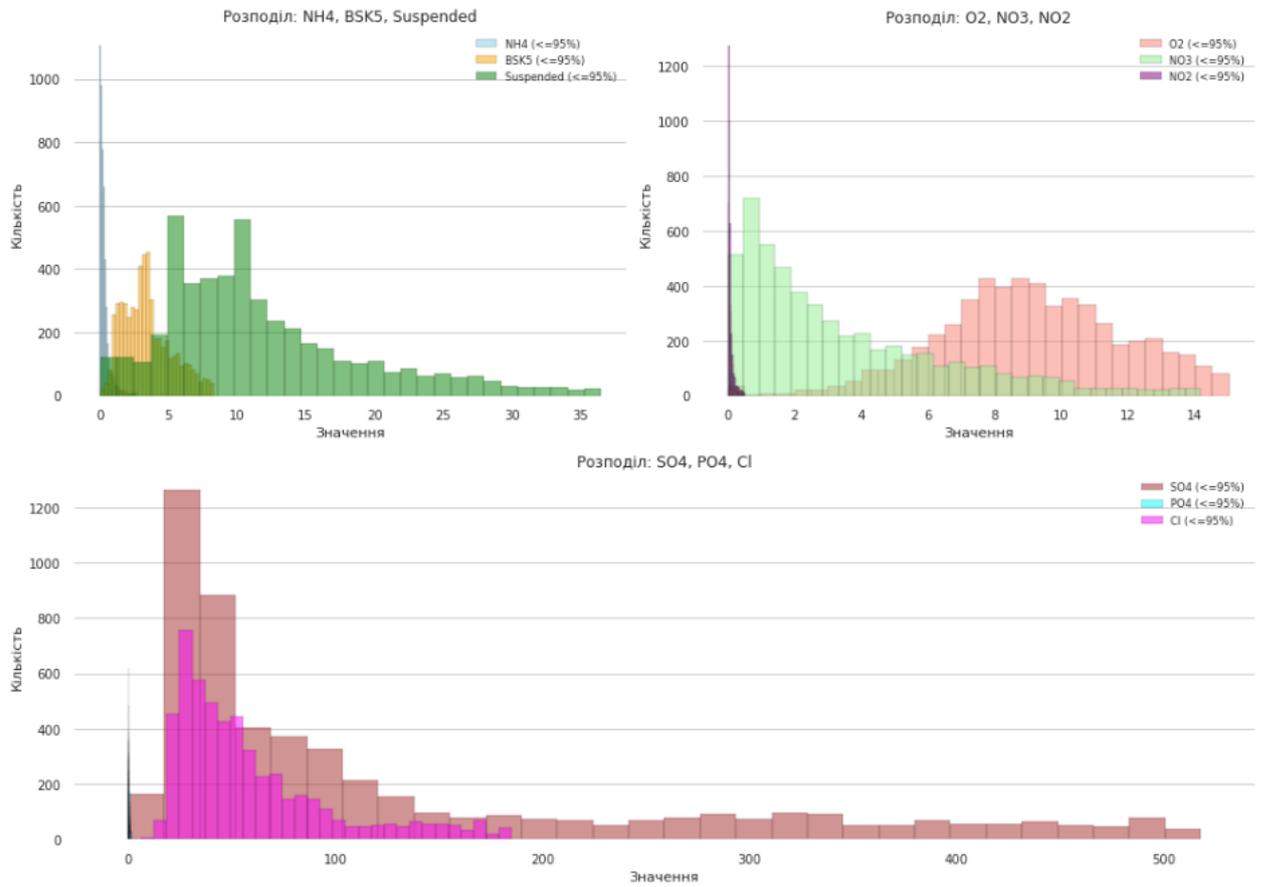


Рисунок Г.6 – Розподіл концентрацій ключових хімічних показників

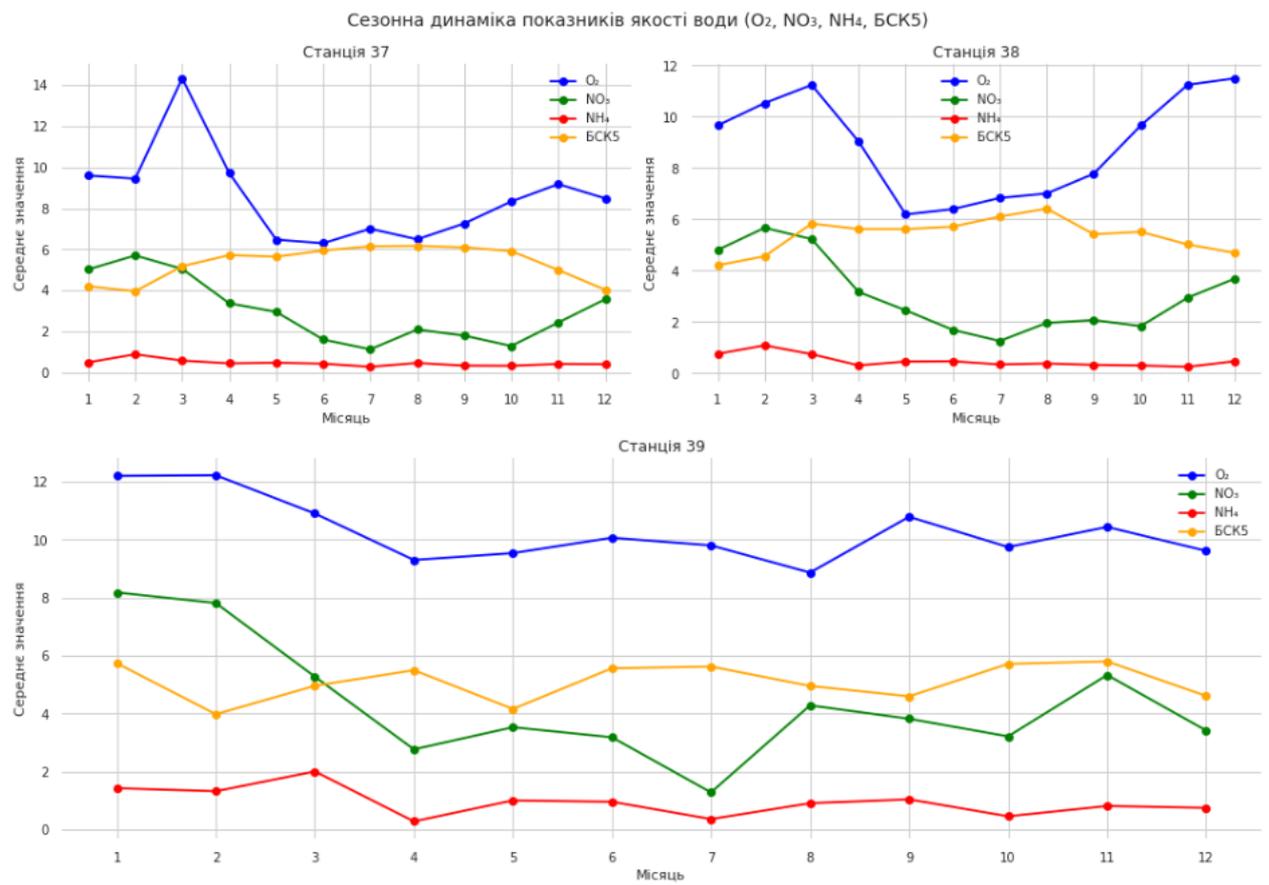


Рисунок Г.7 – Сезона динаміка показників якості води

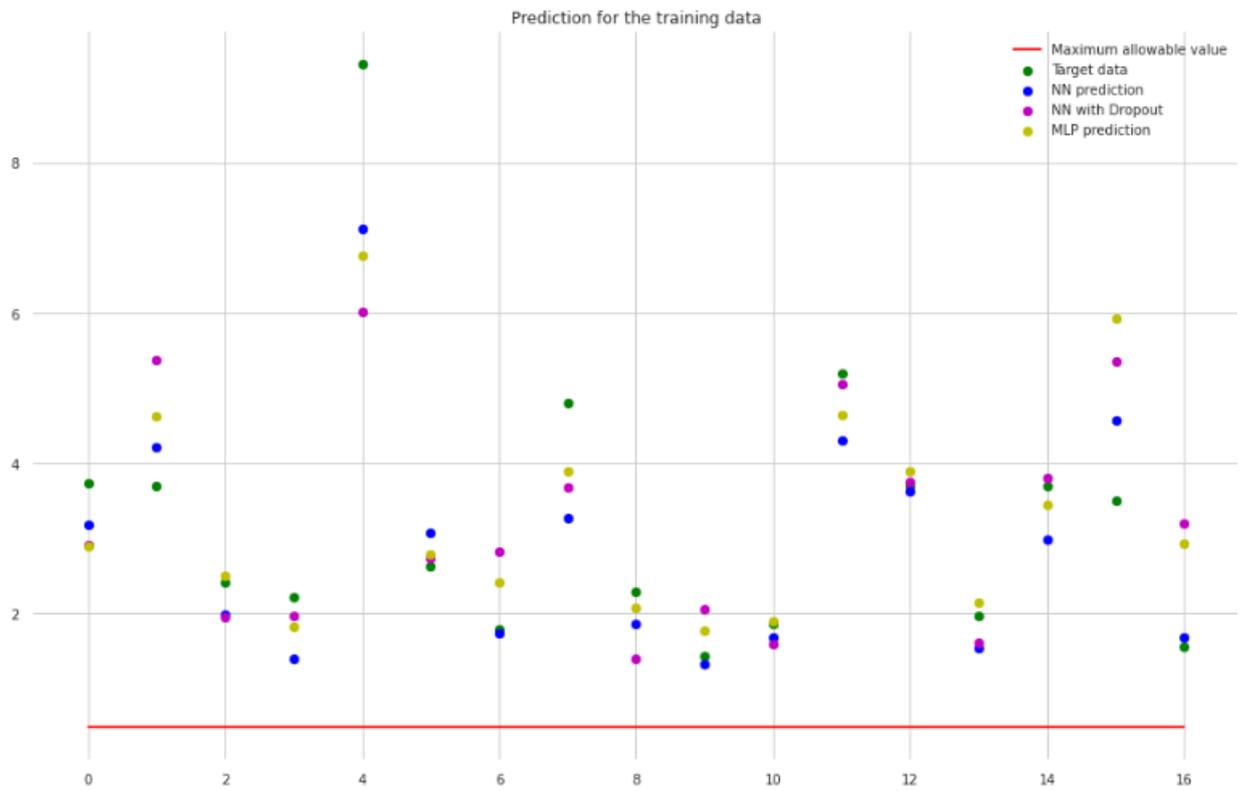


Рисунок Г.8 – Графік прогнозування тренувальних даних

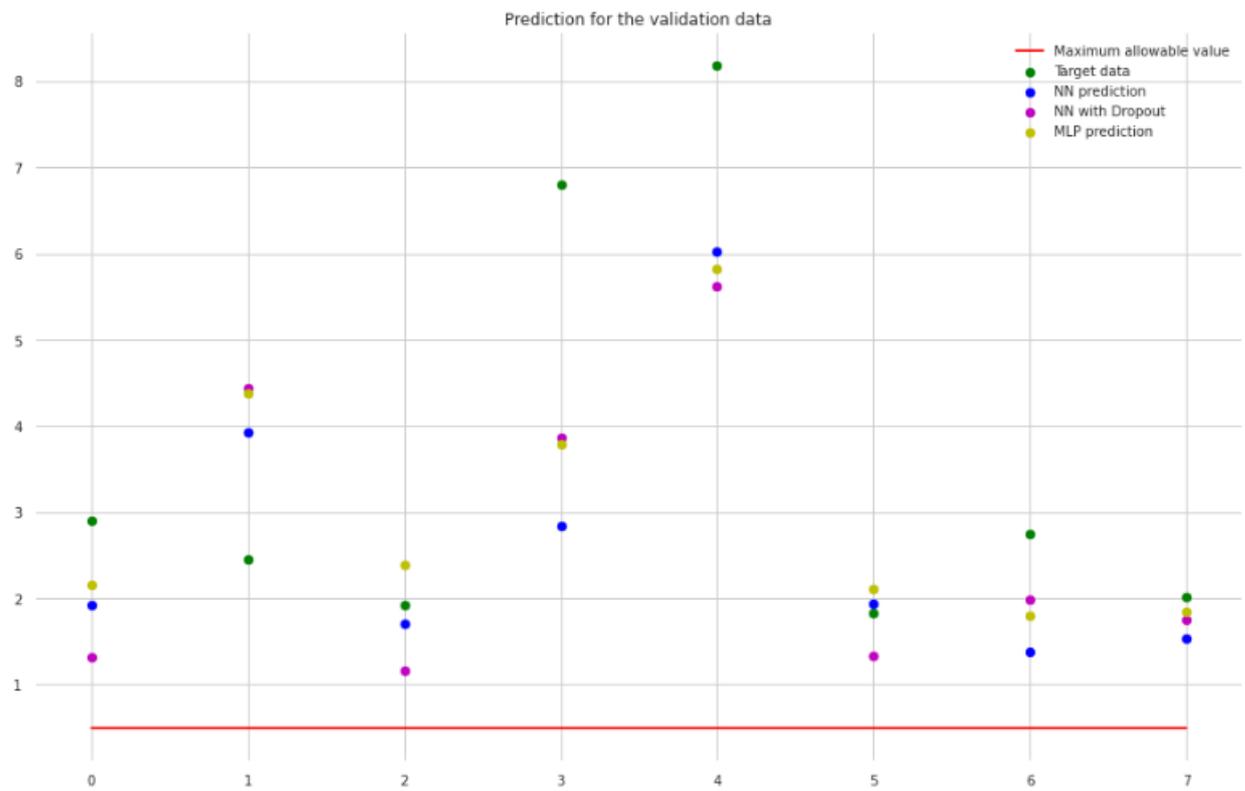


Рисунок Г.9 – Графік прогнозування валідаційних даних

```
plot_prediction(None, y_test_nn, y_test_nn2, y_test_mlp, 'test')
```

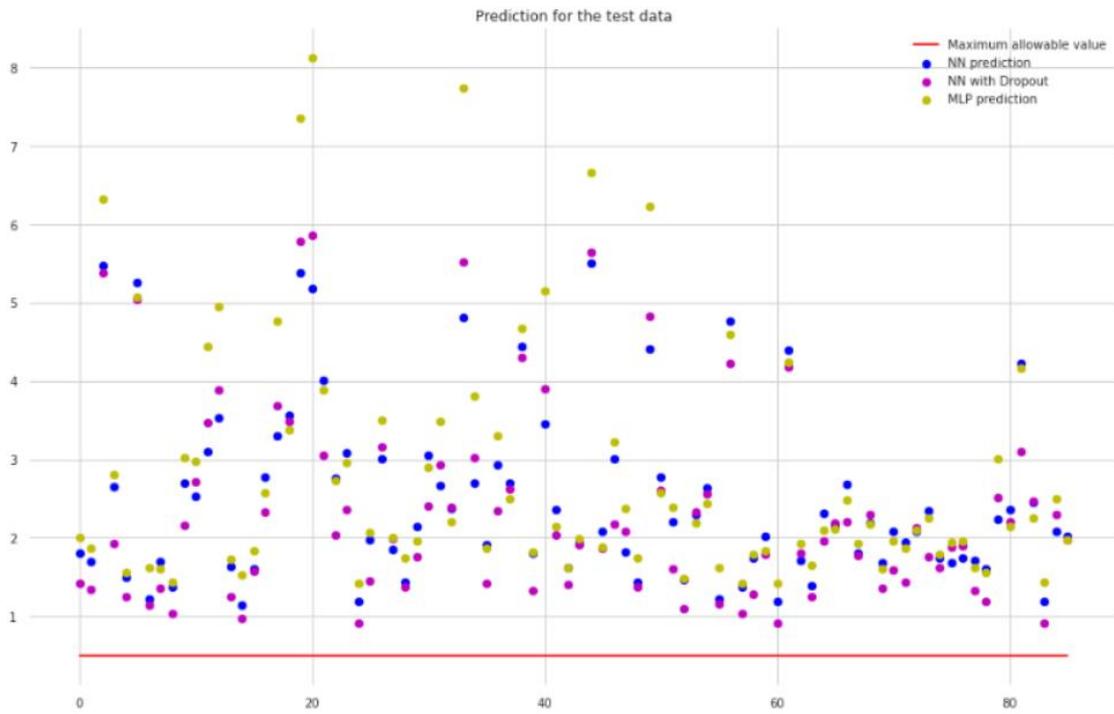


Рисунок Г.10 – Графік прогнозування тестових даних

	model	train_mse	valid_mse	test_mse
0	NN Regressor	0.607303	2.71000	2.76
2	MLP Regressor	0.607303	2.67000	1.95
1	NN Regressor with Dropout	0.610000	2.94673	2.08

Рисунок Г.11 – Таблица навчання моделей