УДК 004.62+551.5 DOI: 10.18664/ikszt.v29i4.320375

ПЕТРЕНКО Т.Г., к.т.н., доцент, доцент кафедри інформаційних технологій, Український державний університет залізничного транспорту
ЗАДОРОЖНИЙ А.Ю., аспірант, Український державний університет залізничного транспорту

Підготовка метеорологічних даних для навчання моделі штучного інтелекту

Прогнозування метеоумов за допомогою сучасних моделей штучного інтелекту потребує попередьої обробки великої кількості даних різних типів. Перед використанням метеоданих для навчання моделей, метеодані мають бути зібрані, об'єднані та структуровані. У статті розглянута підготовка метеоданих для подальшого використання при навчанні графових нейронних мереж з метою прогнозування метеоумов - визначення джерел історичних метеоданих, засобів збору даних, обробки та зберігання у базі даних. Описано властивості метеоресурсів даних вільного використання, з яких здійснюється збір метеоданих, а також підхід до підготовки даних щодо використання. Особлива увага приділяється аспектам забезпечення різноманіття даних для забезпечення навчання моделей штучного інтелекту та росту прогностичних можливостей моделей. Наведено розрахунок статистичних характеристик метеопоказнику «температура» засобами Python. Запропоновано використання бази даних MongoDB Atlas для збереження різноманітних метеоданих.

Ключові слова: *API метеоресурсів, попередня обробка метеоданих, статистичний аналіз, бібліотеки Python, MongoDB Atlas*

Вступ

Прогнозування метеоумов є однією з важливих задач людства, оскільки певні очікування безпосередньо впливають на людське життя та діяльність.

Метеорологічні дані можуть бути отримані із різних джерел. Дані можуть мати різні формати, масштаби та часові інтервали. Наприклад, поточні дані, отримані з локальних метеорологічних станцій, можуть мати високу просторову роздільність, але бути обмеженими за часовими рамками, тоді як супутникові дані можуть покривати великі території, але з меншою просторовою роздільністю. Різноманітність даних метеоумов призводить до неструктурованості сирих метеоданих. Структурування сирих чисельних даних дозволяє подальше використання методів Машинного Навчання (Machine Learning, ML) для аналізу даних, але призводить до часткового втрачання різноманітності.

Історичні метеодані повинні бути зібрані та оброблені засобами, які зберігають різноманітність даних за рахунок перетворення даних у напівструктуровану форму. Далі перетворені дані можна використовувати для прогнозування за допомогою сучасних методів ML. Обробка метеоданих включає етапи очистки сирих даних, перетворення і нормалізації, щоб забезпечити сумісність даних з сучасними методами прогнозування. Метеодані повинні зберігатися в спеціалізованих базах даних, які забезпечують швидкий доступ до великих обсягів інформації різного типу і дозволяють постійне оновлення для покращення моделей.

У статті розглядаються джерела отримання історичних метеоданих вільного використання з сайтів мережі Internet, засоби виявлення тенденцій в метеоданих та обгрунтовано вибір та створення бази даних для зберігання метеоданих. Виконано структурний опис базових параметрів чисельних метеоданих, умов отримання метеоданих та запропоновано алгоритм визначення зовнішнього джерела метеоданих, дані якого відповідають суттєвим властивостям метеоданих за певною геолокацією. Запропонований підхід є першим кроком і для формування синтетичних даних для навчання графових нейронних мереж (Graph Neural Networks, GNNs), які мають властивості близькі до орігінальних даних, але не мають обмежень притаманних орігінальним даним.

Постановка проблеми, аналіз досліджень та публікацій

Проблемою використання історичних метеоданих при навчанні методів ML є незадовільна попередня обробка метеоданих.

Попередня обробка даних має вирішальне значення для результативності навчання GNN з метою покращення прогнозування метеоумов.

Якісні прогнози погоди в основному базуються на методах Чисельного Прогнозування Погоди (Numerical Weather Prediction, NWP) [1]. NWP використовує складні математичні моделі, які описують динаміку атмосфери, океанів та поверхневих процесів. Чисельні моделі потребують великої кількості даних, які отримані в різних точках земної кулі. Метеодані збираються за допомогою супутників, метеорологічних станцій, радарів і літаків. Наступна обробка даних виконується в хмарних сховищах на суперкомп'ютерах і забезпечує відносно точні прогнози, але розрахунки є надзвичайно ресурсномісткими та дорогими.

Швидкий розвиток Штучного Інтелекту (Artificial Intelligence, AI), особливо методів ML, забезпечив появу нових підходів до прогнозування метеоумов, що потенційно можуть зробити процес прогнозування менш затратним і більш точним [2].

Одним із найбільш перспективних напрямків є застосування глибоких нейронних мереж, зокрема GNN [3,4]. GNNs моделі можуть краще відображати складні просторові та часові взаємодії між різними компонентами кліматичної системи, що дозволяє покращити точність прогнозів.

Однак, критичною передумовою для успішного застосування GNNs у прогнозуванні є наявність великої кількості якісних даних про минулі метеоумови. Сучасні засоби обробки даних дозволяють використання метеоданих різного ступеню структурованості [5-8].

Історичні метеодані також відіграють важливу роль у тренуванні та вдосконаленні моделей AI, оскільки вони дають змогу не тільки навчити систему розпізнавати певні закономірності та тенденції в метеоумовах (патерни), що можуть виникати за різних кліматичних сценаріїв, а і покращувати якість прогнозу.

Джерела історичних погодних даних включають глобальні метеорологічні бази даних, такі як бази даних Європейського центру середньострокових прогнозів погоди (European Centre for Medium-Range Weather Forecasts, ECMWF) [9] та Національного управління океанічних і атмосферних досліджень США (National Oceanic and Atmospheric Administration, NOAA) [10]. Історичні метеодані можуть охоплювати періоди від кількох десятків до сотень років і включати різноманітні параметри, які відображають погодні явища на різних рівнях атмосфери. Доступність метеоданих з відкритих метеоресурсів сайтів метеокомпаній відкриває можливість отримання великої кількості даних за допомогою Програмного Інтерфейсу Додатка (Application Programming Interface, API) [11], які на певному етапі обробки даних можна вважати сирими.

Реально сирими даними є метеодані, які отримані як результат моніторингу із локальних метеостанцій. Агрегація різноманітних метеоданих як із зовнішніх, так і локальних джерел [4], потребує розміщення даних в спеціальних базах даних [6-8].

Підготовка даних для подальшого використання моделей AI [12-14], особливо аналіз властивостей метеоданих [15-20], створює умови для подальшого визначення складних структурних та семантичних залежностей метеоданих за допомогою GNNs [3,4].

Важливість підготовки даних з метою подальшого використання оброблених даних за допомогою моделей AI, складно перебільшити, тому що не підготовлені належним чином дані можуть унеможливити переваги використання моделей AI для прогнозування метеоумов взагалі.

Виділення невирішених проблем в дослідженнях

Незважаючи на значний прогрес у метеопрогнозуванні, існують невирішені питання, пов'язані зі збором, обробкою та використанням метеоданих для навчання моделей AI. Визначені проблеми впливають на точність прогнозів та їх адаптацію до локальних умов. Основні невирішені питання включають наступні аспекти:

1) Питання забезпечення якості та повноти даних залишається актуальним завданням, тому що існує значна залежність точності моделей AI від якості вхідних даних, які використовуються для навчання моделей AI.
2) Проблемою залишається інтеграція метеоданих, що отримані із різних джерел метеоданих, таких як, супутникові знімки, дані із наземних метеостанцій та метеорадіолокаційних систем, дані з повітряних суден і т.д., у єдину систему, яка б дозволяла моделям AI враховувати різноманітність джерел та знижувати похибку прогнозування.
3) Недостатня адаптивність моделей до локальних умов виникає як результат недостатнього врахування специфічних локальних умов. Погодні явища, такі як місцеві грози, тумани або сильні вітри, можуть не враховуватись у глобальних моделях, що призводить до суттєвих похибок у прогнозах. Необхідні подальші дослідження щодо адаптації моделей до мікрометеоумов для забезпечення більш точної і своєчасної інформації для окремих населених пунктів або регіонів.
4) Необхідно продовжувати дослідження з покращення моделювання екстремальних

метеоподій, щоб зменшити ризики та підвищити надійність прогнозів, незважаючи на складність моделювання крайніх погодних явищ, таких як шторми, урагани, сильні опади та інші екстремальні явища.

5) Зниження витрат на обчислювальні ресурси та підвищення ефективності використання великих даних є актуальною задачею для підвищення продуктивності сучасних систем прогнозування метеоумов.

Мета та задачі дослідження

Метою даної роботи є формування підходу та створення системи підготовки метеоданих для подальшого навчання GNN з метою покращення прогнозування метеоумов.

Задачі дослідження включають:

1) Вивчення ресурсів і методів збору метеоданих, що передбачає розгляд різноманітних джерел історичних метеоданих, зокрема різноманітних метеоресурсів та міжнародних метеорологічних баз даних. Частина ресурсів є відкритою для використання за допомогою API сайтів відповідних ресурсів. Треба оцінити якість, точність і доступність даних ресурсів для подальшої обробки і використання.
2) Аналіз методів обробки і зберігання метеоданих, що передбачає визначення ефективних методів обробки великих обсягів метеоданих, включаючи очищення даних, інтеграцію даних із різних джерел і зберігання у базі даних. Треба обрати та створити базу даних, що відповідає вимогам навчання GNNs.

Визначені задачі досліджень спрямовані на створення комплексного підходу до збору та обробки даних, що забезпечить основу для ефективного навчання GNNs та підвищення точності прогнозів метеоумов.

Основна частина

Метою дослідження в роботі є визначення етапів підготовки метеоданих для навчання GNN моделі прогнозування метеоумов (рис.1). Створена програма, яка виконує збір та аналіз метеоданих з різних метеорологічних API ресурсів, а також зберігає дані в базі даних для забезпечення прогнозу метеоумов для певної геолокації.

Обробка метеоданих включає декілька етапів і на кожному етапі можливе використання методів AI:

1) Очистка даних, яка забезпечує усунення пропусків, шумів та аномалій, які можуть спотворити результати.
2) Нормалізація даних, яка перетворює різні за форматами дані, наприклад, різними за одиницями виміру та масштабами, до єдиного формату, що полегшує навчання нейронних мереж.
3) Агрегація даних на різних рівнях, наприклад, збір даних про температуру, вологість або про інші параметри на щоденній або погодинній основі, а не на коротших інтервалах, дозволяє зменшити розмір набору даних і зосередитися на ключових трендах.
4) Аналіз додаткових властивостей даних, що вміщує процеси створення нових атрибутів, перетворення атрибутів, вилучення атрибутів, вибір атрибутів та масштабування атрибутів;

Рисунок 1. Підготовка метеоданих для навчання ML моделей

Створення передумов для зберігання та поповнення метеоданих в сучасних базах даних в реальному часі формує практичну безперервність обробки метеоданих, що забезпечує відповідну якість прогнозу метеоумов.

Для реалізації системи підготовки даних в роботі створена програма на мові програмування Python, що використовує бібліотеку Requests для роботи з API метеоресурсами шляхом обробки HTTP-запитів та бібліотеку Pandas для статистичного аналізу даних. Програма опитує кожен з обраних API і отримує такі основні метеопоказники, як температура, вологість, тиск та швидкість вітру. Метеодані інтегруються з кожного API в єдину структуру та виконується їх подальша

обробка та збереження в базі даних MongoDB Atlas.

Метеодані з API метеоресурсів є напівструктурованими, оскільки містять як структуровані елементи, наприклад, значення координат, часу, масиви числових значень, так і менш структуровані елементи, наприклад, текстову інформацію щодо метеоумов. Для зберігання таких даних необхідне рішення, що підтримує гнучкість структури даних та ефективно працює з великими обсягами інформації. Тому було обрано MongoDB Atlas, який дозволяє зберігати дані у вигляді JSON-об'єктів.

Кожен документ у MongoDB представляє собою окремий набір метеоданих для певної геолокації та часу. Основні поля документа включають географічну інформацію (координати), час отримання даних та метеопараметри (температура, вологість, тиск, швидкість вітру тощо). Для багатовимірних значень, що описують погодні умови, MongoDB Atlas дозволяє зберігати масиви числових значень, що спрощує роботу з ними та подальший аналіз. Крім того, можливе зберігання даних у якості таких об'єктів, як зображення, аудіо, відео, файли документів та інших форматів файлів.

Привілеї MongoDB Atlas:

1) Гнучка структура зберігання: MongoDB дозволяє зберігати дані в документно-орієнтованому форматі, що спрощує інтеграцію даних з різних API з різними структурами. Це дозволяє зберігати метеорологічні параметри як JSON-документи з можливістю динамічного додавання нових полів.
2) Масштабованість: MongoDB Atlas надає можливість горизонтального масштабування, що забезпечує обробку великої кількості даних без втрати продуктивності. Це важливо для системи, яка постійно збільшує обсяг даних за рахунок підключення нових джерел.
3) Швидкий доступ до даних: MongoDB Atlas оптимізована для швидкого пошуку і фільтрації даних, що дозволяє оперативно отримувати метеопоказники для аналізу та прогнозування.
4) Реплікація та резервування: хмарна інфраструктура MongoDB Atlas забезпечує надійне зберігання даних завдяки реплікації і автоматичному резервуванню, що дозволяє мінімізувати ризики втрати даних.
5) Підтримка векторних даних: MongoDB Atlas має можливість працювати з векторними даними, що дозволяє зберігати та ефективно обробляти багатовимірні показники метеорологічних умов. Це важливо для складного аналізу прогнозів погоди.

Запропонована в роботі система підготовки метеоданих складається з декількох компонентів:

1) Компонент збирання даних через API кожного із ресурсів забезпечує отримання набору метеопоказників для конкретної геолокації. Кожен API повертає різні набори даних, тому важливо привести їх до спільного формату. Відповідні API ресурси можуть надавати інформацію про температуру, вологість, тиск, швидкість та напрямок вітру, атмосферні явища та інші метеопараметри.
2) Компонент інтеграції даних виконує стандартизацію отриманих даних з API ресурсів шляхом перетворення до спільного формату за допомогою існуючих бібліотек Python. Всі показники нормалізуються за типами та одиницями вимірювання. Під час інтеграції виконується перевірка даних на повноту та коректність: відсутні або некоректні значення обробляються або фільтруються. Це дозволяє об'єднувати дані з різних джерел у єдиний набір для подальшого аналізу та обробки.
3) Компонент обробки зібраних даних агрегує та обчислює акумульовані показники для кожного метеопараметра. На цьому етапі застосовуються алгоритми для згладжування коливань і виявлення трендів. Мета обробки – підвищити точність прогнозів, зменшити вплив відхилень і об'єднати дані з різних джерел для подальшого аналізу.
4) Компонент зберігання даних розміщує та оновлює дані у створеній базі даних MongoDB Atlas, яка є гнучкою NoSQL базою даних. Кожен запис в базі MongoDB Atlas містить інформацію про метеопоказники у форматі документа, що дозволяє зберігати структуровані, напівструктуровані та неструктуровані дані та формувати вектори даних.

В роботі розглянуто на прикладі запропонований підхід, який дозволяє визначити серед виділених безкоштовних API ресурсів метеоумов, такий ресурс, що демонструє якісне визначення тенденцій в метеоданих, а результати прогнозування за таким ресурсом можуть бути інтегровані з даними локальної метеостанції для подальшого навчання GNN. Було виконано аналіз доступності даних, діапазону значень метеопоказників, терміну прогнозування та інших характеристик.

Основними метеопоказниками, на основі яких проводились визначення API ресурса з найбільш репрезентативними даними, обрано значення температури, вологості, тиску та швидкості вітру. Визначені показники названо основними, тому що такий набір показників зустрічається в більшості розглянутих API метеоресурсів.

Агрегація метеоданих в API ресурсах визначена в групах: поточні (далі C-base), погодинні (далі H-base) та поденні показники (далі D-base).

C-base показники: Температура, Вологість, Тиск, Швидкість вітру.

H-base показники: Температура, Вологість, Тиск, Швидкість вітру.

D-base показники: Максимальна температура, Мінімальна температура, Середня вологість, Максимальна швидкість вітру, УФ-індекс.

Опис обраних API метеоресурсів наведено у табл.1. Доступність API метеоданих (табл.1), є безкоштовною з обмеженнями, але для API ресурсів, які помічені *, доступність обмежена 30 добами. Прогнозування метеопоказників на один і той самий час доби по кожному з API ресурсів для визначеної геолокації є ключовим аспектом точності даних. Це дозволяє порівнювати різні джерела в реальному часі та виявляти можливі розбіжності між метеоданими.

Таблиця 1

Опис API метеоресурсів

Назва API ресурса	Метеопоказники, що надаються API ресурсом метеоданих	Доступний план використання API ресурса
WeatherAPI (https://www.weatherapi.com/)	Поточні: C-base, УФ-індекс, температура вітру, опади, хмарність, відчувається як, видимість	Після реєстрації надається Pro Plus план на 14 днів, який надає наступні привілеї: 5 мільйонів запитів на місяць, прогноз погоди на 14 днів вперед, історичний прогноз погоди за останні 365 днів. Після спливу 14 днів надається Free план, який надає наступні привілеї: 1 мільйон запитів на місяць, прогноз погоди на 3 дні вперед, історичний прогноз погоди за останні 7 днів.
	Погодинні: H-base, УФ-індекс, температура вітру, опади, шар снігу, хмарність, відчувається як, індекс тепла, шанс дощу, шанс снігу, видимість	
	Поденні: D-base, УФ-індекс, середня температура, загальна сума опадів, середня видимість, шанс дощу, шанс снігу, загальний шар снігу	
Weatherbit (https://www.weatherbit.io/)	Поточні: C-base, УФ-індекс, індекс якості повітря, хмарність, точка роси, кількість опадів, кількість снігу, видимість	Після реєстрації надається Business Trial план на 21 днів, який надає наступні привілеї: 1500 запитів на день, поденний прогноз погоди на 16 днів вперед, погодинний прогноз на 240 годин вперед, історичний прогноз погоди за останні 20 років. Після спливу 21 дня надається Free план, який надає наступні привілеї: 50 запитів на день, прогноз погоди на 7 днів вперед, історичний прогноз погоди відсутній.
	Погодинні: H-base, УФ-індекс, хмарність, точка роси, вміст озону, опади, кількість снігу, видимість	
	Поденні: D-base, УФ-індекс, хмарність, точка роси, вміст озону, опади, тиск, кількість снігу, видимість	
Tomorrow.io (https://app.tomorrow.io/)	Поточні: C-base, УФ-індекс, хмарність, точка роси, опади, видимість, інтенсивність снігу	Після реєстрації надається Free план, який надає наступні привілеї: 500 запитів на день, 25 запитів на годину, 3 запити на секунду, прогноз погоди на 5 днів вперед, історичний прогноз за минулу добу.
	Погодинні: H-base, УФ-індекс, хмарність, точка роси, опади, кількість снігу, видимість	
	Поденні: D-base, УФ-індекс, хмарність, точка роси, опади, тиск, кількість снігу, видимість, напрямок вітру	
Visual Crossing (https://www.visualcrossing.com/)	Поточні: C-base, точка роси, кількість опадів, кількість снігу, видимість, хмарність, сонячна радіація, УФ-індекс, погодні	Після реєстрації надається Free план, який надає наступні привілеї: 1000 запитів на день, прогноз погоди на 15 днів вперед,

	умови	історичний прогноз за останні 50 років.
	Погодинні: H-base, точка роси, кількість опадів, кількість снігу, тиск, видимість, хмарність, сонячна радіація, УФ-індекс, погодні умови	
	Поденні: D-base, УФ-індекс, середня температура, точка роси, кількість опадів, кількість снігу, тиск, хмарність, видимість, сонячна радіація	
Open-Meteo (https://open-meteo.com/)	Поточні: C-base, кількість опадів, кількість снігу, хмарність	Після реєстрації надається Free план, який надає наступні привілеї: 10000 запитів на день, прогноз погоди на 16 днів вперед, історичний прогноз з 1940 року.
	Погодинні: H-base, точка роси, кількість опадів, кількість снігу, хмарність, видимість	
	Поденні: D-base, УФ-індекс, кількість опадів, кількість снігу	
Xweather * (https://www.xweather.com/)	Поточні: C-base, точка роси, кількість опадів, кількість снігу, хмарність, видимість, УФ-індекс, погодні умови	Після реєстрації надається Free Trial план, який надає наступні привілеї: 1000 запитів на день, прогноз погоди на 7 днів вперед, історичний прогноз відсутній.
	Погодинні: H-base, точка роси, кількість опадів, кількість снігу, хмарність, видимість, УФ-індекс, погодні умови	
	Поденні: D-base, УФ-індекс, середня точка роси, кількість опадів, кількість снігу, погодні умови	
OpenWeatherMap (https://openweathermap.org/)	Поточні: C-base, хмарність, видимість	Після реєстрації надається Free план, який надає наступні привілеї: 1000000 запитів на місяць, 60 запитів на хвилину, прогноз погоди на 5 днів вперед з трьох годинними інтервалами, історичний прогноз за останні 40 років (платно).
	Погодинні: H-base, кількість опадів, хмарність, погодні умови	
	Поденні: D-base, кількість опадів, хмарність, погодні умови	
Foreca * (https://developer.foreca.com/)	Поточні: C-base, хмарність, видимість, точка роси, кількість опадів, УФ-індекс	Після реєстрації надається Free план, який надає наступні привілеї: 2000 запитів на день, прогноз погоди на 1 день вперед з годинними інтервалами, історичний прогноз відсутній.
	Погодинні: H-base, кількість опадів, хмарність, погодні умови	
	Поденні: D-base, кількість опадів	
Meteosource * (https://www.meteosource.com/)	Поточні: Температура, тиск, швидкість вітру, хмарність, кількість опадів, погодні умови	Після реєстрації надається Free план, який надає наступні привілеї: 400 запитів на день, прогноз погоди на 7 днів вперед, погодинні прогнози на 1 день вперед, історичний прогноз відсутній.
	Погодинні: Температура, швидкість вітру, хмарність, кількість опадів, погодні умови	

	Поденні: Максимальна температура, мінімальна температура, максимальна швидкість вітру, хмарність, кількість опадів, погодні умови	
Meteoblue (https://www.meteoblue.com/)	Поточні: Температура, швидкість вітру	Після реєстрації надається Free план, який надає наступні привілеї: прогноз погоди на 7 днів вперед з годинними інтервалами, історичний прогноз за останні 4 дні.
	Погодинні: Температура, вологість, швидкість вітру, хмарність, кількість опадів, тиск, УФ-індекс	
	Поденні: Максимальна температура, мінімальна температура, максимальна швидкість вітру, максимальна вологість, мінімальна вологість, тиск, УФ-індекс	

Метеопоказники на одну й ту ж саму годину по кожному API ресурсу на певну геолокацію (місто Харків, Україна) наведені в табл.2.

Таблиця 2

Метеорологічні показники з API ресурсів

Назва API	Температура (°C)	Вологість (%)	Тиск (hPa)	Швидкість вітру (км/год)
WeatherAPI	-0.6	82	1034	1.4
Weatherbit	-0.5	84	1017	1.83
Tomorrow.io	-1.06	92.74	1015.34	0.61
Visual Crossing	-0.7	94.31	1034	2.2
Open-Meteo	-1.1	90	1033.5	1.5
XWeather	-1.2	87	1034	1
OpenWeatherMap	-0.22	78	1034	0.54
Foreca	0	-	-	1
Meteosource	-1	-	-	0.4
Meteoblue	-0.11	81	1033.9	0.32

Якщо порівняти один із ключових метеопоказників, наприклад, температуру для відповідної геолокації, зафіксовану в той самий час, коли були зроблені прогнозовані значення, що розміщенні в API ресурсах, можна зробити такі висновки: локальна фактична температура становила -0.5°C (1 грудня 2024 року о 23:00), і наближені прогнозовані дані до цього показника надали сервіси: WeatherAPI – -0.6°C та Visual Crossing – -0.7°C. Найбільш наближеним виявився API ресурс Weatherbit, прогноз якого повністю співпав із фактичною температурою, склавши -0.5°C. Наведені результати аналізу є спрощеними. Використання різних метрик якості результатів прогнозу можливо при наявності повноти бази даних для системи прогнозування.

Підготовка метеоданих для навчання графової нейронної мережі дозволила виявити певні тенденції в метеоданих, використовуючи статистичний аналіз. Враховані аномалії та варіації для покращення моделі прогнозування. В роботі використана Python бібліотека pandas [21].

В табл.3, для прикладу, описано визначення статистичних показників атрибуту «температура» методом describe бібліотеки pandas.

Таблиця 3

Статистичні характеристики атрибуту «температура»

Параметр	Опис параметру	Формула	Опис формули
mean	Середнє арифметичне значення	$\overline{x} = \frac{\sum x}{n}$	$\overline{x}$ – середнє арифметичне, x – значення температури для певного API ресурсу, n – кількість API ресурсів.

std	Стандартне відхилення	$s = \sqrt{\frac{\sum(x-\overline{x})^2}{n-1}}$	s – стандартне відхилення, x – значення температури для певного API ресурсу, $\overline{x}$ – середнє арифметичне, n – кількість API ресурсів.
min	Мінімальне значення	$x_{min} = \min(x_1, x_2, \dots, x_n)$	x_{min} – найменше значення температури серед API даних, $x_1, x_2, \dots, x_n$ – показники температури.
max	Максимальне значення	$x_{max} = \max(x_1, x_2, \dots, x_n)$	x_{max} – найбільше значення температури серед API даних, $x_1, x_2, \dots, x_n$ – показники температури.
25%	Перший квартиль	$Q_1 = \text{x}[0.25\text{n}]$	Q_1 – значення, нижче якого знаходиться 25% температур, визначене як медіана нижньої половини даних, x – значення температури для певного API ресурсу, n – кількість API ресурсів.
50%	Другий квартиль (медіана)	$Q_2 = \text{x}[0.5\text{n}]$	Q_2 – значення, що ділить впорядкований набір температур на дві рівні частини (50% нижче, 50% вище), x – значення температури для певного API ресурсу, n – кількість API ресурсів.
75%	Третій квартиль	$Q_3 = \text{x}[0.75\text{n}]$	Q_3 – значення, нижче якого знаходиться 75% температур, визначене як медіана верхньої половини даних, x – значення температури для певного API ресурсу, n – кількість API ресурсів.

На рис.2 наведено фрагмент Python програми визначення статистичних властивостей метеоданих, на прикладі атрибута «температура».

```python
import pandas as pd

data = {
    "API": [
        "WeatherAPI", "Weatherbit", "Tomorrow.io", "Visual Crossing",
        "Open-Meteo", "XWeather", "OpenWeatherMap", "Foreca", "Meteosource",
        "Meteoblue"
    ],
    "Temperature (°C)":
    [-0.6, -0.5, -1.06, -0.7, -1.1, -1.2, -0.22, 0, -1, -0.11]
}
df = pd.DataFrame(data)
description = df["Temperature (°C)"].describe()
print(description)
```

Рисунок 2. Визначення статистичних властивостей метеоданих

Отримані статистичні характеристики (табл.4) використані для аналізу даних. Наприклад,

мінімальне та максимальне значення дозволяють ідентифікувати крайні випадки (аномалії), стандартне відхилення показує рівень варіативності в даних, а квартилі дозволяють визначити розподіл значень.

Аналіз метеоданих допомагає виявити тенденції та зробити висновки щодо стабільності роботи API ресурсів, що особливо важливо для навчання графової нейронної мережі в реальному часі.

Таблиця 4

Отримані значення статистичних характеристик атрибуту «температура» із 10 API ресурсів

Параметр	Значення
mean	-0.649
std	0.437
min	-1.2
max	0
25%	-1.04
50%	-0.65
75%	-0.29

Висновки

У роботі було розглянуто підхід до збору метеоданих з використанням 10 безкоштовних API метеоресурсів, а також обробки та зберігання метеоданих. Проаналізовано інтеграцію даних з різних джерел та перетворення метеоданих для подальшого аналізу. Особлива увага була приділена нормалізації та перевірці даних для забезпечення точності та повноти. Виконано статистичний аналіз метеоданих.

Створена програма на мові Python, яка забезпечує автоматичну підготовку метеоданих для наступних етапів аналізу та використання.

В роботі обґрунтовано вибір MongoDB Atlas для зберігання метеоданих. MongoDB Atlas дозволяє гнучко працювати з багатовимірними метеопоказниками та ефективно масштабуватися відповідно до обсягів даних. Завдяки можливості зберігання даних у документно-орієнтованому форматі, MongoDB Atlas забезпечує швидкий доступ до необхідних показників для аналізу та прогнозування.

Запропонований в роботі підхід є комплексним відносно інтеграції та обробки метеоданих, що дозволить отримувати більш впевнені результати прогнозів. Майбутній розвиток системи підготовки метеоданих для моделей AI, що використовуються для прогнозування метеоумов, передбачає впровадження нових методів обробки даних, які сприятимуть глибшому виявленню патернів у метеоданих і підвищенню точності прогнозів.

СПИСОК ПОСИЛАНЬ

1) Brotzge J. et al. Challenges and Opportunities in Numerical Weather Prediction. URL: https://journals.ametsoc.org/view/journals/bams/104/3/BAMS-D-22-0172.1.xml (Last accessed: 20.09.2024)
2) Bochenek B., Ustrnul Z. Machine Learning in Weather Prediction and Climate Analyses - Applications and Perspectives. URL: https://www.mdpi.com/2073-4433/13/2/180 (Last accessed: 22.09.2024)
3) Keisler R. Forecasting Global Weather with Graph Neural Networks. URL: https://arxiv.org/abs/2202.07575 (Last accessed: 21.09.2024)
4) Yang O. et al. Multi-modal graph neural networks for localized off-grid weather forecasting. URL: https://arxiv.org/abs/2410.12938 (Last accessed: 1.11.24)
5) Structured vs unstructured data. URL: https://www.ibm.com/think/topics/structured-vs-unstructured-data#:~:text=Storage%3A%20Structured%20data%20is%20stored,databases%2C%20which%20require%20more%20space (Last accessed: 22.11.2024)
6) Structured vs Unstructured Data: An Overview. URL: https://www.mongodb.com/resources/basics/unstructured-data/structured-vs-unstructured (Last accessed: 22.11.2024)
7) Unstructured Data. URL: https://www.mongodb.com/resources/basics/unstructured-data (Last accessed: 22.11.2024)
8) Parsons N. MongoDB Atlas - Technical Overview & Benefits. URL: https://medium.com/@nparsons08/mongodb-atlas-technical-overview-benefits-9e4cff27a75e (Last accessed: 22.09.2024)
9) European Centre for Medium-Range Weather Forecasts. URL: https://www.ecmwf.int/en/about (Last accessed: 21.09.2024)
10) National Oceanic and Atmospheric Administration. URL: https://www.noaa.gov/about-our-agency (Last accessed: 21.09.2024)
11) Introduction. WeatherAPI. URL: https://www.weatherapi.com/docs/ (Last accessed: 22.09.2024)
12) Lawton G. What is data preprocessing? URL: https://www.techtarget.com/searchdatamanagement/definition/data-preprocessing#:~:text=What%20is%20data%20preprocessing%3F,for%20the%20data%20mining%20process (Last accessed: 1.11.24)
13) A Comprehensive Guide to Data Preprocessing. URL: https://neptune.ai/blog/data-preprocessing-guide (Last accessed: 1.11.24)

14) Feature Engineering. URL: https://www.heavy.ai/technical-glossary/feature-engineering. (Last accessed: 1.11.24)
15) Checa-Garcia R. Statistics for Weather and Climate: Introduction. URL: https://www.researchgate.net/publication/312490308_Statistics_for_Weather_and_Climate_Introduction (Last accessed: 1.11.24)
16) Jayakumar R., Saravanan R. Weather data an alysis data preprocessing. URL: https://www.pnrjournal.com/index.php/home/article/download/4548/5018/5589 (Last accessed: 1.11.24)
17) Juneja A., Das N. Big Data Quality Framework: Pre-Processing Data in Weather Monitoring Application. URL: https://ieeexplore.ieee.org/document/8862267 (Last accessed: 1.11.24)
18) Labeeb K. et al. Pre-Processing Data In Weather Monitoring Application By Using Big Data Quality Framework. URL: https://doi.org/10.1109/WIECON-ECE52138.2020.9397990
19) Thosar S., Bhoyar B., Patil T. Pre-Processing of Data to achieve Quality in Weather Monitoring App. URL: https://www.irjet.net/archives/V7/i5/IRJET-V7I51394.pdf (Last accessed: 1.11.24)
20) Paranjape A., Katta P., Ohlenforst M. Automated Data Preprocessing for Machine Learning Based Analyses. COLLA 2022: The Twelfth Intern. Conf. on Advanced Collaborative Networks, Systems and Applications. IARIA, 2022. URL: https://www.researchgate.net/publication/361026018_Automated_Data_Preprocessing_for_Machine_Learning_Based_Analyses (Last accessed: 1.11.24)
21) pandas.DataFrame.describe. URL: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.describe.html (Last accessed: 1.11.24)

Petrenko T., Zadorozhnyi A. Preprocessing of Meteorological Data for Training an Artificial Intelligence Model

Forecasting weather conditions by classical methods is now successfully supplemented by artificial intelligence methods that allow processing unstructured, semi-structured and structured data. The article considers and analyzes such sources of semi-structured weather data as open APIs of weather resources. An approach to preparing weather data as data for training a graph neural network for forecasting weather data is proposed. The ability to obtain JSON objects with weather parameters (temperature, humidity, pressure, wind speed) by processing HTTP requests using the created Python program and the Requests library ensured the execution of the first stage of data preprocessing. The properties of the selected weather resources and the approach to preparing data for use are described. After receiving weather data from 10 different weather resource APIs, the data are combined and structured. Cleaning, normalization, aggregation and analysis of additional properties of the collected weather data are performed. For example, the calculation of statistical characteristics of the weather indicator "temperature" using Python tools is given.

The article justifies the use of the MongoDB Atlas database to store unstructured meteorological data as objects such as images, audio, video, document files, and other file formats. MongoDB Atlas supports a document-oriented format that increases the flexibility and scalability of data management, which is an advantage for processing large and complex meteorological datasets used in training a graph neural network. The proposed approach combines preprocessing and data storage into a single structure, ensuring the completeness and representativeness of meteorological data. This integration increases the reliability of weather forecasts by using a variety of data. Research confirms the advantages of using MongoDB Atlas and a graph neural network together in capturing spatial and temporal relationships in meteorological data.

Keywords: weather API, weather data preprocessing, statistical analysis, Python libraries, MongoDB Atlas

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

***Задорожний Антон Юрійович**, аспірант кафедри інформаційних технологій, Український державний університет залізничного транспорту, Харків, Україна. E-mail: zadorojniy85@kart.edu.ua, ORCID ID: https://orcid.org/0009-0000-5044-6068*

***Tetyana Petrenko,** PhD, associate professor, department of information technology, Ukrainian State University of Railway Transport, Kharkiv, Ukraine. E-mail: petrenko_tg@kart.edu.ua, ORCID ID http://orcid.org/0000-0001-6305-7918*

***Anton Zadorozhnyi**, post-graduate student, department of information technology, Ukrainian State University of Railway Transport, Kharkiv, Ukraine. E-mail: zadorojniy85@kart.edu.ua, ORCID ID: https://orcid.org/0009-0000-5044-6068*