

ЗАСТОСУВАННЯ АДАПТИВНОЇ ПРЕДИКАТИВНОЇ АНАЛІТИКИ ДЛЯ ПРОГНОЗУВАННЯ ЛОКАЛЬНОЇ МЕТЕОСИТУАЦІЇ

Ю. Корчак, Ю. Фургала, Ю. Панасюк, Д. Рожанківський

*Львівський національний університет імені Івана Франка
вул. ген. Тарнавського, 107, 79017 Львів, Україна*

yuriy.korchak@lnu.edu.ua

У роботі розглянуто основні аспекти розробки базової метеостанції та методи предикативного прогнозування часових рядів. Для побудови апаратного комплексу станції використано сучасні комплектуючі та технології: зібрано прототип станції на основі мікроконтролера ESP32, обрано архітектурне рішення на основі протоколу LoRaWAN та обміну даними з мережею IoT TTN. Аналіз сучасних методів прогнозування часових рядів дозволив зробити висновок про доцільність використання для довгострокових прогнозувань температурних змін математичних моделей, які враховують сезонність. Однак, ці моделі гірше працюють для короткострокового прогнозування (на найближчі 4-5 годин). Обґрунтовано використання та програмно реалізовано на основі типових бібліотек гібридний метод для щопівгодинного прогнозування погоди на наступні два дні.

Ключові слова: часовий ряд, предикативна аналітика, мікроконтролер, протокол обміну даними, сенсор температури, сенсор вологості, сенсор атмосферного тиску, математична модель, стаціонарність часового ряду, точність прогнозування.

Вступ

Сьогодні надзвичайно важливим питанням постає можливість передбачення метеорологічних умов як у короткостроковій, так і в довгостроковій перспективі. Цього потребує практично кожна сфера життя людини. Наприклад, сільське господарство та забезпечення людства продовольством, комунальне господарство та забезпечення комфортних умов проживання людини, військова справа та ефективність проведення військових операцій і т. п. Ця робота присвячена розробці прототипу «розумної» метеостанції – метеостанції, яка зможе збирати не лише погодні дані, а й прогнозувати погоду на найближчий час; пам'ятати, як змінювалися погодні умови в заданому локальному місці в минулому; розпізнавати та попереджати можливі несприятливі для певних ситуацій погодні умови до їх настання; відслідковувати шаблони кліматичних умов, які потрібні для певних конкретних цілей, та повідомляти про майбутнє від них відхилення і т. п.

Ця розробка складається з двох частин: компонування та інтегрування апаратного забезпечення, а також відповідного його програмного відлагодження; програмного забезпечення для аналізу та передбачення погодних умов, тобто, своєрідного «розуму» метеостанції.

Як відомо [1, 2], статистика спостереження за погодою подається у вигляді часових рядів. Тому проведення аналізу відповідного часового ряду є першочерговим завданням

для розуміння його природи та прогнозування майбутніх значень. Обидві ці цілі вимагають, щоб модель ряду була ідентифікована та більш-менш формально описана. Визначення моделі дозволяє з її допомогою інтерпретувати представлені дані і надалі екстраполювати ряд на основі знайденої моделі, тобто передбачити його майбутні значення. Розглянемо декілька найпоширеніших моделей, які сьогодні використовують для проведення аналізу часових рядів.

Математичні моделі опису часових рядів

ARIMA (Autoregressive Integrated Moving Average Model) – модель авторегресійного інтегрування ковзного середнього. Це метод прогнозування часових рядів, запропонований П. Вітлом ще у 1951 році [3] та надалі популяризований Дж. Боксом і Г. М. Дженкінсом у 1970 році [4]. Модель створюється шляхом регресії залежної змінної лише за її величиною запізнення та поточною величиною, а також величиною запізнення члена випадкової похибки в процесі перетворення нестационарних часових рядів у стаціонарні часові ряди. Основна ідея моделі *ARIMA*: обробити послідовність даних, сформовану об'єктом для передбачення з часом, як випадкову послідовність, та використати певну математичну модель, щоб приблизно описати цю послідовність.

ARIMA (p, d, q) загалом складається з декількох процесів:

- ковзного середнього (*MA* – описується параметром q – кількістю елементів ковзного середнього);
- авторегресії (*AR* – описується параметром p – порядком авторегресії (компонентою лінійної регресії в моделі));
- авторегресії ковзного середнього (*ARMA* – описується параметрами q та p);
- авторегресії інтегрального ковзного середнього залежно від того, чи стабільна вихідна послідовність (*ARIMA* – описується параметрами p, d, q , де d – кількість різниць, виконаних для забезпечення стаціонарності часового ряду).

Аналітично цю модель для нестационарного часового ряду $\{X_t\}$ можна описати формулою:

$$\Delta^d X_t = c + \sum_{i=1}^p a_i \Delta^d X_{t-i} + \sum_{j=1}^q b_j \varepsilon_{t-j} + \varepsilon_t, \quad (1)$$

де ε_t – стаціонарний часовий ряд; c, a_i, b_j – параметри моделі; Δ^d – оператор різниці часового ряду порядку d (послідовне взяття d разів різниць першого порядку – спочатку від часового ряду, потім від отриманих різниць першого порядку, далі від другого порядку і т. д.).

Процес реалізації моделі *ARIMA* можна представити чотирма кроками [5]:

- отримати дані часових рядів системи, яка спостерігається;
 - побудувати із цих даних графік та перевірити часовий ряд на стаціонарність. У випадку нестационарного часового ряду взяти різниці між сусідніми значеннями стільки разів, доки часовий ряд не стане стаціонарним (так визначається в цьому випадку параметр d);
 - побудувати корелограми для знаходження оптимальних значень параметрів p та q , використовуючи методи побудови частинних автокореляційних функцій (для оцінки параметра p) та автокореляційних функцій (для оцінки параметра q) [6];
 - підставити отримані величини d, p і q та отримати математичну модель *ARIMA*.
- Далі протестувати її та переконатися, що отримана модель дійсно відповідає

спостережуваним характеристикам даних. Якщо вони не співпадають, повернутись до попереднього кроку.

Модель *ARIMA* призначена для несезонних нестационарних даних. Бокс і Дженкінс [4] узагальнили дану модель для обліку сезонності. Пропонована ними модель відома як модель *Сезонна ARIMA (SARIMA)*. У цій моделі сезонна диференціація відповідного порядку використовується для видалення нестационарності з часового ряду. Сезонна різниця першого порядку – це різниця між спостереженням даного часу і відповідним спостереженням в попередньому році і розраховується як $Y_t = X_t - X_{t-m}$. Для місячних часових рядів, наприклад, $m = 12$. Цю модель, зазвичай, називають моделлю *SARIMA(p,d,q)(P,D,Q)m*, де P – порядок компоненти *AR* сезонної складової; D – порядок диференціювання сезонної складової; Q – порядок компоненти *MA* сезонної складової; m – кількість кроків у сезонному циклі. Важливо, що параметр m впливає на параметри P , D і Q . У випадку моделі прогнозування метеорологічних умов сезонність чітко прослідковується як на подобовому рівні, так і на річному.

SARIMA формується методом додавання додаткових членів до *ARIMA*. Сезонна частина моделі складається з компонентів, які мають походження від несезонних компонентів моделі, але включають зсуви назад сезонного періоду [7].

Аналітично модель *SARIMA* для нестационарного часового ряду $\{X_t\}$ можна описати формулою [7]:

$$\Delta^d \Delta_m^D X_t = c + \sum_{i=1}^p a_i \Delta^d X_{t-i} + \sum_{j=1}^q b_j \varepsilon_{t-j} + \sum_{n=1}^P \varphi_n \Delta_m^D X_{t-mn} + \sum_{l=1}^Q \theta_l \varepsilon_{t-ml} + \varepsilon_t, \quad (2)$$

де четвертий і п'ятий доданки враховують сезонність даних.

Ще одним різновидом моделі *SARIMA* є сезонна модель *ARIMA* з екзогенними (зовнішніми) параметрами, яка отримала назву *SARIMAX*. Ці зовнішні параметри мають бути відомі на момент передбачення. Для прикладу, у випадку метеорологічних прогнозів такими параметрами можуть бути час до/після зеніту сонця, час від сходу сонця і т. п. У цьому випадку в формулі (2) з'являється ще один доданок $\sum_{k=1}^r \beta_k Y_{t-k}$, де Y_t – параметри зовнішнього часового ряду, r – кількість цих параметрів, β_k – параметри екзогенного введення Y_t [8].

Усі розглянуті вище моделі є одновимірними. Розширеннями для багатовимірного випадку є векторна авторегресія *VAR (Vector Autoregression)* та векторна авторегресія з ковзним середнім *VARMA (Vector Autoregression Moving-Average)*. У цьому випадку елементи часового ряду X_t подаються у вигляді векторів, а параметри авторегресії та ковзного середнього – у матричному вигляді [9].

Ще однією моделлю аналізу часових рядів та відповідного прогнозування, котра сьогодні набуває все більшої популярності, є *Facebook Prophet*. Ця бібліотека прогнозування часових рядів була розроблена співробітниками компанії *Facebook* для прогнозування великої кількості бізнес-показників та викладена у вільному доступі для використання в 2017 році.

Фактично, ця бібліотека базується на адитивній регресійній моделі, яка складається з наступних компонентів [10]:

$$x(t) = g(t) + s(t) + h(t) + \varepsilon_t, \quad (3)$$

де:

- *сезонні компоненти* $s(t)$ відповідають за моделювання періодичних змін, пов'язаних з тижневою та річною сезонністю. Тижнева сезонність моделюється за допомогою *dummy variables*. Додаються 6 додаткових ознак, наприклад [понеділок, вівторок, середа, четвер, п'ятниця, субота], котрі приймають значення 0 або 1 залежно від дати. Ознаку *неділя*, яка відповідає сьомому дню тижня, не додають, тому що вона буде лінійно залежати від інших днів тижня і це буде впливати на модель. Річна сезонність моделюється рядами Фур'є;
- *тренд* $g(t)$ – це кусково-лінійна чи логістична функція. З лінійною функцією все зрозуміло. Логістична функція у вигляді

$$g(t) = \frac{C}{1 + \exp(-k(t - b))}$$

дозволяє моделювати зростання з насиченням, коли у випадку збільшення показника (b) знижується темп його зростання (k). Типовий приклад – це ріст аудиторії додатку чи сайту. Крім того, бібліотека вміє за історичними даними автоматично обирати оптимальні точки зміни тренду. Хоча їх також можна задати вручну (наприклад, якщо відомі дати релізів нової функціональності, котрі сильно вплинуть на ключові показники);

- компонента $h(t)$ відповідає за задані користувачем аномальні дні, в тому числі і нерегулярні, такі як, наприклад, чорна п'ятниця;
- помилка ε містить інформацію, яка не врахована моделлю.

Апаратне та програмне забезпечення прототипу метеостанції

Компонування прототипу метеостанції (рис.1, *a*) здійснювалося на основі мікроконтролера ESP32. Це серія мікроконтролерів типу «система на кристалі», що мають інтегровані контролери *WiFi* і *Bluetooth*, низьке енергоспоживання і невисоку ціну (рис. 1, *б*). Збір та інтерпретація метеоданих відбувається завдяки під'єднанню до мікроконтролера набору сенсорів. Для вимірювання температури та атмосферного тиску використовувався сенсор BMP180 (рис. 1, *в*), а для вимірювання вологості – сенсор HTU2X (рис. 1, *г*). Така система збирає погодні дані та за допомогою протоколу *LoRaWAN* передає їх на ще один мікроконтролер ESP32, який уже за допомогою *WiFi* скеровує дані на сервер (для тестування і відлагодження використовується серверний сервіс *TTN – The Things Network*).

LoRaWAN – протокол мережевої радіочастотної комунікації. Має достатній радіус дії (до 20 км), а також споживає незначну кількість електроенергії (значно менше, ніж *WiFi*) [11], що в контексті вирішуваної задачі є надзвичайно важливим.

TTN – серверний сервіс, за допомогою якого можна отримувати дані від іншого контролера по *WiFi*, накопичувати їх, зберігати та скачувати [12]. Це одне із програмних рішень Інтернету речей та використовувався для відлагодження проекту, доки він не отримає власної серверної частини.

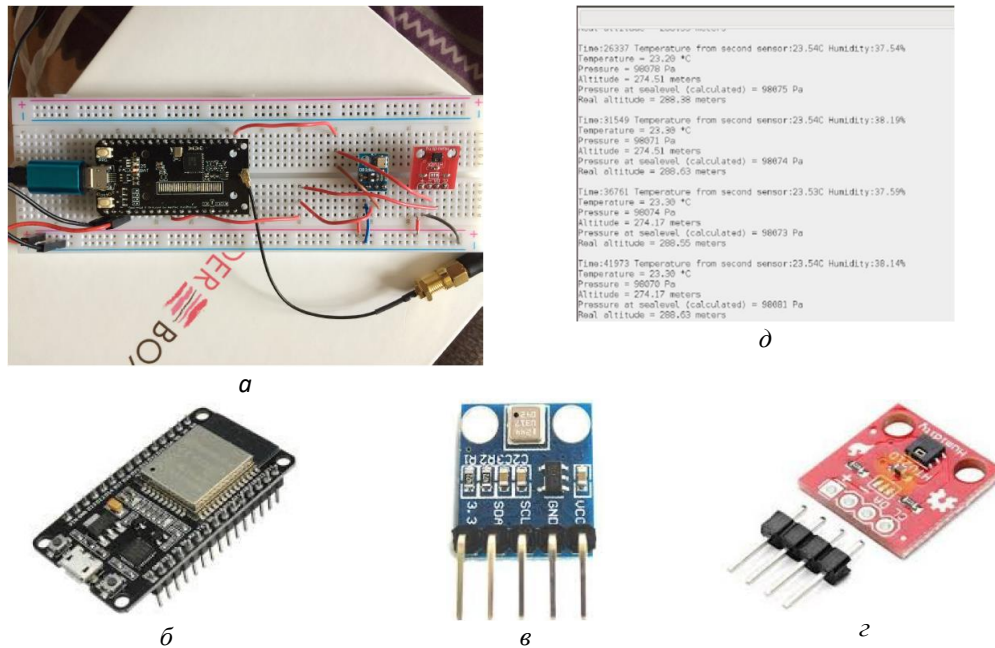


Рис. 1. Прототип метеостанції (а) та основні складові її частини: мікроконтролер ESP32 (б), сенсор температури та атмосферного тиску BMP180 (в), сенсор вологості HTU2X (г). Фрагмент роботи створеної метеостанції (д).

Для програмування основних функцій апаратного забезпечення метеостанції використовувалася мова програмування C++ та середовище для розробки Arduino IDE.

Перелік бібліотек для програмування апаратної частини комплексу:

- `#include <lmic.h>` - бібліотека для роботи з протоколом зв'язку *LoRaWAN*;
- `#include <hal/hal.h>` - бібліотека для роботи з форматом пакетів, що пересилаються по мережевому протоколу;
- `#include <SPI.h>` - бібліотека для роботи з зовнішніми компонентами. За її допомогою контролюється зв'язок і робота з окремими пристроями, під'єднаними до мікроконтролера (сенсори в нашому випадку);
- `#include <Wire.h>` - бібліотека для роботи з одношиновими зовнішніми компонентами. Використовувалася для контролю сенсора температури з одношиновою комунікацією, який згодом був виключений з проекту, але знадобиться згодом для інших сенсорів подібної будови;
- `#include <Adafruit_BMP85.h>` - бібліотека спеціально для роботи з сенсорами BMP. Забезпечує їхню коректну роботу та зручні засоби отримання даних від них;
- `#include "SparkFunHTU21D.h"` - бібліотека спеціально для роботи з сенсорами HTU. Забезпечує їхню коректну роботу та зручні засоби отримання даних від них.

Після здійснення усіх етапів програмування зібраного комплексу та відповідного запуску показано, що система коректно збирає та персилає дані (рис. 1, д).

Приклади практичного застосування різних моделей для предикативної аналітики температурних змін

Для проведення порівняльного аналізу застосування різних моделей передбачення отримано метеодані (температура, атмосферний тиск, вологість) з веб-сайту www.weather.com у форматі *json* у м. Львові за 2 роки (період 2019-2021 років) (рис. 2) з кроком щопівгодини. Предикативна аналітика проводилася для змін температури, а вологість та атмосферний тиск розглядалися як екзогенні (зовнішні) параметри. Для передбачення зміни температури обрано місяць жовтень 2020 року і прогнозування здійснювалося на короткостроковий (19^{00} – 23^{30} год. 22.10.2020 р. – 10 прогнозованих значень) та довгостроковий (10^{00} год. 20.10.2020 р. – 23^{30} год. – 76 прогнозованих значень) періоди. Тестування моделей реалізувалося протягом 2-6 попередніх до зазначених періодів місяців.

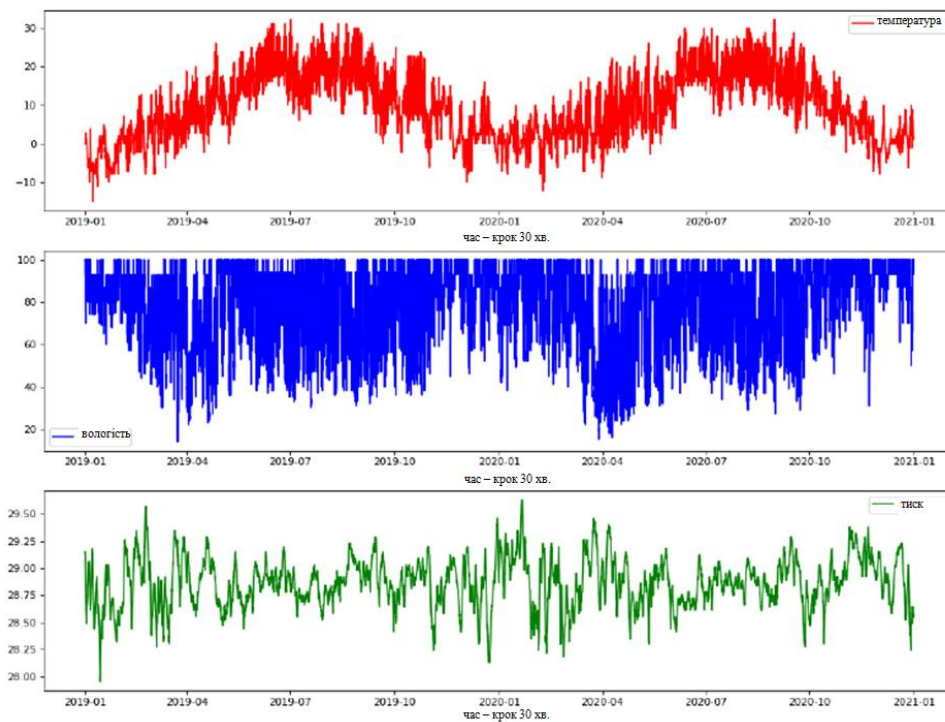


Рис. 2. Статистика зміни метеоданих (температури, вологості, атмосферного тиску) у м. Львові за 2 роки (2019-2021), отримана зі сайту www.weather.com.

Розглянемо реалізацію передбачення змін температури на прикладі використання моделей *ARIMA*, *SARIMA* та *Prophet*.

Як уже було зазначено вище, після побудови часового ряду у випадку застосування моделі *ARIMA* потрібно здійснити перевірку цього часового ряду на його стаціонарність. З цією метою використаний прикладний пакет *statsmodels* [13], в якому ця перевірка реалізується на основі методів Дікі-Фулера [14] та Квятковскі-Філіпса-Шмідта-Шина [15].

Якщо проаналізувати температурний часовий ряд у жовтні 2020 року, то для більшості днів стаціонарним є ряд перших різниць, хоча трапляються дні, коли стаціонарним є ряд других різниць, наприклад, 20 жовтня (рис. 3). Отже, у випадку застосування моделі *ARIMA* слід використовувати величини параметра d , рівні 1 або 2.

```

період: 2020-10-20 00:00:00 - 2020-10-20 23:30:00
ADF Statistic: -1.922070
p-value: 0.321751
Critical Values: 10% -2.602344
Critical Values: 5% -2.928299
Critical Values: 1% -3.584829
є одиничні корені, ряд НЕ Є СТАЦІОНАРНИМ.

Тест на стаціонарність перших різниць:
ADF Statistic: -2.718147
p-value: 0.070950
Critical Values: 10% -2.602344
Critical Values: 5% -2.928299
Critical Values: 1% -3.584829
є одиничні корені, ряд перших різниць НЕ Є СТАЦІОНАРНИМ.

Тест на стаціонарність перших різниць:
ADF Statistic: -7.170501
p-value: 0.000000
Critical Values: 10% -2.603185
Critical Values: 5% -2.929886
Critical Values: 1% -3.588573
немає одиничних коренів, ряд 2-их різниць Є СТАЦІОНАРНИМ.

період: 2020-10-01 00:00:00 - 2020-10-01 23:30:00
ADF Statistic: -2.444003
p-value: 0.129700
Critical Values: 1% -3.577848
Critical Values: 10% -2.600774
Critical Values: 5% -2.925338
є одиничні корені, ряд НЕ Є СТАЦІОНАРНИМ.

Тест на стаціонарність перших різниць:
ADF Statistic: -7.453829
p-value: 0.000000
Critical Values: 1% -3.581258
Critical Values: 10% -2.601541
Critical Values: 5% -2.926785
немає одиничних коренів, ряд перших різниць Є СТАЦІОНАРНИМ.

```

Рис. 3. Приклади перевірки на стаціонарність часових температурних рядів 1 та 20 жовтня 2020 року.

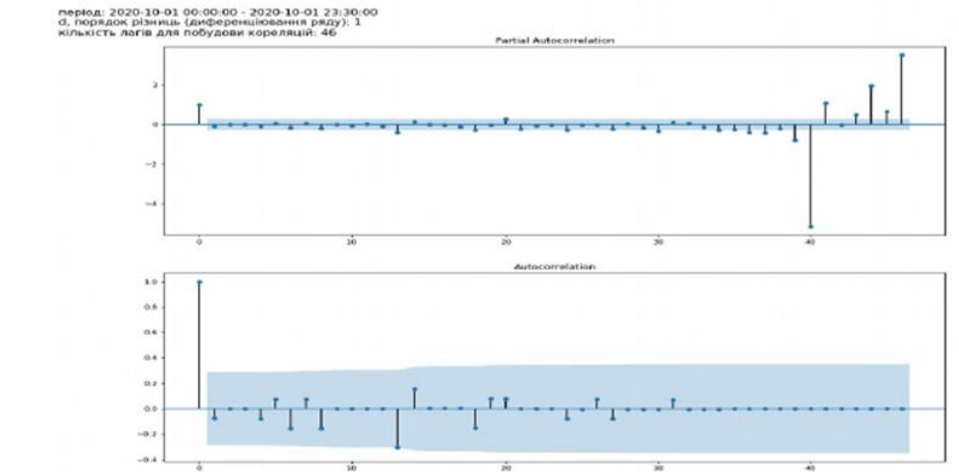
З метою оцінки параметрів p і q побудовано корелограми частинних автокореляційних функцій (для оцінки параметра p) та автокореляційних функцій (для оцінки параметра q) відповідно для перших (рис. 4, *a*) та других (рис. 4, *б*) різниць. Для цього були використані методи `plot_pacf()` і `plot_acf()` у пакеті `statsmodels`. Як видно з побудованих корелограм, параметри p і q можуть приймати значення 2 або 3.

Після експериментування як з наборами термінів для навчання моделі, так і з різними параметрами виявлено, що найкраще описує задані температурні дані модель *ARIMA* (3, 2, 3), яка будується на наборі даних у 1,5-2 місяці. Для оцінки точності подальшого прогнозування використовувався пакет `NumPy` [16], за допомогою якого визначалися параметри середньої абсолютної похибки у відсотках, кореляції між актуальним рядом і передбаченим, помилки міні-макс. У результаті показано (рис. 5), що модель добре підходить до прогнозування на найближчі 4-5 годин (приблизно 10 щопівгодинних прогнозованих значень), де абсолютна похибка у відсотках становила 8 %.

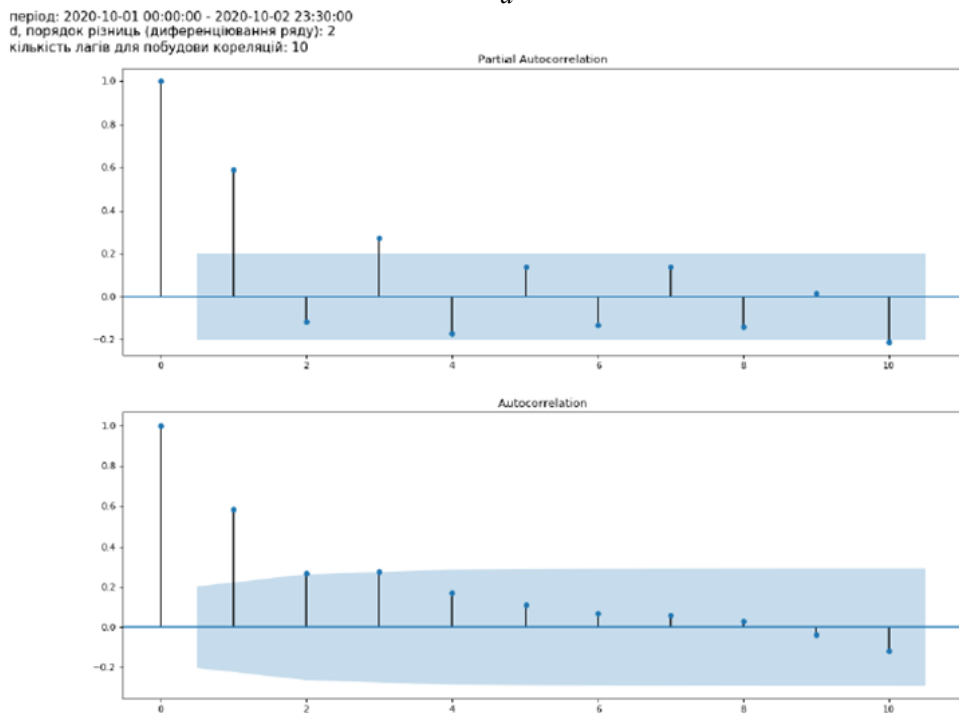
Використання пакету `pmdarima` [17] дозволяє шляхом автоматичного перебирання моделей пришвидшити процес пошуку оптимальних параметрів і термінів навчання моделі *ARIMA*. Зокрема, похибку у ~10 % отримано для моделі *ARIMA* (0, 2, 1) для знову ж таки 10 найближчих щопівгодинних прогнозованих значень, але у випадку тренування моделі на даних максимум двох діб перед прогнозованим періодом.

Якщо ж використовувати модель *SARIMA*, то в короткостроковій перспективі потрібно враховувати подобову сезонність. За допомогою пакету `pmdarima` виявлено, що для величин 0 або 1 параметрів p і q та тренувальним набором даних в 10 днів найкращі

результати для прогнозування на найближчі 2 доби дає модель $SARIMA(1, 0, 1)(1, 1, 0)48$ (рис. б).



a



б

Рис. 4. Корелограми частинних автокореляційних функцій та автокореляційних функцій для $d = 1$ (а) і $d = 2$ (б).

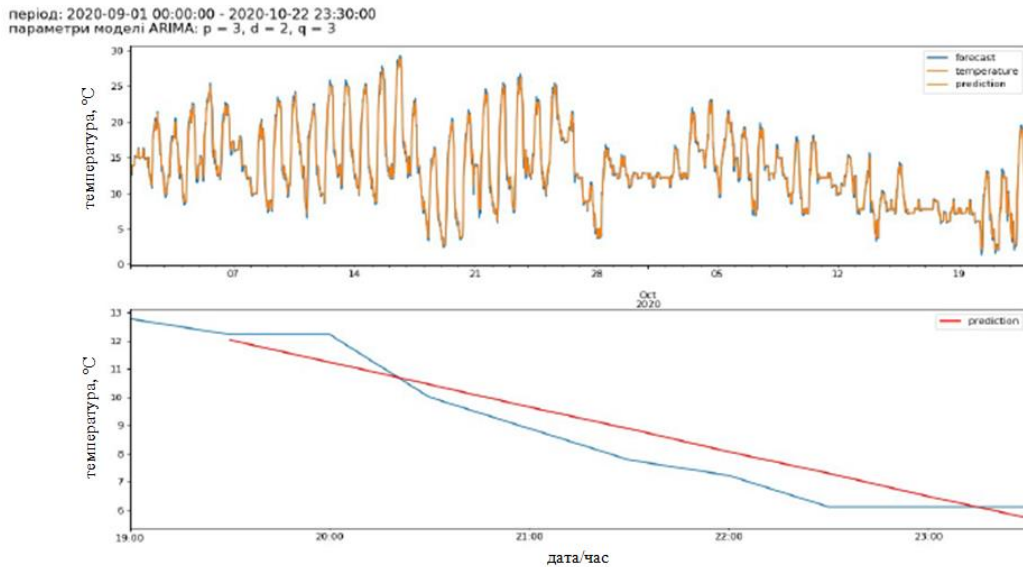


Рис. 5. Застосування моделі $ARIMA(3, 2, 3)$ для прогнозування температурних змін. На рисунку знизу синім кольором позначено реальні значення температури, а червоним – прогнозовані значення.

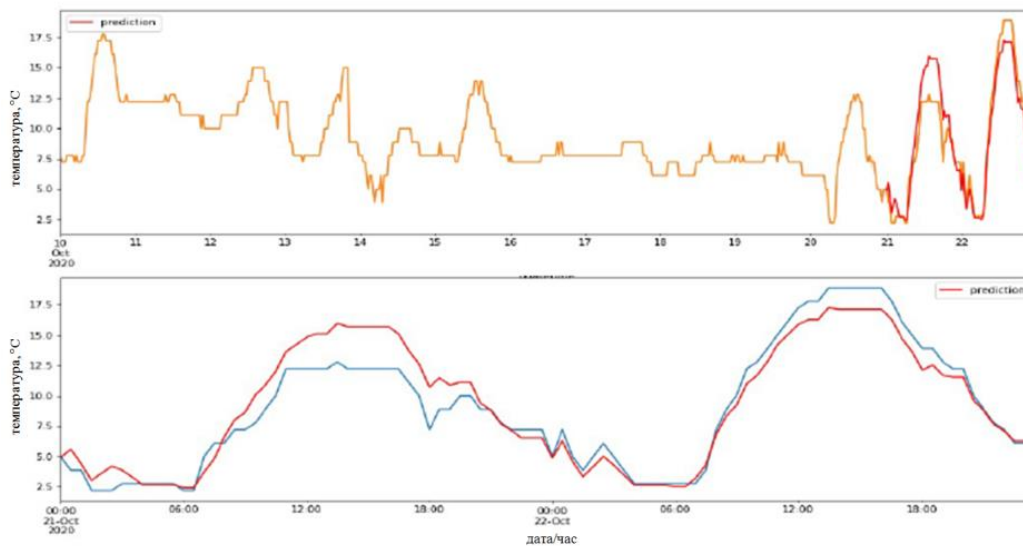


Рис. 6. Застосування моделі $SARIMA(1, 0, 1)(1, 1, 0)_{48}$ для прогнозування зміни температури. На рисунку знизу синім кольором позначено реальні значення температури, а червоним – прогнозовані значення.

У роботі також проаналізовано застосування моделей *Prophet*, *VAR* та *VARIMA* як у чистому вигляді, так і з використанням екзогенних параметрів (змінних-регресорів). Зокрема, на рис. 7 представлено застосування моделі *Prophet* для прогнозування температури на найближчі два дні (21-22.10.2020 р.) без використання екзогенних параметрів (атмосферного тиску, вологості) (рис.7, а) та з їхнім використанням (рис. 7, б). Як видно з рис. 7, у першому випадку отримуємо точніше прогнозування. Загалом, це цілком очевидний факт, адже ці змінні-регресори також зазнають прогнозування на цей же період з деякою точністю, що зумовлює внесок їхньої похибки в загальний результат.

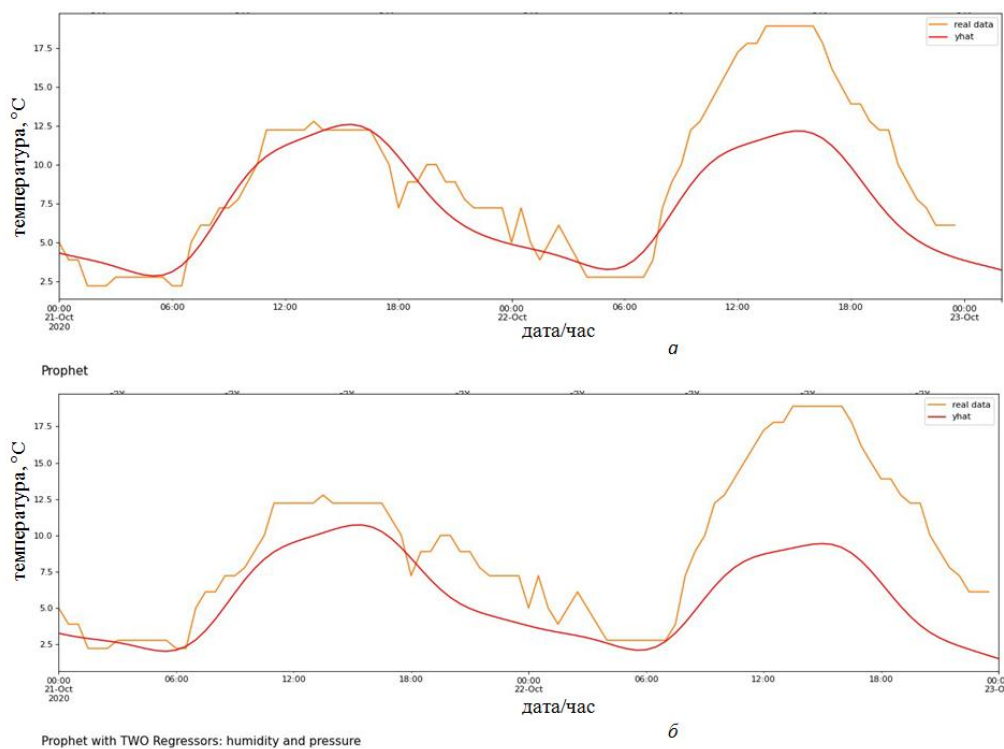


Рис. 7. Застосування моделі *Prophet* для прогнозування зміни температури на найближчі два дні без використання змінних-регресорів (а) та з їхнім використанням (б). На рисунку жовтим кольором позначено реальні значення температури, а червоним – прогнозовані значення.

Порівняння точності прогнозування розглянутих моделей наведено в таблиці 1. Аналізуючи дані цієї таблиці та враховуючи загальноприйняті підходи до оцінки точності прогнозування часових рядів [18], можна зробити висновок, що у випадку короткострокового (4-5 годин) прогнозування температурних змін придатними для використання є моделі *ARIMA*, *VAR*, *VARIMA* у різних модифікаціях, точність прогнозування яких є висока ($MAPE < 10\%$) або добра ($MAPE 10-20\%$), тоді як у випадку довгострокових прогнозів (2 дні) – моделі *SARIMA* і *Prophet* у різних модифікаціях, точність прогнозування яких є добра ($MAPE 10-20\%$) або задовільна

(MAPE 20-50 %). У табл. 1 не наведено даних щодо моделей *ARIMA*, *VAR*, *VARIMA* у випадку довгострокового прогнозування, бо середня абсолютна похибка в цьому випадку перевищувала 50 %, що свідчить про незадовільну точність прогнозування.

Таблиця 1

Порівняння точності прогнозування різних моделей часових рядів

Модель	Середня абсолютна похибка у відсотках (<i>MAPE</i>)	Кореляція між актуальним рядом і передбаченням (<i>corr</i>)	Помилка міні-макс (<i>minmax</i>)
Наступні 10 прогнозованих значень (19 ⁰⁰ -23 ³⁰ год. 22.10.2020)			
<i>ARIMA</i> (3, 2, 3)	0,0830599618	0,9688275747	0,0756311631
<i>ARIMA</i> (0, 2, 1)	0,0968588050	0,9673459780	0,0852050615
<i>SARIMA</i> (1, 0, 1)(1, 1, 1)48	0,3716584216	0,9768388501	0,2405413762
<i>Prophet</i>	0,3894806280	0,9674890495	0,3894806280
<i>VAR</i>	0,1882022642	0,9665590125	0,1477591690
<i>VARIMA</i>	0,1890764589	0,9587095505	0,1495525444
<i>Prophet</i> з екзог. парам.	0,4862581354	0,9677321978	0,4862581354
<i>VARIMA</i> з екзог. парам.	0,0950810750	0,9781901916	0,0825817710
Наступні 76 прогнозованих значень (10 ⁰⁰ год. 21.10.2020 – 23 ³⁰ год. 22.10.2020)			
<i>SARIMA</i> (1, 0, 1)(1, 1, 0)48	0,1396890463	0,9538132720	0,1150054516
<i>SARIMA</i> (1, 0, 1)(1, 1, 1)48	0,4121356784	0,9224870971	0,2875276130
<i>Prophet</i>	0,2766029857	0,8418690647	0,2555695657
<i>Prophet</i> з екзог. парам.	0,3408670472	0,8311488760	0,3388046558

Враховуючи ці результати, зроблена спроба побудови температурного прогнозування на найближчі два дні (23-24.10.2020 р.) із врахуванням двох підмоделей: на перших 10 прогнозованих значеннях із використанням моделі *ARIMA* (1, 2, 2), а далі – з використанням моделі *Prophet* (рис. 8). Тренувальні набори для цих підмоделей теж обрано різні. Модель *ARIMA* тренували на наборі даних двох днів (48 значень), а *Prophet* – на двомісячному наборі даних.

На вставці верхнього рисунку (рис. 8) зображено приклад застосування моделі *Prophet* і для перших 10 прогнозованих значень. Як бачимо, підтверджується попередній висновок про краще короткострокове прогнозування з використанням моделі *ARIMA*.

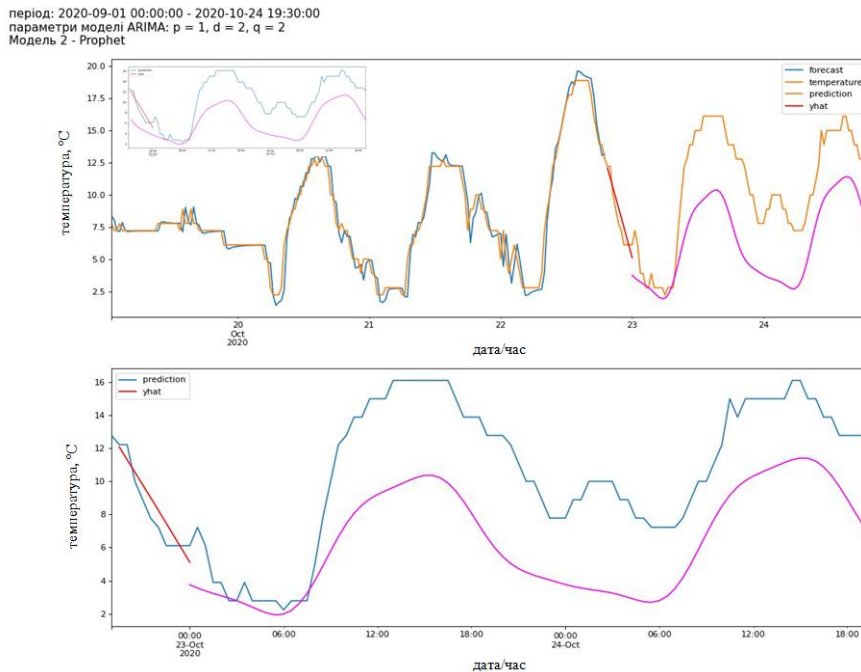


Рис. 8. Застосування гібридної моделі, яка складається з підмоделей *ARIMA* (1, 2, 2) та *Prophet*, для температурного прогнозування. Червоним кольором позначено прогнозовані значення, отримані з використанням моделі *ARIMA* (1, 2, 2), а пурпуровим – з використанням моделі *Prophet*.

Загалом, у цьому випадку отримується задовільна точність і такий підхід теж можна цілком застосовувати у випадку прогнозування зміни метеоданих.

Висновки

Використання сучасних технологій, які містять як апаратні засоби (мікроконтролери, сенсори), так і архітектурні та програмні рішення, дало змогу у рамках цієї роботи створити прототип метеостанції, яка цілком відповідає сучасним критеріям самостійного апаратного комплексу.

Проведення порівняльного аналізу ефективності використання існуючих на сьогодні математичних моделей (алгоритмів, бібліотек) для прогнозування майбутніх даних на основі реальних на прикладі температурних змін дозволило зробити такі висновки:

- прогнозування більше як на 3-5 наступних годин можна робити лише на основі моделей, що враховують сезонність, зокрема *SARIMA* та *Prophet*, як гірше працюють на короткострокових прогнозах;
- використання деяких прогнозованих параметрів (вологість, атмосферний тиск) як екзогенних призводить до погіршення точності прогнозування. Очевидно, прогнозовані параметри як екзогенні вносять помилку їхнього прогнозування в кінцевий результат. Отже, в цьому випадку вартує враховувати як екзогенні ті параметри, які завжди відомі наперед, наприклад, висоту сонця над горизонтом;

- моделі без врахування сезонності гірше працюють на інтервалах, де зустрічаються локальні екстремуми температур.

Як наслідок, запропоновано гібридну модель для прогнозування температурних змін, яка для короткотермінової (4-5 годин) перспективи використовує модель *ARIMA*, а для довготермінової (2 дні) – модель *Prophet*.

Список використаних джерел

- [1] *Manly Bryan F.J.* Statistics for Environmental Science and Management. 2nd Edition / Bryan F.J. Manly. – New York: Chapman and Hall/CRC, 2008. – 292 p.
- [2] *Вельміскін Д. І.* Автоматичні системи метеорологічних вимірювань. Конспект лекцій / Д. І. Вельміскін. – Одеса: ТЕС, 2010. – 82 с.
- [3] *Whittle P.* Hypothesis Testing in Time Series Analysis/ P. Whittle. – Uppsala: Almqvist and Wiksells Boktryckeri AB, 1951. – 120 p.
- [4] *Vox G.P.E.* Time Series Analysis: Forecasting and Control / G.P.E. Vox, G.M. Jenkins. – San Francisco: Holden-Day, 1970. – 575 p.
- [5] *Дзендзелюк О.* Побудова *ARIMA* моделей часових рядів для прогнозування метеоданих на мові програмування R / О. Дзендзелюк, Л. Костів, В. Рабик // Електроніка та інформаційні технології. – 2013. – Вип. 3. – С. 211-219.
- [6] Моделі сталого розвитку {Електронний ресурс}. – Режим доступу: <http://dspace.tnpu.edu.ua/bitstream/123456789/26067/1/masliu.pdf>
- [7] *Hyndman R. J.* Forecasting: Principles and Practice / R.J. Hyndman, G. Athanasopoulos. – Melbourne: OTexts.org/fpp, 2013. – 242 p.
- [8] *Artley B.* Time Series Forecasting with *ARIMA*, *SARIMA* and *SARIMAX* [Electronic resource]. – Mode of access: <https://towardsdatascience.com/time-series-forecasting-with-arima-sarima-and-sarimax-ee61099e78f6>
- [9] *Wei William W.S.* Time Series Analysis: Univariate and Multivariate Methods / William W.S. Wei. – New York: Addison-Wesley Publishing Company, Inc, 2006. – 614 p.
- [10] *Krieger M.* Time Series Analysis with Facebook Prophet: How it works and How to use it [Electronic resource]. – Mode of access: <https://towardsdatascience.com/time-series-analysis-with-facebook-prophet-how-it-works-and-how-to-use-it-f15ecf2c0e3a>
- [11] LoRaWAN. Learn [Electronic resource]. – Mode of access: <https://www.thethingsnetwork.org/docs/lorawan/> ; Single Channel LoRaWAN Gateway [Electronic resource]. – Mode of access: <https://github.com/things4u/ESP-1ch-Gateway>
- [12] The Things Network. Learn [Electronic resource]. – Mode of access: <https://www.thethingsnetwork.org/docs/>
- [13] statsmodels.tsa.arima.model.ARIMA [Electronic resource]. – Mode of access: <https://www.statsmodels.org/devel/generated/statsmodels.tsa.arima.model.ARIMA.html>
- [14] *Verma Yu.* Complete Guide to Dickey-Fuller Test in Time-Series Analysis [Electronic resource]. – Mode of access: <https://analyticsindiamag.com/complete-guide-to-dickey-fuller-test-in-time-series-analysis/>
- [15] KPSS Test: Definition and Interpretation [Electronic resource]. – Mode of access: <https://www.statisticshowto.com/kpss-test/>
- [16] NumPy [Electronic resource]. – Mode of access: <https://numpy.org>
- [17] pmdarima [Electronic resource]. – Mode of access: <https://pypi.org/project/pmdarima/>
- [18] What is a good MAPE score? [Electronic resource]. – Mode of access: <https://stephenallwright.com/good-mape-score/>

APPLICATION OF ADAPTIVE PREDICATIVE ANALYTICS FOR FORECASTING THE LOCAL WEATHER SITUATION

Yu. Korchak, Yu. Furgala, Yu. Panasiuk, D. Rozhankivskyi

*Ivan Franko National University of Lviv,
107 Tarnavsky St., UA-79017 Lviv, Ukraine
yuriy.korchak@lnu.edu.ua*

The paper considers the main aspects of the development of a basic weather station and methods of predicative forecasting of time series in case of meteorological changes.

To build the hardware complex of the station, modern components and technologies were used: a prototype of the station was assembled based on an ESP32 microcontroller using temperature and atmospheric pressure sensor BMP180 and humidity sensor HTU2X, an architectural solution based on the *LoRaWAN* protocol and data exchange with the IoT *TTN* network was chosen. Testing the operation of the created complex confirmed the correctness of data collection and their forwarding.

The method of forecasting local temperature changes is considered in detail using the *ARIMA* and *SARIMA* models as an example. The analysis of modern methods of forecasting time series made it possible to draw a conclusion about the feasibility of using mathematical models that take seasonality into account for long-term forecasts of temperature changes. However, these models perform worse for short-term forecasting (for the next 4-5 hours). In addition, it is shown that the use of some predicted parameters (humidity, atmospheric pressure) as exogenous leads to a deterioration of forecasting accuracy. In this case, it is worth using as exogenous well-known parameters in advance.

A hybrid method for half-hourly weather forecasting for the next two days is substantiated and programmatically implemented on the basis of standard libraries, which uses the method *ARIMA* (1, 2, 2) for the short-term (4-5 hours) forecast period, and then the *Prophet* model. The proposed solution provides quite satisfactory prediction accuracy.

Key words: time series, predicative analytics, microcontroller, data exchange protocol, temperature sensor, humidity sensor, atmospheric pressure sensor, mathematical model, time series stationarity, forecasting accuracy.

*Стаття надійшла до редакції 31.08.2022.
Прийнята до друку 08.09.2022.*