

Т. ПЕТРЕНКО, А. ЗАДОРЖНИЙ

ПЛАТФОРМА ДЛЯ ІНТЕГРАЦІЇ ІНСТРУМЕНТІВ І СЕРВІСІВ ОБРОБЛЕННЯ МЕТЕОДАНИХ ЗАСОБАМИ ШТУЧНОГО ІНТЕЛЕКТУ

Предметом дослідження є інструменти, сервіси та платформи забезпечення прогнозування локальних метеоумов. Процес прогнозування метеоумов за певною геолокацією доволі складний. Джерелами помилок прогнозування є об'єктивні причини, які є наслідками складності метеопроцесів, що взагалі існували завжди, а також суттєвих кліматичних змін через глобальне потепління. Використання моделей машинного та глибокого навчання (*Machine Learning and Deep Learning*, ML&DL) разом з уточненням результатів класичних фізичних моделей атмосфери – важливий крок підвищення точності моделей прогнозування. Моделі для прогнозування метеоумов стають усе більше гібридними, а інформація, що застосовується для навчання ML&DL-моделей, – усе більш різноманітною та має різні джерела походження. Для трансформації структурованих, неструктурованих та напівструктурованих метеоданих і прогнозування метеоумов використовуються потужні й не завжди безкоштовні середовища провідних розробників. **Мета роботи** – аналіз можливостей наявних платформ використання ML&DL-моделей для прогнозування метеоумов і створення платформи для прогнозування метеоумов, яка має гібридну полегшену архітектуру (*Hybrid LightWeight Architecture*, HLWA). Платформа на основі HLWA виконує такі **завдання**: розподілення етапів оброблення метеоданих між різними постачальниками інструментів і сервісів із хмарних середовищ, але водночас дає змогу інтегрувати ресурси та інструменти оброблення на одній платформі. Розгортання інструментів і сервісів підготовки метеоданих і прогнозування метеоумов у роботі пропонується на сервері *AWS Lightsail* з використанням *Node-RED*, *MongoDB* та *AWS SageMaker AI*. У статті впроваджено **методи** декомпозиції процесів прогнозування метеоумов. **Результатом роботи** є створення моделі платформи у вигляді UML-діаграми компонентів з уточненням властивостей кожного компонента платформи та інтерфейсів. **Висновком статті** є твердження, що застосування запропонованої платформи для дослідження гібридних моделей прогнозування метеоумов на основі ML&DL-моделей є зручним, економічним і перспективним рішенням.

Ключові слова: модель платформи прогнозування метеоумов; штучний інтелект; сервіси хмарних середовищ.

Вступ

Створення сучасних моделей, підходів і технологій прогнозування метеоумов на основі штучного інтелекту речей (*Artificial Internet of Things*, AIoT) посилило розвиток гібридних моделей прогнозування метеоумов і платформ щодо науки про дані та машинного навчання для метеопрогнозу (*Data science and machine learning platforms*, DSML platforms). Нові підходи до прогнозування метеоумов, побудовані на ML&DL-моделях, довели спроможність покращити процес і результати прогнозування. Поєднання моделей відбувається на різних рівнях і додає різні властивості гібридним моделям.

Використання гібридних моделей стимулює створення платформ прогнозування, що дають змогу:

1) інтегрувати процеси отримання інформації для прогнозування метеоумов з різних глобальних і локальних джерел, зокрема з процесами отримання даних з інших платформ, які також підтримують генерацію синтетичних даних;

2) забезпечувати підготовку до застосування метеоданих відповідно до вимог ML&DL-моделей;

3) зберігати інформацію для прогнозування метеоумов і його результати в сучасних базах даних;

4) навчати й тестувати ML&DL-моделі із використанням метрик оцінювання результативності моделей;

5) забезпечувати безпосередньо процес прогнозування метеоумов;

6) створювати та виконувати сценарії, що дають змогу зменшувати розмірність ML&DL-моделей з метою створення малих лінгвістичних моделей (*Small Language Models*, SLMs), що підтримують оброблення гетерогенних метеоданих на рівні граничних обчислень;

7) забезпечувати відповідні внутрішні інтерфейси між компонентами системи й зовнішні інтерфейси між платформою прогнозування та джерелами інформації, а також з користувачами системи;

8) зменшити накладні витрати для прогнозування метеоумов за певною геолокацією.

Визначення архітектури платформи прогнозування метеоумов потребує комплексного підходу та є актуальним. У цій роботі запропоновано модель створення платформи підготовки інформації та прогнозування на основі гібридної полегшеної архітектури HLWA, яка забезпечує розподілення етапів оброблення метеоданих між різними постачальниками інструментів і сервісів, але водночас дає змогу інтегрувати ресурси та інструменти оброблення на одній платформі. Розгортання інструментів і сервісів підготовки метеоданих і прогнозування метеоумов запропоновано на сервері *AWS Lightsail* з використанням *Node-RED*, *MongoDB Atlas* та *AWS SageMaker AI*. У статті обґрунтовано нову модель інтеграції ресурсів та інструментів прогнозування метеоумов, що забезпечує компактність і економічність створеної платформи прогнозування.

Аналіз останніх досліджень і публікацій

Прогнозування метеоумов є нелегким завданням [1–4] через труднощі природних процесів формування метеоумов. З метою зменшення складності модельного уявлення використовується декомпозиція процесів формування метеоумов, яка розподіляє процеси на різні рівні їх просторового уявлення, іншими словами, горизонтального та вертикального розподілення. Для просторового уявлення процесів упроваджено термін "просторова роздільна здатність", що задає параметри куба атмосфери (км^3), для якого значення властивостей метеоумов є однаковими в межах цього куба загальної тривимірної сітки охоплення прогнозу.

Тривимірний куб охоплення прогнозу є фактично нерівномірною для атмосфери над різними територіями планети Земля. Крім цього, просторова роздільна здатність для різних моделей прогнозування метеоумов є неоднаковою. Для вертикального уявлення зазвичай використовують до 64 рівнів вертикального розподілення, починаючи з поверхні землі. Але, наприклад, для моделі ERA5 значення вертикального розподілення дорівнює до 137 рівнів щодо поверхні землі до 0.01 hPa (приблизно 80 км), а площа горизонтального розподілення дорівнює 31 км^2 [1].

Моделі чисельного прогнозування погоди (*Numerical Weather Prediction*, NWP), до яких належить ERA5, побудовані на фізичних рівняннях динаміки шарів атмосфери та гідросфери [1, 3, 4]. NWP використовує складні математичні моделі, що описують динаміку процесів атмосфери, океанів і

поверхневих процесів, але розрахунки за NWP є надзвичайно ресурсомісткими [5].

Для декомпозиції систем прогнозування метеоумов розглядається також часова розподіленість метеоумов за допомогою застосування різних термінів прогнозування. Виокремлюють постійне прогнозування метеоумов (тобто на наступні години на сьогодні й на завтра на основі погодинних метеоданих за сьогодні), короткочасне прогнозування (на 3–5 діб), середньострокове прогнозування (на тиждень або 10–14 діб) та довгострокове прогнозування метеоумов (на місяць і пізніше) [6]. Системи прогнозування метеоумов надають доволі точні постійні та короткочасні прогнози (до 90%), але для середньострокового та довгострокового прогнозування якість результату знижується (до 80 та 50%) [6].

Розвиток AIoT й супутникового зв'язку забезпечив наявність значних обсягів метеоданих. Приблизно 85% інформації для глобальних моделей клімату отримують зараз саме із систем супутникового зв'язку [7]. Також збільшилась кількість та якість отримання метеоданих із локальних метеостанцій. Розвиток апаратного забезпечення хмарних середовищ дає змогу інтегрувати не тільки спостереження, а й результати метеопрогнозів, добутих за різними сучасними моделями прогнозування. Інтеграція джерел метеоданих і моделей прогнозування метеоумов стає найбільш актуальною проблемою, розв'язання якої дослідники вбачають в об'єднанні підходів NWP та ML&DL-моделей [8, 9].

Застосування ML&DL покращує якість прогнозу [10–15], але успішність ML&DL у прогнозуванні потребує наявності значної кількості різних так званих історичних показників про минулі погодні умови [1, 3, 4].

Історичні метеодані відіграють важливу роль у тренуванні та вдосконаленні ML&DL-моделей, оскільки вони дають змогу навчити систему розпізнавати певні тенденції (патерни) в метеоумовах, що можуть виникати за різних кліматичних сценаріїв. Для навчання ML&DL-моделей використовують різні за технологією формування дані. Застосування метеоданих повторного аналізу (*Reanalysis Weather Data*, RWD) допомагає акумулювати інформацію з різних джерел та аналізувати спостереження, які були недоступні в режимі реального часу [1, 3, 4].

Отримання історичних метеоданих з глобальних метеорологічних баз даних, зокрема Європейського центру середньострокових прогнозів погоди (*European*

Centre for Medium-Range Weather Forecasts, ECMWF) [16], Національного управління океанічних і атмосферних досліджень США (*National Oceanic and Atmospheric Administration, NOAA*) [17] та з API сайтів метеоресурсів [18], дає змогу сформувати необхідну кількість інформації, що разом з локальними метеоданими відповідної геолокації спроможні забезпечити навчання й тестування ML&DL-моделей.

Метеодані мають бути зібрані, оброблені та структуровані до навчання ML&DL-моделей. Кліматичний стан планети динамічно змінюється та реагує на виклики, що спричиняють техногенні катастрофи. Війна в Україні не тільки призвела до загибелі значної кількості населення та суттєво вплинула на життя мільйонів людей, але й змінила метеоумови й умови отримання впевнених показників із локальних метеостанцій, а також на окремих територіях спричинила знищення метеостанцій [19]. Масштабність таких змін на фоні мінливості клімату всієї планети також змінює уяву про поняття аномалії в метеоданих, а тому до етапів їх оброблення, таких як очищення, перетворення й нормалізація, для дотримання сумісності інформації з ML&DL-моделями додаються також етапи аналізу додаткових показників. Крім того, інформація має зберігатися в спеціалізованих базах даних, що забезпечують швидкий доступ до інформації різного типу й уможливають постійне оновлення для покращення моделей.

Успішність використання ML&DL-моделей залежить також від обраної DSML-платформи [20]. Успішними є, наприклад, платформи таких провідних компаній, як *Microsoft, Google, Amazon* та *IBM* [21]. Набули поширення DSML-платформи *Databricks Unified Analytics Platform, KNIME Analytics platform, TIBCO Software, Alteryx Analytics, SAS, H2O.ai* та *DataRobot* [21]. Популярні інструменти, зокрема *Scikit-learn, PyTorch, TensorFlow, Weka, KNIME, Colab* тощо, також відіграють важливу роль у дослідженнях ML&DL-моделей [21].

Наявність у провідних компаній своїх моделей прогнозування метеоумов покращує інтеграцію ресурсів для прогнозування. Спеціалізовані платформи для прогнозування метеоумов створюють і використовують сотні різних постачальників метеопрогнозу для кінцевих користувачів мобільних застосунків [14]. Важливість прогнозування локальних метеоумов для підприємств різних галузей (агропромисловість, транспорт, охорона здоров'я тощо) стимулює створення інтегрованих платформ, що беруть до уваги властивості

певного напрямку застосування метеопрогнозів і використовують моделі AIoT для оброблення метеоданих та прогнозування метеоумов [14, 15].

Якість прогнозу метеоумов залежить від розуміння складності формування метеоумов, обрання або створення відповідних баз метеоданих, моделей, платформ та інструментів прогнозування метеоумов.

Створення платформи для прогнозування метеоумов потребує з'ясування місця розміщення платформи у відповідному хмарному середовищі, визначення компонентів платформи та ресурсів для використання компонентів, а також визначення інтерфейсів між компонентами для злагодженої роботи платформи.

У роботі запропоновано гібридну полегшену архітектуру платформи прогнозування метеоумов, що дає змогу проводити експерименти та інтегрує можливості AIoT, сучасних ML&DL-моделей та документо-орієнтованих баз даних. Для розміщення платформи обрано сервер *AWS Lightsail* [22]; *Node-RED* як інтеграційне середовище [23]; *MongoDB Atlas* для збереження історичних і прогнозованих метеоданих [24] та *AWS SageMaker AI* як сервіс для розроблення, навчання та розгортання ML&DL-моделей [25].

Мета роботи й завдання

Інтенсивні дослідження ML&DL-моделей, що дають змогу прогнозувати метеоумови [5, 7–15], а також створювати різноманітні бази метеоданих [16–18], підтверджує актуальність експериментів з метою покращення метеопрогнозів. Але потужні середовища прогнозування та великі бази метеоданих здебільшого налаштовані на попит з боку бізнес-користувачів, а безкоштовний режим використання сервісів та інформації є обмеженим.

Безоплатні бази метеоданих забезпечують зацікавленість до них з боку освітньо-наукової спільноти як до інформаційного ресурсу для навчання та дослідження. Якість і формати зберігання метеоданих у різних базах можуть бути неоднаковими. Тому одним із важливих питань прогнозування метеоумов є розуміння джерел і властивостей інформації, що використовується для метеопрогнозування. Первинні метеодані отримують унаслідок вимірювання показників метеоумов, таких як температура, вологість, тиск, швидкість вітру, ультрафіолет тощо. Метеоінформацію збирають за допомогою наземних метеорологічних станцій, радарів, літаків і супутників.

У бази даних метеопказники потрапляють уже в RWD-виді, але наявність різних баз із різними джерелами метеоданих призводить до необхідності повторного оброблення вже обробленої інформації для подальшого її використання в ML&DL-моделях. Для навчання, тестування та валідації ML&DL-моделей обсяги інформації мають бути достатніми для виявлення тенденцій у даних. Показники, зібрані з відкритих метеоресурсів за допомогою програмного інтерфейсу застосунку (*Application Programming Interface, API*), вимагають спеціального оброблення, оскільки формати даних і методи доступу змінюються, що ускладнює інтеграцію та стабільність метеопказників. Проблемою є об'єднання різної метеорологічної інформації в єдину базу, здатну адаптуватися до нових джерел без втрати якості даних.

Метою роботи є аналіз можливостей наявних платформ використання ML&DL-моделей для прогнозування метеоумов і створення платформи для прогнозування метеоумов, яка має гібридну полегшену архітектуру, здатна збирати різноманітні метеопказники з різних метеорологічних API-ресурсів і з локальної наземної метеостанції, зберігати метеопказники в документо-орієнтованій базі даних; аналізувати метеоінформацію та прогнозувати метеоумови відповідно до певної геолокації за допомогою ML&DL-моделей.

Матеріали й методи

Виконання аналізу наявних DSML-платформ дало змогу виокремити властивості, які є важливими щодо платформи прогнозування метеоумов для певної геолокації:

1) залучення структурованої та неструктурованої інформації (текст, зображення, відео, аудіо та геопросторові показники) з безкоштовних баз метеоданих і файлових сховищ, які можуть бути розташовані локально або в хмарі, а також отримання інформації від локальної метеостанції;

2) розміщення отриманих метеоданих у сучасній документо-орієнтованій базі, що дає змогу зберігати гетерогенні метеопказники;

3) використання, створення та оцінювання за відповідними метриками гібридних ML&DL-моделей і за допомогою популярних інструментів їх досліджень переважно в безкоштовних хмарних середовищах;

4) розгортання, розміщення та обслуговування компонентів платформи на сервері, який забезпечує

захищений доступ і віддалене керування компонентами платформи;

5) забезпечення інтерфейсу з низьким вмістом коду, придатного для неекспертів у сфері оброблення інформації, але експертів із прогнозування метеоумов;

6) забезпечення кодового інтерфейсу для науковців з оброблення інформації для доступу до даних, їх підготовки й розміщення в базі, розроблення ML&DL-моделей та публікації;

7) управління життєвим циклом ML&DL-моделей після розгортання для перенавчання та адаптації моделей;

8) створення передумов використання на проєктованій платформі SLM, що підтримує оброблення гетерогенних метеопказників на рівні граничних обчислень.

У роботі запропоновано формування платформи на основі HLWA-архітектури для підготовки даних та прогнозування. Ця платформа забезпечує важливі властивості платформи прогнозування метеоумов для певної геолокації. Вагомим компонентом запропонованої платформи є *Node-RED*, що інтегрує інструменти керування компонентами платформи для збирання, попереднього оброблення, збереження інформації та прогнозування на основі ML&DL-моделей.

Платформа побудована на сервері *AWS Lightsail*, що втілює архітектуру захищеного хостингу та забезпечує доступ до віртуальних машин і розгортання компонентів платформи. *AWS Lightsail* є приватним віртуальним сервером (*Virtual Private Server, VPS*), що уможливує просте розгортання віртуальних машин, стабільність і масштабованість за умови доступної ціни. Використання *AWS Lightsail* потребує реєстрації (рис. 1). VPS має переваги щодо спільного хостингу, оскільки дає змогу ефективніше контролювати, налаштовувати, масштабувати ресурси та забезпечує більш надійну безпеку. *Lightsail* працює на базі *Amazon Web Service* та підтримує різноманітні конфігурації для задоволення потреб оброблення метеоданих. У запропонованій платформі прогнозування метеоумов сервер працює під управлінням ОС *Ubuntu* з 1 GB оперативної пам'яті та 40 GB SSD. Для безпечного доступу до сервера *Lightsail* використовується мережевий протокол SSH (*Secure Shell*), що дає змогу адміністратору виконувати команди з віддаленого комп'ютера. Протокол SSH забезпечує захист інформації, а завдяки клієнтській програмі *Putty* з віддаленого комп'ютера можна керувати, наприклад, установленням необхідних компонентів, таких як *Node-RED*.

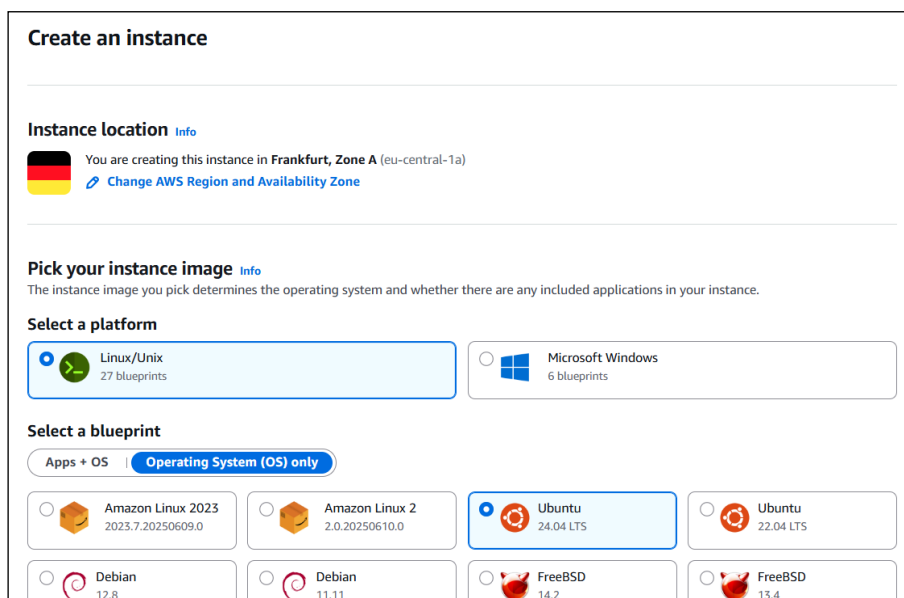


Рис. 1. Етап створення та налаштування віртуального сервера *AWS Lightsail*

Відповідно до HLWA-архітектури платформи, запропонованій у цій роботі, передбачено два основних компоненти, що розгортаються на сервері *AWS Lightsail*.

Компонент *Reanalysis Weather Data Retrieval* реалізовано як програму *Python*, яка автоматично запускається щоденно за визначеним часом доби планування завдань *crontab*, що є системним ресурсом ОС *Ubuntu*, та забезпечує налаштування режиму отримання метеопказників із відкритих API-метеоресурсів.

Наприклад, якщо необхідно порівняти результати прогнозів метеопказників з відкритих API-метеоресурсів з оновленими результатами

вимірювання метеопказників з тих самих відкритих API-метеоресурсів за попередню годину, можна налаштувати виклик програми *Python* для запуску двічі:

- 1) о 13:00 UTC запускається програма на отримання прогнозованих метеопказників по всіх AP-ресурсах на наступну годину для певної геолокації (з параметром *forecast* для виклику запитів);
- 2) о 14:00 UTC запускається програма на отримання вже уточнених (оновлених) метеопказників по всіх API-ресурсах на попередню годину для певної геолокації (з параметром *current*).

У цьому разі налаштований файл конфігурації *crontab* для періодичного запуску програми *Python* на сервері *AWS Lightsail* має вигляд, зображений на рис. 2.

```
00 13.*.*.* /home/ubuntu/weather-forecasting/.venv/bin/python3 /home/ubuntu/weather-forecasting/main.py --task forecast >> /home/ubuntu/weather-forecasting/main.log 2>&1¶
00 14.*.*.* /home/ubuntu/weather-forecasting/.venv/bin/python3 /home/ubuntu/weather-forecasting/main.py --task current >> /home/ubuntu/weather-forecasting/main.log 2>&1¶
```

Рис. 2. Текст файлу *crontab*

Компонент *Reanalysis Weather Data Retrieval* опитує відкриті API-метеоресурси з метою відбору необхідних для певного прогнозування історичних RWD-метеоданих та відбору значень метеопараметрів, які метеоресурси пропонують як результати прогнозування. Відкриті метеоресурси попередньо проаналізовано та обрано. Основні метеопказники, такі як температура, вологість, швидкість вітру, ультрафіолетовий індекс і тиск, отримуються асинхронно з використанням бібліотеки *asyncio*.

Зібрані відомості передаються за допомогою бібліотеки *paho-mqtt* через протокол MQTT у режимі реального часу до другого компонента *Node-RED*, що розгортається на сервері *AWS Lightsail*.

Node-RED є платформою з відкритим кодом і дає змогу інтегрувати метеодані з різних джерел, а також створювати потоки оброблення метеоданих за допомогою зручного графічного інтерфейсу. *Node-RED* налаштовується на сервері *AWS Lightsail* у глобальному файлі *settings.js*,

де також створюються облікові записи з різними правами доступу.

Для безперебійної роботи *Node-RED* використовується *tmux*, термінальний мультиплексор, який дає змогу паралельно запускати декілька процесів на одному сервері. Це забезпечує можливість збереження та продовження роботи в разі розриву з'єднання із сервером.

Основними компонентами *Node-RED* є вузли (*Nodes*), що забезпечують функціональність окремих етапів оброблення метеоданих. Вузли згруповані за категоріями, зокрема *common*, *function*, *network*, *storage* тощо, що полегшує використання. Наприклад, вузли категорії *network* дають змогу отримувати інформацію із зовнішніх джерел за допомогою HTTP-запитів або MQTT-протоколу.

Для контролю процесу оброблення використовуються *Node-RED* вузли *debug*, що забезпечують перевірку стану метеоданих на різних етапах. Результати прогнозування та метрики якості прогнозу користувач системи може отримати на панелі *Dashboard*.

Вузли в *Node-RED* забезпечують з'єднання між такими компонентами запропонованої платформи:

1) *Reanalysis Weather Data Retrieval*, що отримує значення метеопараметрів з API відкритих джерел метеоданих;

2) *Local Observation Weather Data*, який отримує числові значення метеопараметрів з локальної метеостанції та за протоколом MQTT передає компоненту *Weather Data Preprocessing* для підготовки інформації до використання компонентом оброблення метеоданих;

3) *Weather Data Preprocessing*, що забезпечує оброблення метеопараметрів (очищення інформації, її нормалізацію, агрегацію та аналіз її додаткових властивостей), передачу метеоданих до компонента *MongoDB Atlas*, передачу параметрів налаштування ML&DL-моделей у компонент ML&DL для прогнозування метеоумов та отримання результатів прогнозування й значень метрик для відтворення результатів користувачеві;

4) *MongoDB Atlas*, який дає змогу зберігати метеоінформацію різних типів у документо-орієнтованій базі даних;

5) ML&DL (*AWS SageMaker AI*), що виконує навчання, тестування та валідацію ML&DL-моделей прогнозування метеоумов.

Безпосередньо в *Node-RED* реалізовано у вигляді потоків компонент *Weather Data Preprocessing*, *function*-вузли якого сформовано на *JavaScript*. Фрагменти налаштованих потоків у *Node-RED* зображено на рис. 3–5, а призначення вузлів описано в табл. 1.

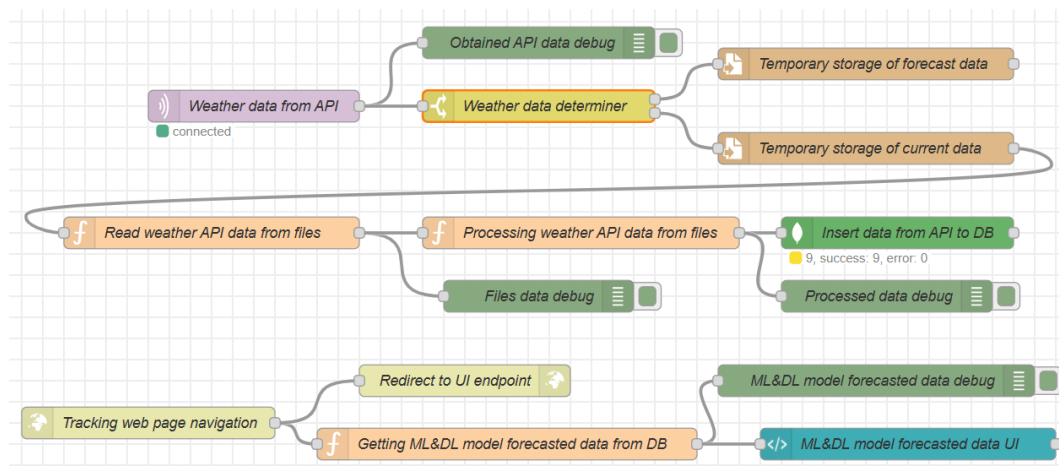


Рис. 3. Потік *Node-RED* оброблення інформації з API-метеоресурсів

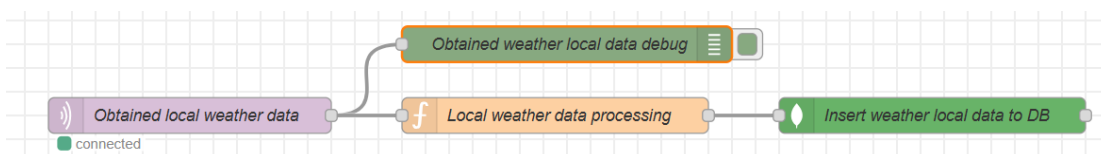


Рис. 4. Потік *Node-RED* оброблення метеоданих із локальної метеостанції

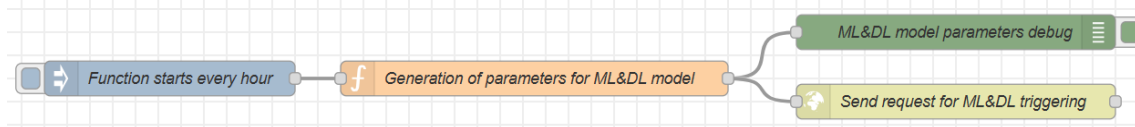


Рис. 5. Потік Node-RED налаштування та передачі гіперпараметрів моделі прогнозування до компонента ML&DL

Таблиця 1. Опис вузлів Node-RED як компонента запропонованої платформи

Вузол Node-RED	Призначення вузла
Weather data from API	MQTT-брокер, що отримує метеопказники з API-ресурсів.
Obtained API data debug	Перегляд метеопказників із зовнішніх API-ресурсів.
Weather data determiner	Аналіз метеопказників і визначення властивостей інформації для подальшого оброблення.
Temporary storage of forecast data	Тимчасове збереження метеопказників прогнозу з API-ресурсів у файлі.
Temporary storage of current data	Тимчасове збереження метеопказників поточного стану із API-ресурсів у файлі.
Read weather API data from files	Інтеграція метеопказників в єдиному JavaScript-об'єкті.
Files data debug	Перегляд об'єднаних метеопказників із файлів.
Processing weather API data from files	Попереднє оброблення об'єднаних метеопказників.
Processed data debug	Перегляд оброблених метеоданих.
Insert data from API to DB	Збереження оброблених метеопказників з API в базі даних MongoDB Atlas.
Tracking web page navigation	Відстеження переходів користувачів на Web-сторінку.
Redirect to UI endpoint	Відповідь користувачеві та перенаправлення користувача на Web-сторінку огляду результатів прогнозування та метрик моделі.
Getting ML&DL model forecasted data from DB	Отримання останніх спрогнозованих метеопказників із MongoDB Atlas.
ML&DL model forecasted data debug	Перегляд отриманих спрогнозованих метеопказників із MongoDB Atlas.
ML&DL model forecasted data UI	Опис структури Web-сторінки відтворення результатів прогнозування метеопказників із MongoDB Atlas.
Obtained local weather data	MQTT-брокер, що отримує метеопказники з локальної метеостанції.
Obtained local weather data debug	Перевірка метеопказників з локальної метеостанції.
Local weather data processing	Попереднє оброблення метеопказників з локальної метеостанції.
Insert weather local data to DB	Збереження метеопказників з локальної метеостанції в базу даних MongoDB Atlas.
Function starts every hour	Запуск роботи функції генерації параметрів щогодини для роботи ML&DL-моделі.
Generation of parameters for ML&DL model	Генерація параметрів для запуску ML&DL-моделей.
ML&DL model parameters debug	Перегляд інформації, отриманої у відповідь на POST-запит.
Send request for ML&DL triggering	POST HTTP-запит для запуску ML&DL-моделі з параметрами.

Одним із ключових компонентів платформи є база даних *MongoDB Atlas*. Завдяки своїй гнучкій схемі зберігання *MongoDB Atlas* дає змогу зберігати неструктуровану або напівструктуровану інформацію, отриману з різноманітних API без необхідності попередньої трансформації в жорстко визначений формат. Такий підхід спрощує оброблення даних, які можуть мати різні структури залежно від джерела. *MongoDB Atlas* підтримує динамічне оновлення схеми, що є надійним інструментом для об'єднання інформації з різних джерел, навіть якщо формат вихідних даних змінюється з часом. Крім того, *MongoDB Atlas* забезпечує високопродуктивний доступ до інформації завдяки вбудованій індексації та можливостям горизонтального масштабування, що допомагає зберігати великі обсяги метеопказників і швидко обробляти запити навіть у режимі реального часу.

MongoDB є зручною базою даних для зберігання метеоданих, адже структура JSON-документів бази дає змогу масштабувати систему та адаптувати формат інформації. Для з'єднання з *MongoDB Atlas* в *Node-RED* налаштовано спеціальний вузол, що автоматично записує зібрані дані в *MongoDB Atlas* – хмарний сервіс, який підтримує реплікацію та моніторинг інформації в реальному часі.

Оброблена інформація надходить з *Node-RED* у форматі JSON-документів та зберігається в базі даних *MongoDB* під назвою *weather-forecasting*, а саме в колекцію *historic-data* (рис. 6).

Колекція в *MongoDB Atlas* є набором документів, що мають схожу структуру та зберігаються в межах однієї бази даних. Колекція подібна до таблиці в реляційних базах даних, проте документи в одній колекції можуть розрізнятися за своєю структурою.

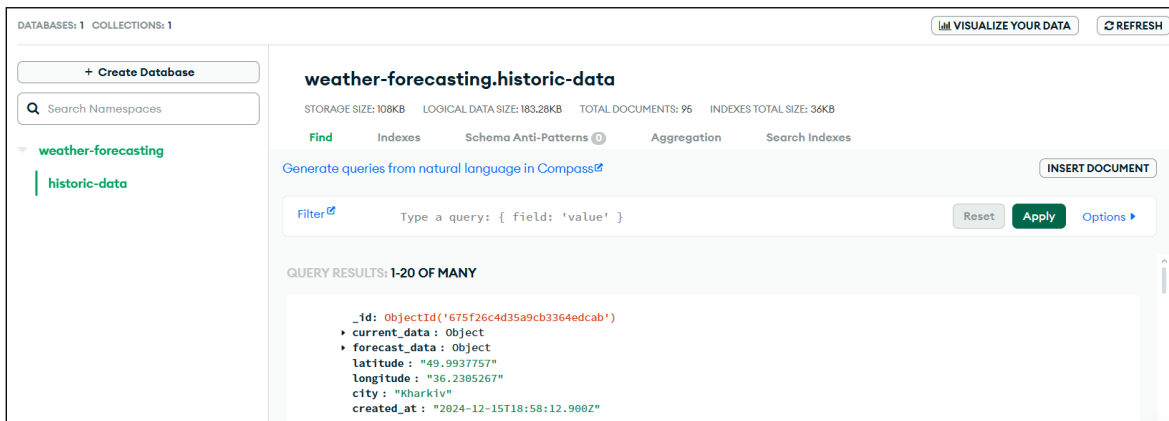


Рис. 6. Збереження метеопоказників у колекції *MongoDB Atlas*

MongoDB Atlas має свої обмеження, але вони не є суттєвими для використання бази з метою збереження метеопоказників.

Можливості безкоштовного збереження метеоінформації у *MongoDB Atlas* обмежені 512 МБ, але її збереження є оптимізованим, а запити до *MongoDB Atlas* можуть використовувати вбудовані функції, що дають змогу обробляти метеодані з огляду на час і відсутні значення. Обмеження на вхідний та вихідний трафік (10 GB вхідного та 10 GB вихідного трафіку за 7-денний період) можуть бути зняті внаслідок переходу до платних кластерів у зв'язку з доступністю вертикального масштабування. Перевагою використання *MongoDB Atlas* є також можливість вибору хмарного провайдера, наприклад AWS.

Одним із потужних сервісів *Amazon Web Services* для розроблення, навчання та розгортання ML&DL-моделей є хмарна платформа *AWS SageMaker AI*. Сервіс надає різноманітні інструменти та гнучке середовище для роботи з ML&DL-моделями, які можуть бути використані з метою прогнозування метеоумов для певної геолокації. *AWS SageMaker AI* підтримує автоматизоване масштабування, оптимізацію гіперпараметрів та інтеграцію з потоковими метеоданими, що робить його ефективним рішенням для експериментів у сфері AIoT та аналізу кліматичних змін.

Використання як компонента ML&DL-сервісу *AWS SageMaker AI* дає змогу спростити процеси створення, навчання та розгортання моделей прогнозування. Запит до історичних метеоданих у *MongoDB Atlas* з компонента ML&DL є прямим.

Сервіс *AWS SageMaker AI* допомагає обрати та налаштувати певні екземпляри (*instances*) програмних і апаратних інструментів та кількість таких екземплярів; надає різні конфігурації досягнення

мети формування моделі (*endpoint configuration*); дає змогу обрати й налаштувати процес отримання висновків (*inferences*), які забезпечують виконання прогнозування метеоумов за обраними ML&DL-моделями. Кожна з можливостей *AWS SageMaker AI* має певні обмеження як за ресурсами, так і термінами безкоштовного використання. Наприклад, можливе застосування безоплатного рівня *AWS SageMaker AI* протягом перших двох місяців з моменту початку роботи з *AWS SageMaker AI*. Але щодо таких екземплярів програмних інструментів, як блокноти (*Notebooks*), наразі також надається обмежена (відповідно до регіону) кількість годин застосування екземплярів (*instances*) апаратних ресурсів за місяць. Наприклад, *ml.t3.medium* означає використання двох віртуальних CPU, 4 GiB пам'яті, збереження тільки в *Amazon Elastic Block Store (Amazon EBS)* і продуктивність мережі до 5 Gigabit. Після двох місяців або в разі перевищення лімітів усі ресурси необхідно оплачувати за стандартними тарифами AWS, наприклад, оплата *ml.t3.medium* може становити приблизно \$0.06 за годину.

Для збереження інформації та моделей *AWS SageMaker AI* інтегрується з хмарним сховищем об'єктів AWS S3. Інструменти *SageMaker AI* можуть використовувати як локальні сховища на час своєї роботи, або для тривалого збереження інформації застосовувати S3. AWS пропонує безкоштовний рівень використання S3 з параметрами 5 GB стандартного сховища, 2000 запитів GET, 2000 запитів PUT, COPY, POST або LIST на місяць. Ліміти діють упродовж 12 місяців з моменту реєстрації нового облікового запису AWS.

Підтримка життєвого циклу ML&DL-моделей прогнозування метеоумов передбачає етапи збирання

та підготовки метеоданих, обрання або створення власної моделі з подальшим тренуванням моделі, оцінювання та вдосконалення моделі, і, нарешті, її розгортання для прогнозування. Незважаючи на використання в запропонованій платформі компонентів від різних постачальників, що може ускладнювати процес керування життєвим циклом ML&DL-моделей,

платформа на основі HLWA-архітектури має переваги для виконання експериментів з ML&DL-моделями для прогнозування метеоумов.

Модель описаної платформи у вигляді UML-діаграми демонструє автономність компонентів платформи (рис. 7) та їх взаємодію у хмарному середовищі за допомогою різних інтерфейсів (табл. 2).

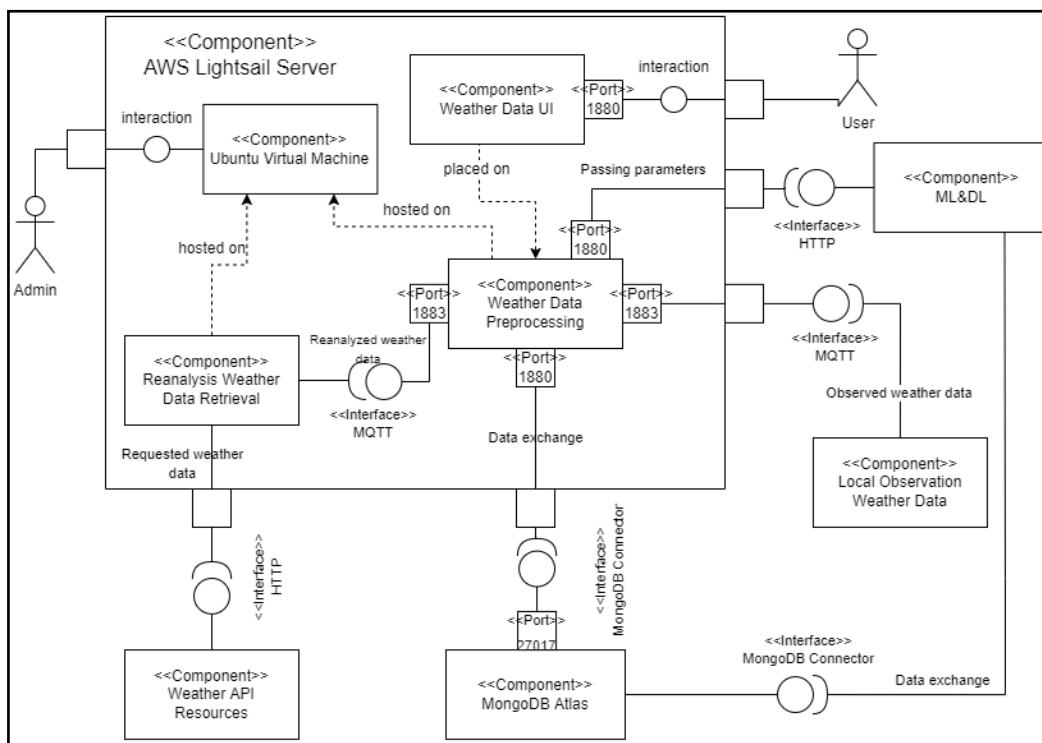
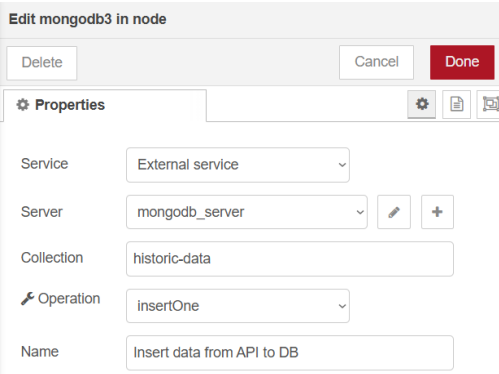


Рис. 7. UML-діаграма компонентів запропонованої платформи на основі HLWA-архітектури

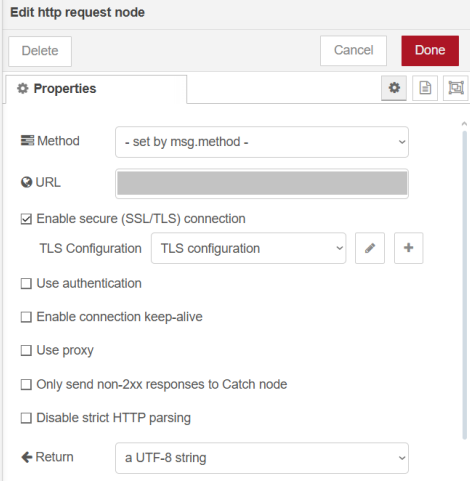
Таблиця 2. Опис інтерфейсів між компонентами платформи на основі HLWA-архітектури

Інтерфейс 1	Опис 2
<<Interface>> HTTP (<<Component>> Reanalysis Weather Data Retrieval → <<Component>> Weather API Resources)	Запит метеопоказників з обраних API-метеоресурсів за методом GET HTTP: async def fetch_weather_data(scraper, url): try: response = scraper.get(url) if response.status_code == 200: return response.json() else: print(f"Unexpected status code from {url}: {response.status_code}") send_email("Weather API Failed", f"Failed to retrieve data from the API: {url}.\nDetails: {response}")) return None except Exception as e: print(f"Error fetching data from {url}: {e}") send_email("Weather API Failed", f"Failed to retrieve data from the API: {url}.\nDetails: {e}")) return None

Продовження таблиці 2

1	2
<<Interface>> MQTT (<<Component>> Reanalysis Weather Data Retrieval → <<Component>> Weather Data Preprocessing)	Передача метеопоказників з API-метеоресурсів до <i>Node-RED</i> за допомогою MQTT-брокера: <pre> mqtt_client = mqtt.Client() mqtt_broker = "localhost" mqtt_topic = "weather_data_topic" def publish_to_mqtt(client, topic, data): try: client.publish(topic, json.dumps(data)) except Exception as e: print(f"Error publishing to MQTT: {e}") </pre>
<<Interface>> MQTT (<<Component>> Local Observation Weather Data → <<Component>> Weather Data Preprocessing)	Передача метеопоказників з локальної метеостанції до <i>Node-RED</i> за допомогою MQTT-брокера (приклад): <pre> String payload = "{\"temperature\": " + String(temperature) + ", \"humidity\": " + String(humidity) + "}"; client.publish(mqtt_topic, payload.c_str()); </pre>
<<Interface>> MongoDB Connector (<<Component>> Weather Data Preprocessing → <<Component>> MongoDB Atlas)	Обмін даними між компонентами <i>Node-RED</i> і <i>MongoDB Atlas</i> – передача підготовленого об'єкта метеопоказників у вузлі <i>Processing of all received data</i> до вузла <i>Insert data from API to DB</i> : <pre> msg.payload = { current_data: sanitizeKeys(currents), forecast_data: sanitizeKeys(forecasts), latitude: currentObject.latitude, longitude: currentObject.longitude, city: currentObject.city, created_at: todayFullData }; </pre> <p>Налаштування вузла <i>Insert data from API to DB</i> у <i>Node-RED</i></p> 
<<Interface>> MongoDB Connector (<<Component>> ML&DL → <<Component>> MongoDB Atlas)	Обмін даними між компонентами ML&DL (<i>SageMaker</i>) та <i>MongoDB Atlas</i> : MONGO_URI = "mongodb+srv://username:password@cluster-url" DATABASE_NAME = "database" COLLECTION_NAME = "collection" <pre> def get_data_from_mongo(): client = pymongo.MongoClient(MONGO_URI) db = client[DATABASE_NAME] collection = db[COLLECTION_NAME] # Fetch data from MongoDB data = pd.DataFrame(list(collection.find({}))) # Drop MongoDB object ID field if exists if "_id" in data.columns: data = data.drop("_id", axis=1) return data </pre>

Продовження таблиці 2

1	2
<<Interface>> HTTP (<<Component>> Weather Data Preprocessing → <<Component>> ML&DL)	<p>Передача параметрів з компонента <i>Node-RED</i> до компонента ML&DL (<i>SageMaker</i>). Використання ML&DL відбувається щогодини.</p> <p>Передача підготовленого об'єкта у вузлі <i>Generation of parameters for ML&DL</i> до вузла <i>Send request with parameters</i>:</p> <pre>msg.method = "POST"; msg.url = "Amazon API url"; msg.payload = JSON.stringify({}); return msg;</pre> <p>Налаштування вузла <i>Send request with parameters</i> в <i>Node-RED</i> для відправлення HTTP-запиту на <i>API endpoint</i></p> 

Результати дослідження

Розвиток платформ щодо науки про дані та машинне навчання (*DSML platforms*) стимулює створення предметно-орієнтованих платформ для різних напрямів використання ML&DL-моделей. У роботі вперше запропоновано модель платформи автоматичної підготовки метеоінформації та прогнозування метеоумов на основі HLWA-архітектури. HLWA-архітектура втілює якості, необхідні для проведення дослідницьких експериментів з різними інструментами аналізу метеоданих. Запропонована платформа містить компоненти, що дають змогу отримувати метеопказники будь-яких типів з різних джерел, зберігати в документо-орієнтованій базі даних і прогнозувати метеоумови за допомогою ML&DL-моделей. Платформа спроможна отримувати метеопказники температури, вологості, тиску, швидкості та напрямку вітру, атмосферних явищ та інших метеопараметрів для конкретної геолокації за допомогою API-метеоресурсів відкритого доступу, а також метеоінформацію від локальної метеостанції.

Інтеграція різних за типом і джерелами отримання метеопказників, їх збереження у створеній базі даних *MongoDB Atlas*, подальша підготовка метеоінформації для використання ML&DL-моделей щодо прогнозування та, зрештою, візуалізація результатів прогнозування виконуються на одній платформі.

Запропонована платформа, створена на захищеному сервері *AWS Lightsail. Node-RED*, що розгортається на сервері *AWS Lightsail*, дає змогу інтегрувати в платформу результати роботи інструментів отримання та збереження метеоданих, а також результатів інструментів прогнозування. Інструменти оброблення метеопказників здебільшого розміщуються в хмарах постачальників інструментів, що забезпечує компактність практично безкоштовного розміщення інформації за етапами її оброблення. Запропонована платформа розв'язує проблеми, що виникають під час прогнозування метеоумов за допомогою сучасних ML&DL-моделей – обмеженість в інтеграції різних джерел метеоданих та їх оброблення для прогнозування локальних метеоумов; складність створення єдиного інструменту для ефективного збереження та доступу

до історичних метеопказників у масштабованих базах даних, таких як *MongoDB Atlas*, та автоматизації налаштування регулярних завдань для збирання та оброблення метеоданих з огляду на часові пояси та різні частоти оновлення інформації.

Для формування платформи в цій роботі використано мову програмування *Python* та *Python*-бібліотеки для роботи з інформацією. У *Node-RED* для формування вузлів застосовано *Node-RED*-бібліотеки й мову програмування *JavaScript*. Визначення певних інтерфейсів для взаємодії між компонентами платформи забезпечило можливість формування складних сценаріїв дослідження метеоданих з використанням ML&DL-моделей.

Висновки й перспективи подальшого дослідження

У роботі запропоновано платформу на основі HLWA-архітектури для підготовки метеоданих і прогнозування метеоумов. Платформа забезпечує автоматичний збір метеопказників із безкоштовних API-метеоресурсів і локальної метеостанції за певним розкладом; оброблення метеоданих з метою подальшого використання їх для прогнозування за допомогою ML&DL-моделей; зберігання метеопказників у документо-орієнтованій базі даних, та саме прогнозування локальних метеоумов. Платформа підтримує всі необхідні етапи для виконання

прогнозування метеоумов і є економічною, масштабованою та зручною для проведення експериментів з різними ML&DL-моделями та комбінаціями моделей. Застосування *Node-RED* як інтеграційного середовища дає змогу виконувати потоки оброблення метеопказників, залучаючи до цього процесу різні компоненти платформи, і постійно мати доступ до візуалізації результатів кожного етапу оброблення метеоданих. Суттєвою перевагою платформи є можливість використання як компонентів платформи системи моніторингу метеопказників засобами локальної метеостанції, застосування бази даних *MongoDB Atlas*, що забезпечує надійне сховище з можливістю масштабування та підтримкою JSON-документів, а також упровадження *AWS SageMaker AI*, що виконує навчання, тестування та валідацію ML&DL-моделей прогнозування метеоумов. Завдяки можливостям *AWS Lightsail* платформа є гнучкою і може бути легко адаптована для розширення та використання як нових джерел метеоінформації, так і нових компонентів її оброблення. Перспективним є долучення компонента з SLMs.

Подальший розвиток платформи також передбачає створення різних сценаріїв застосування платформи та уточнення характеристик ML&DL-моделей прогнозування метеоумов, дослідження з якими проводились автономно в *AWS SageMaker AI*.

Список літератури

1. Hersbach H. ERA5 atmospheric reanalysis. URL: <https://climatedataguide.ucar.edu/climate-data/era5-atmospheric-reanalysis> (дата звернення: 1.04.2025)
2. Beucler T. et al. Next-Generation Earth System Models: Towards Reliable Hybrid Models for Weather and Climate Applications. *Atmospheric and Oceanic Physics*. 2023. DOI: <https://doi.org/10.48550/arXiv.2311.13691>
3. Lopez N. Reanalysis Q&As. URL: <https://climate.copernicus.eu/reanalysis-qas#:~:text=Reanalysis%2C%20on%20the%20other%20hand,where%20weather%20observations%20are%20missing> (дата звернення: 1.04.2025)
4. Hersbach H. et al. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*. 2020. No. 146 (730), P. 1999–2049. DOI: <https://doi.org/10.1002/qj.3803>
5. Bauer P. What if? Numerical weather prediction at the crossroads. *Journal of the European Meteorological Society*. 2024. No. 1. DOI: <https://doi.org/10.1016/j.jemets.2024.100002>
6. Redmon M. How Accurate Are Weather Forecasts? 5 Reasons for Inaccuracy. URL: <https://tempest.earth/resources/how-accurate-are-weather-forecasts/#:~:text=It's%20accurate%20about%2090%25%20of,term%20planning%20and%20decision%2Dmaking> (дата звернення: 19.01.2025)
7. Waqas M. et al. Artificial intelligence and numerical weather prediction models: A technical survey. *Natural Hazards Research*. 2024. DOI: <https://doi.org/10.1016/j.nhres.2024.11.004>
8. Willard J. et al. Integrating Physics-Based Modeling with Machine Learning: A Survey. *Computational Physics*. 2022. DOI: <https://doi.org/10.48550/arXiv.2003.04919>
9. Chen L. et al. Machine Learning Methods in Weather and Climate Applications: A Survey. *Applied Sciences*. 2023. No. 13. 12019 p. DOI: <https://doi.org/10.3390/app132112019>

10. The Ultimate Guide to Weather Forecast Models in 2025. URL: <https://climavision.com/resources/the-ultimate-guide-to-weather-forecast-models/> (дата звернення: 1.04.2025)
11. Lam R. et al. Learning skillful medium-range global weather forecasting. *Science*. 2023. No. 382 (6677). P. 1416–1421. DOI: <https://doi.org/10.1126/science.adi233>
12. Zhang H. et al. Machine Learning Methods for Weather Forecasting: A Survey. *Atmosphere*. 2025. No. 16(1). 82 p. DOI: <https://doi.org/10.3390/atmos16010082>
13. Kochkov D. et al. Neural General Circulation Models for Weather and Climate. *Nature*. 2024. No. 632. P. 1060–1066. DOI: <https://doi.org/10.1038/s41586-024-07744-y>
14. Martorana F. et al. A new tool to process forecast meteorological data for atmospheric pollution dispersion simulations of accident scenarios: A Sicily-based case study. *Journal of Sustainable Development of Energy, Water and Environment Systems*. 2021. No. 9(3). 1080377 p. DOI: <https://doi.org/10.13044/j.sdewes.d8.0377>
15. Das S., Nayak P. Integration of IoT- AI powered local weather forecasting: A Game-Changer for Agriculture. *Other Computer Science*, 14 p. 2025. DOI: <https://doi.org/10.48550/arXiv.2501.14754>
16. European Centre for Medium-Range Weather Forecasts. URL: <https://www.ecmwf.int/en/about> (дата звернення: 1.04.2025)
17. National Oceanic and Atmospheric Administration. URL: <https://www.noaa.gov/about-our-agency> (дата звернення: 1.04.2025)
18. Top Websites Ranking. URL: <https://www.similarweb.com/top-websites/science-and-education/weather/> (дата звернення: 1.04.2025)
19. Balabukh V. et al. Possible contamination of Ukraine and neighboring countries by Cs-137 due to a hypothetical accident at the Zaporizhzhia NPP as a consequence of the Russian aggression. *European Geosciences Union General Assembly 2024 (EGU24)*, Vienna, Austria, 2024. URL: <https://meetingorganizer.copernicus.org/EGU24/EGU24-10247.html> (дата звернення: 1.04.2025)
20. Jaffri A. et al. Magic Quadrant for Data Science and Machine Learning Platforms. 2024. URL: https://www.gartner.com/doc/reprints?id=1-2HIEGHWG&ct=240508&st=sb&__hstc=&__hssc=&hsCtaTracking=b9f5a528-32c7-4b6e-8dce-c3d23844ba2f%7C08736e49-9973-4af1-b53e-3ba4c12dd1ff (дата звернення: 1.04.2025)
21. Patel B. 10 Best Machine Learning Platforms in 2025. 2025. URL: <https://www.spaceotechnologies.com/blog/machine-learning-platforms/> (дата звернення: 1.04.2025)
22. Amazon's Free Virtual Cloud Server. URL: <https://aws.amazon.com/free/compute/lightsail/> (дата звернення: 1.04.2025)
23. Demme M. Node-RED in a Unified Namespace Architecture. 2024. URL: <https://flowfuse.com/blog/2024/02/node-red-unified-namespace-architecture/> (дата звернення: 1.04.2025)
24. Structured vs Unstructured Data: An Overview. URL: <https://www.mongodb.com/resources/basics/unstructured-data/structured-vs-unstructured> (дата звернення: 1.04.2025)
25. Amazon SageMaker AI. Guide to getting set up with Amazon SageMaker AI. URL: https://docs.aws.amazon.com/sagemaker/latest/dg/gs.html?icmpid=docs_sagemaker_lp/index.html (дата звернення: 1.04.2025)

References

1. Hersbach, H. "ERA5 atmospheric reanalysis", available at: <https://climatedataguide.ucar.edu/climate-data/era5-atmospheric-reanalysis> (last accessed: 1.04.25)
2. Beucler, T., Koch, E., Kotlarski, S., Leutwyler, D., Michel, A., Koh, J. (2023), "Next-Generation Earth System Models: Towards Reliable Hybrid Models for Weather and Climate Applications". *Atmospheric and Oceanic Physics* DOI: <https://doi.org/10.48550/arXiv.2311.13691>
3. Lopez, N. "Reanalysis Q&As", available at: <https://climate.copernicus.eu/reanalysis-qas#:~:text=Reanalysis%2C%20on%20the%20other%20hand,where%20weather%20observations%20are%20missing> (last accessed: 1.04.25)
4. Hersbach, H. et al. (2020), "The ERA5 global reanalysis", *Quarterly Journal of the Royal Meteorological Society*, No. 146 (730), P. 1999–2049. DOI: <https://doi.org/10.1002/qj.3803>
5. Bauer, P., (2024), "What if? Numerical weather prediction at the crossroads", *Journal of the European Meteorological Society*, No.1. DOI: <https://doi.org/10.1016/j.jemets.2024.100002>
6. Redmon, M. "How Accurate Are Weather Forecasts? 5 Reasons For Inaccuracy", available at: <https://tempest.earth/resources/how-accurate-are-weather-forecasts#:~:text=It's%20accurate%20about%2090%25%20of,term%20planning%20and%20decision%20making> (last accessed: 19.01.25)
7. Waqas, M., Humphries, U., Chueasa, B., Wangwongchai, A. (2024), "Artificial intelligence and numerical weather prediction models: A technical survey", *Natural Hazards Research*. DOI: <https://doi.org/10.1016/j.nhres.2024.11.004>

8. Willard, J., Jia, X., Xu, S., Steinbach, M., Kumar, V. (2022), "Integrating Physics-Based Modeling with Machine Learning: A Survey". *Computational Physics*. DOI: <https://doi.org/10.48550/arXiv.2003.04919>
9. Chen, L., Han, B., Wang, X., Zhao, J.; Yang, W.; Yang, Z. (2023), "Machine Learning Methods in Weather and Climate Applications: A Survey", *Applied Sciences*, No. 13, 12019 p. DOI: <https://doi.org/10.3390/app132112019>
10. "The Ultimate Guide to Weather Forecast Models in 2025", available at: <https://climavision.com/resources/the-ultimate-guide-to-weather-forecast-models/> (last accessed: 1.04.25)
11. Lam, R. et al. (2023), "Learning skillful medium-range global weather forecasting", *Science*, No. 382 (6677), P. 1416–1421. DOI: <https://doi.org/10.1126/science.adi233>
12. Zhang, H., Liu, Y., Zhang, C., Li, N. (2025), "Machine Learning Methods for Weather Forecasting: A Survey", *Atmosphere*, No. 16(1), 82 p. DOI: <https://doi.org/10.3390/atmos16010082>
13. Kochkov, D. et al. (2024), "Neural General Circulation Models for Weather and Climate", *Nature*, No. 632, P. 1060–1066. DOI: <https://doi.org/10.1038/s41586-024-07744-y>
14. Martorana, F., Giardina, M., Buffa, P., Beccali, M., Zammuto, C. (2021), "A new tool to process forecast meteorological data for atmospheric pollution dispersion simulations of accident scenarios: A Sicily-based case study", *Journal of Sustainable Development of Energy, Water and Environment Systems*, No. 9(3), 1080377 p. DOI: <https://doi.org/10.13044/j.sdewes.d8.0377>
15. Das, S., Nayak, P. (2025), "Integration of IoT- AI powered local weather forecasting: A Game-Changer for Agriculture". *Other Computer Science*, 14 p. DOI: <https://doi.org/10.48550/arXiv.2501.14754>
16. "European Centre for Medium-Range Weather Forecasts", available at: <https://www.ecmwf.int/en/about> (last accessed: 1.04.2025)
17. "National Oceanic and Atmospheric Administration", available at: <https://www.noaa.gov/about-our-agency> (last accessed: 1.04.2025)
18. "Top Websites Ranking", available at: <https://www.similarweb.com/top-websites/science-and-education/weather/> (last accessed: 1.04.25)
19. Balabukh, V., Skrynyk, O., Sergiy Bubin, S., Laptev, G. (2024), "Possible contamination of Ukraine and neighboring countries by Cs-137 due to a hypothetical accident at the Zaporizhzhia NPP as a consequence of the Russian aggression", *European Geosciences Union General Assembly 2024 (EGU24)*, Vienna, Austria, Apr. 14–19, available at: <https://meetingorganizer.copernicus.org/EGU24/EGU24-10247.html> (last accessed: 1.04.2025)
20. Jaffri, A., Popa, A., Krensky, P., Hare, J., Bhati, R., Hassanlou, M., Zhang, T. (2024), "Magic Quadrant for Data Science and Machine Learning Platforms", available at: https://www.gartner.com/doc/reprints?id=1-2HIEGHWG&ct=240508&st=sb&_hstc=&_hssc=&hsCtaTracking=b9f5a528-32c7-4b6e-8dce-c3d23844ba2f%7C08736e49-9973-4af1-b53e-3ba4c12dd1ff (last accessed: 1.04.25)
21. Patel, B. (2025), "10 Best Machine Learning Platforms in 2025", available at: <https://www.spaceotechnologies.com/blog/machine-learning-platforms/> (last accessed: 1.04.25)
22. "Amazon's Free Virtual Cloud Server", available at: <https://aws.amazon.com/free/compute/lightsail/> (last accessed: 1.04.25)
23. Demme, M., (2024), "Node-RED in a Unified Namespace Architecture", available at: <https://flowfuse.com/blog/2024/02/node-red-unified-namespace-architecture/> (last accessed: 1.04.25)
24. "Structured vs Unstructured Data: An Overview", available at: <https://www.mongodb.com/resources/basics/unstructured-data/structured-vs-unstructured> (Last accessed: 1.04.2025)
25. "Amazon SageMaker AI. Guide to getting set up with Amazon SageMaker AI", available at: https://docs.aws.amazon.com/sagemaker/latest/dg/gs.html?icmpid=docs_sagemaker_lp/index.html (last accessed: 1.04.25)

Надійшла (Received) 05.04.2025

Відомості про авторів / About the Authors

Петренко Тетяна Григорівна – кандидат технічних наук, доцент, Український державний університет залізничного транспорту, доцент кафедри інформаційних технологій, Харків, Україна; e-mail: petrenko_tg@kart.edu.ua, ORCID ID: <http://orcid.org/0000-0001-6305-7918>

Задорожний Антон Юрійович – аспірант, Український державний університет залізничного транспорту, кафедра інформаційних технологій, Харків, Україна; e-mail: zadorojnyi85@kart.edu.ua, ORCID ID: <https://orcid.org/0009-0000-5044-6068>

Petrenko Tetyana – PhD, associate professor, department of information technology, Ukrainian State University of Railway Transport, Kharkiv, Ukraine.

Zadorozhnyi Anton – PhD student, Ukrainian State University of Railway Transport, department of information technology, Kharkiv, Ukraine.

PLATFORM FOR INTEGRATION OF METEODATES PROCESSING TOOLS AND SERVICES USING ARTIFICIAL INTELLIGENCE

The **subject** of the study is tools, services and platforms for forecasting local weather conditions. The process of forecasting weather conditions for a specific geolocation is quite complex. The sources of forecasting errors are objective reasons that are consequences of the complexity of weather processes, which have always existed, as well as significant climate changes due to global warming. The use of Machine Learning and Deep Learning (ML&DL) models, together with the refinement of the results of classical physical models of the atmosphere, is an important step in increasing the accuracy of forecasting models. Models for forecasting weather conditions are increasingly becoming hybrid, and the data used to train ML&DL models is increasingly diverse and has different sources of origin. Powerful and not always free environments from leading developers are used to transform structured, unstructured and semi-structured weather data and forecast weather conditions. **The purpose of the work** is to analyze the capabilities of existing platforms for using ML&DL models for weather forecasting and to create a platform for weather forecasting that has a hybrid lightweight architecture (Hybrid LightWeight Architecture, HLWA). The HLWA-based platform solves such **problems** as distributing the stages of weather data processing between different providers of tools and services from cloud environments, but at the same time allows integrating resources and processing tools on a single platform. The deployment of tools and services for preparing weather data and forecasting in the work is proposed on the AWS Lightsail server using Node-RED, MongoDB and AWS SageMaker AI. The article uses **methods** for decomposition of weather forecasting processes. The **results** of the research are the creation of a platform model in the form of a UML component diagram with clarification of the properties of each platform component and interfaces. The **conclusion** of the article is the statement that using the proposed platform for studying hybrid weather forecasting models based on ML&DL models is a convenient, economical and promising solution.

Keywords: weather forecasting platform model; artificial intelligence; cloud services.

Бібліографічні описи / Bibliographic descriptions

Петренко Т. Г., Задорожний А. Ю. Платформа для інтеграції інструментів і сервісів оброблення метеоданих засобами штучного інтелекту. *Сучасний стан наукових досліджень та технологій в промисловості*. 2025. № 3 (33). С. 73–87. DOI: <https://doi.org/10.30837/2522-9818.2025.3.073>

Petrenko, T., Zadorozhnyi, A. (2025), "Platform for integration of meteodates processing tools and services using artificial intelligence", *Innovative Technologies and Scientific Solutions for Industries*, No. 3 (33), P. 73–87. DOI: <https://doi.org/10.30837/2522-9818.2025.3.073>