

Івано-Франківський національний технічний університет нафти і газу

Факультет автоматизації та енергетики

Кафедра інформаційно-вимірювальних технологій та енергетичного менеджменту

Данів Павло Олегович

(прізвище, ім'я, по батькові)

УДК 620.9

(індекс)

БАКАЛАВРСЬКА РОБОТА

Розроблення системи машинного навчання для опрацювання вимірних даних та прогнозування споживання енергії будівлею

(назва роботи)

Інженерія відновлюваної енергетики

(назва освітньої програми)

152 «Метрологія та інформаційно вимірювальна техніка»

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня _____ П.О.Данів

(підпис, ініціали та прізвище здобувача)

Науковий керівник _____ Сорока Назарій-Андрій Юрійович

доктор філософії, доцент

(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту

Завідувач кафедри

_____ В.С.Цих

(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ - 2025 рік

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Факультет автоматизації та енергетики

Кафедра інформаційно-вимірювальних технологій та енергетичного менеджменту

Освітній рівень *бакалавр*

Спеціальність *152 «Метрологія та інформаційно-вимірювальна техніка»*

Освітня програма *Інженерія відновлюваної енергетики*

ЗАТВЕРДЖУЮ

Завідувач кафедри _____

« ____ » _____ 20__ року

З А В Д А Н Н Я НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Даніву Павлу Олеговичу

(прізвище, ім'я, по батькові)

1. Тема роботи *Розроблення системи машинного навчання для опрацювання вимірних даних та прогнозування споживання енергії будівлею*

керівник роботи *Сорока Назарій-Андрій Юрійович доктор філософії, доцент*

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом ІФНТУНГ від *“29” квітня 2025 року № 268/7*

2. Строк подання студентом роботи *“09” червня 2025 року*

3. Вихідні дані до роботи

База даних для проведення дослідження. Літературні джерела за тематикою _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз глобального попиту на електроенергію. Способи підвищення енергоефективності

2. Прогнозування даних методами машинного навчання.

3. Аналіз і передбачення використаної та згенеровані електроенергії з використанням методів машинного навчання.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
	<i>Нормоконтролер</i>	<i>доцент Яворський А. В.</i>	
	<i>Перевірка на плагіат</i>	<i>доцент Миндюк В. Д.</i>	

7. Дата видачі завдання “29” квітня 2025 року**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Видача завдання (відповідно до наказу)	29.04.2025	
2	Розроблення 1 розділу: «Аналіз предметної області»	30.04.2025 – 09.05.2025	
3	Розроблення 2 розділу: «Прогнозування даних методами машинного навчання»	10.05.2025 – 21.05.2025	
4	Розроблення 3 розділу: «Аналіз і передбачення використаної та згенерованої електроенергії з використанням методів машинного навчання»	22.05.2025 – 30.05.2025	
5	Оформлення роботи	31.05.2025 – 07.06.2025	

Студент

(підпис)

Данів П.О.

(прізвище та ініціали)

Керівник роботи

(підпис)

Сорока Н.-А.Ю.

(прізвище та ініціали)

РЕФЕРАТ

Данів П.О. «Розроблення системи машинного навчання для опрацювання вимірянних даних та прогнозування споживання енергії будівлею». Дипломна робота освітнього рівня – бакалавр, на правах рукопису. Спеціальність – 152 «Метрологія та інформаційно-вимірвальна техніка». – Івано-Франківськ, 2025.

Структура бакалаврської роботи включає вступ, три основні розділи, висновки та додатки. Загальний обсяг становить 71 сторінка, включаючи 3 рисунки і 5 таблиць. Бібліографія охоплює 39 офіційних та наукових джерел.

Перший розділ присвячено аналізу предметної області та огляду існуючих підходів до управління енергоспоживанням будівель. Другий розділ розкриває теоретичні основи застосування методів машинного навчання для прогнозування енергетичних даних. Третій розділ містить практичну реалізацію системи прогнозування, аналіз результатів та рекомендації щодо впровадження. У висновках узагальнено основні результати дослідження та окреслено перспективи подальших досліджень.

Робота виконана з використанням сучасних програмних засобів для аналізу даних та машинного навчання, включаючи мову програмування Python та спеціалізовані бібліотеки для обробки часових рядів і реалізації алгоритмів машинного навчання. Особлива увага приділена практичним аспектам впровадження розроблених методів у реальних умовах експлуатації будівель, що забезпечує високу прикладну цінність дослідження.

Дослідження проводиться в контексті світових тенденцій цифровізації енергетичного сектору та переходу до сталих енергетичних систем. Розроблена система має потенціал для масштабування та адаптації до різних типів будівель, що робить її перспективним інструментом для підвищення енергоефективності будівельного сектору в цілому.

Ключові слова: прогнозування, енергоспоживання, машинне навчання, алгоритми навчання, будівлі, база даних, часовий ряд.

ABSTRACT

Daniv P.O. “Development of a machine learning system for processing measured data and predicting energy consumption by a building”. Thesis of the educational level – bachelor, in the form of a manuscript. Specialty – 152 “Metrology and information and measuring technology”. – Ivano-Frankivsk, 2025.

The structure of the bachelor's thesis includes an introduction, three main chapters, conclusions and appendices. The total volume is 71 pages, including 3 figures and 5 tables. The bibliography includes 39 official and scientific sources.

The first section is devoted to the analysis of the subject area and an overview of existing approaches to building energy management. The second section reveals the theoretical foundations of the application of machine learning methods to predict energy data. The third section contains the practical implementation of the forecasting system, analysis of the results, and recommendations for implementation. The conclusions summarize the main results of the study and outline the prospects for further research.

The work was performed using modern software tools for data analysis and machine learning, including the Python programming language and specialized libraries for processing time series and implementing machine learning algorithms. Special attention is paid to the practical aspects of implementing the developed methods in real building operation conditions, which ensures high applied value of the research.

The research is conducted in the context of global trends in the digitalization of the energy sector and the transition to sustainable energy systems. The developed system has the potential to be scaled and adapted to different types of buildings, making it a promising tool for improving the energy efficiency of the building sector as a whole.

Keywords: forecasting, energy consumption, machine learning, learning algorithms, buildings, database, time series.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1.АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	12
1.1 Аналіз глобального попиту на електроенергію	12
1.2 Основні фактор впливу на енергоспоживання	15
1.3 Способи підвищення енергоефективності.....	19
1.4 Необхідність моделей прогнозування для підвищення раціонального споживання енергії.....	23
РОЗДІЛ 2. ПРОГНОЗУВАННЯ ДАНИХ МЕТОДАМИ МАШИННОГО НАВЧАННЯ.....	29
2.1 Класифікація завдань машинного навчання.....	29
2.2 Методи оцінки точності передбачень	30
2.3 Моделі машинного навчання.....	31
2.4 Розробка коду для реалізації прогнозування даних енергоспоживання..	33
РОЗДІЛ 3. АНАЛІЗ І ПЕРЕДБАЧЕННЯ ВИКОРИСТАНОЇ ТА ЗГЕНЕРОВАНОЇ ЕЛЕКТРОЕНЕРГІЇ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ	38
3.1 Підбір бази даних для машинного навчання	38
3.2 Підготовка, аналіз та відбір даних для прогнозування	40
3.3 Порівняння та вибір найкращого методу машинного навчання для поставленого завдання.....	43
3.4 Аналіз результатів та оцінка похибок.....	47
ВИСНОВКИ	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	53
ДОДАТКИ	

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД – база даних

API- (Application Programming Interface) - інтерфейс прикладного програмування

BEMS (Building Energy Management System) — система управління енергоспоживанням будівлі

BMS (Building Management System) — система управління будівлею

COP (Coefficient of Performance) — коефіцієнт ефективності (співвідношення отриманої корисної енергії до витраченої)

ERV (Energy Recovery Ventilation) — вентиляція з рекуперацією енергії

HRV (Heat Recovery Ventilation) — вентиляція з рекуперацією тепла

HVAC (Heating, Ventilation, and Air Conditioning) — системи опалення, вентиляції та кондиціонування повітря

HVDC (High Voltage Direct Current) — високовольтний постійний струм

IoT (Internet of Things) — інтернет речей

KNN - K-nearest neighbors (K-найближчих сусідів)

LED (Light-Emitting Diode) — світловипромінюючий діод

LoRaWAN (Long Range Wide Area Network) — мережа дальнього радіусу дії з низьким енергоспоживанням
MAE – mean absolute error (середня абсолютна помилка)

MAPE - mean absolute percentage error (середня абсолютна процентна похибка)

ML – Machine Learning (машинне навчання)

MSE – mean square error (середньоквадратична помилка)

PV (Photovoltaic) — фотовольтаїчний (сонячні панелі)

PIR (Passive Infrared) — пасивний інфрачервоний (датчики руху)

RFID (Radio Frequency Identification) — радіочастотна ідентифікація

SCADA (Supervisory Control and Data Acquisition) — диспетчерське управління та збір даних

VRF (Variable Refrigerant Flow) — система змінного холодоагенту

ZigBee — протокол бездротового зв'язку для автоматизації будівель

ВСТУП

Сучасний світ стоїть перед безпрецедентними викликами в області енергетики, які зумовлені зростаючим попитом на електроенергію, необхідністю зниження викидів парникових газів та підвищенням ефективності використання енергетичних ресурсів. Будівлі, які споживають близько 40% загального обсягу електроенергії у світі, є критично важливою сферою для впровадження інноваційних технологій енергоменеджменту. У цьому контексті розроблення систем машинного навчання для прогнозування споживання енергії будівлями набуває особливої актуальності як інструмент оптимізації енергетичних процесів та забезпечення сталого розвитку.

Актуальність теми дослідження визначається кількома ключовими факторами. По-перше, глобальне споживання електроенергії продовжує зростати з середньорічними темпами 3-4% [1, 2], що створює додатковий тиск на енергетичні системи та вимагає розроблення ефективних методів управління попитом. Будівлі споживають близько 40% загального енергоспоживання в США [3, 4], а 85% будівель ЄС були побудовані до 2000 року, з яких 75% мають низьку енергоефективність [5, 6]. Ці статистичні дані свідчать про величезний потенціал для впровадження систем енергоменеджменту на основі машинного навчання.

По-друге, розвиток штучного інтелекту та машинного навчання відкриває нові можливості для аналізу великих обсягів енергетичних даних та створення точних прогнозних моделей. Ринок систем управління енергоспоживанням оцінювався в 44,19 млрд доларів у 2024 році і прогнозується досягти 84,34 млрд доларів до 2029 року [7], що демонструє високу затребуваність таких технологій. Сучасні алгоритми машинного навчання, включаючи методи глибокого навчання, регресійні моделі та ансамблеві методи, дозволяють досягати високої точності прогнозування енергоспоживання з урахуванням множини факторів впливу.

По-третє, особливої актуальності проблема набуває в Україні, де приблизно 62% електроенергії генерується на атомних електростанціях у 2023 році [8], а енергетична система зазнала значних пошкоджень. З 2024 року в

Україні запрацював спеціальний фонд для реалізації проектів енергоефективності, що підкреслює державну підтримку впровадження інноваційних енергетичних технологій. В умовах обмежених енергетичних ресурсів розроблення ефективних систем прогнозування споживання енергії стає критично важливим завданням.

Мета дослідження полягає у розробленні ефективної системи машинного навчання для опрацювання вимірних даних та прогнозування споживання енергії будівлею з високою точністю та надійністю. Досягнення цієї мети передбачає вирішення комплексу взаємопов'язаних завдань, що включають аналіз існуючих методів прогнозування енергоспоживання, розроблення оптимальної архітектури системи машинного навчання, створення ефективних алгоритмів обробки та аналізу енергетичних даних, а також забезпечення високої точності та надійності прогнозування в умовах змінних факторів навколишнього середовища та експлуатаційних характеристик будівлі.

Для досягнення поставленої мети необхідно розв'язати наступні завдання:

1. Провести комплексний аналіз предметної області, включаючи дослідження глобальних тенденцій енергоспоживання, основних факторів впливу на споживання енергії будівлями та існуючих підходів до підвищення енергоефективності.
2. Здійснити детальний огляд методів машинного навчання, придатних для прогнозування часових рядів енергоспоживання, включаючи класифікацію алгоритмів, методи оцінки точності та особливості їх застосування для енергетичних даних.
3. Сформувати та підготувати базу даних енергоспоживання для тренування моделей машинного навчання, включаючи очищення даних, виявлення та усунення аномалій, а також створення релевантних ознак.
4. Реалізувати та порівняти різні алгоритми машинного навчання, зокрема методи лінійної регресії з ваговими коефіцієнтами (LSTM), алгоритм k-найближчих сусідів (KNN) та метод екстримального підсилення (XGBoost) для прогнозування енергоспоживання.

5. Провести валідацію розроблених моделей на реальних даних енергоспоживання будівлі ліцею, оцінити їх точність та надійність за допомогою статистичних метрик.
6. Розробити рекомендації щодо практичного впровадження системи машинного навчання для моніторингу та прогнозування енергоспоживання в умовах реальної експлуатації будівель.

Об'єкт дослідження – процес енергоспоживання будівель та його прогнозування за допомогою методів машинного навчання. Об'єкт дослідження охоплює комплексний аналіз використання електричної енергії в будівлі, що включає всі основні групи споживачів: системи освітлення, системи опалення, вентиляції та кондиціонування повітря (HVAC), електричне обладнання загального призначення, спеціалізовані технічні системи та інші споживачі електроенергії. Дослідження враховує динаміку енергоспоживання в часі, вплив зовнішніх факторів та експлуатаційних характеристик будівлі на загальний енергетичний баланс.

Предмет дослідження – алгоритми та методи машинного навчання для аналізу та прогнозування часових рядів енергоспоживання будівель на основі вимірних даних. Предмет дослідження включає розроблення та оптимізацію математичних моделей для обробки енергетичних даних, створення ефективних обчислювальних алгоритмів для прогнозування енергетичних потреб, розроблення програмних засобів для автоматизації процесів аналізу та прогнозування, а також методи оцінки точності та надійності прогнозних моделей в умовах мінливих експлуатаційних параметрів будівлі.

Методи дослідження базуються на комплексному підході, що поєднує теоретичні та практичні аспекти машинного навчання. Основними методами є: статистичний аналіз енергетичних даних для виявлення закономірностей та трендів; методи попередньої обробки даних для забезпечення їх якості та придатності для навчання моделей; алгоритми контрольованого навчання, включаючи лінійну регресію з методом найменших квадратів (LSTM), алгоритм k-найближчих сусідів (KNN) та екстремальне градієнтне підсилення (XGBoost); методи валідації моделей, включаючи крос-валідацію та оцінку на тестових

наборах даних; статистичні методи оцінки якості прогнозів, включаючи середньоквадратичну похибку (MSE), середню абсолютну похибку (MAE) та коефіцієнт детермінації (R^2).

Практична цінність роботи полягає у можливості застосування розроблених моделей у реальних системах управління енергоспоживанням будівель. Глобальний ринок розумних будівель оцінювався в 108 млрд доларів у 2023 році з прогнозованим зростанням на 28,5% щорічно до 2030 року [9], що створює значні можливості для комерціалізації розроблених рішень. Результати дослідження можуть бути впроваджені в системи управління будівлями (Building Management Systems, BMS) для автоматичної оптимізації енергоспоживання. Розроблені методи дозволяють знизити витрати на електроенергію до 15-25% за рахунок точного прогнозування потреб та оптимізації режимів роботи енергетичного обладнання. Системи автоматизації будівель, інтегровані з HVAC та управлінням освітленням, можуть заощадити близько 10-20% загального споживання електроенергії будівлею [10], а BEMS може допомогти організаціям заощадити до 30% витрат на енергію [11].

Прогнозування обсягів енергоспоживання є критично важливим для України в сучасних умовах, зокрема через військову агресію росії. Цей фактор додає до існуючих викликів, таких як економічна нестабільність, енергетична залежність, кліматичні зміни та необхідність впровадження ефективних енергетичних політик. У сучасному глобалізованому світі, де енергетичні ресурси відіграють ключову роль у розвитку економіки, забезпечення стабільного та прогнозованого енергоспоживання стає основою енергетичної безпеки країни.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз глобального попиту на електроенергію

Сучасний світ характеризується постійно зростаючою залежністю від електричної енергії, що робить аналіз глобального попиту на електроенергію критично важливим питанням для розуміння енергетичного майбутнього планети. Електроенергія є основою функціонування промисловості, комунальних послуг, транспорту та побутового життя мільярдів людей, що обумовлює необхідність детального вивчення тенденцій її споживання.

За даними Міжнародного енергетичного агентства (IEA) [12], глобальне споживання електроенергії демонструє стійку тенденцію до зростання. У 2024 році споживання електроенергії зросло на 4,3% порівняно з попереднім роком, що значно перевищує середньорічний показник зростання у 2,5% у 2023 році. Загальний обсяг світового споживання електроенергії у 2024 році досяг приблизно 30 439 ТВт·год, що на 968 ТВт·год більше порівняно з 2023 роком.

Регіональний розподіл споживання електроенергії характеризується значною нерівномірністю. Китай залишається найбільшим споживачем електроенергії у світі, на частку якого припадає 31% глобального споживання та 55% споживання в Азійському регіоні. Сполучені Штати Америки займають друге місце за абсолютними показниками споживання, проте їх частка у глобальному споживанні становить приблизно 11%. Європейський Союз, незважаючи на зусилля з енергоефективності, також демонструє значні обсяги споживання електроенергії.[13]

Особливо важливо відзначити динаміку розвитку країн, що розвиваються. Африканський континент, незважаючи на низькі абсолютні показники споживання на душу населення (які становлять лише половину від показників Індії та на 70% нижчі за показники Південно-Східної Азії), демонструє найвищі темпи зростання попиту на електроенергію. Прогнозується, що середньорічне зростання попиту на електроенергію в Африці протягом 2024-2026 років

становитиме 4%, що вдвічі перевищує середні темпи зростання у 2017-2023 роках. [13]

Ключовими драйверами зростання глобального попиту на електроенергію є кілька фундаментальних факторів. По-перше, це процеси урбанізації та індустріалізації в країнах, що розвиваються. Зростання міського населення супроводжується підвищенням стандартів життя та збільшенням використання електричних приладів побутового призначення. По-друге, це цифровізація економіки та розвиток центрів обробки даних, які споживають значні обсяги електроенергії для охолодження та функціонування серверного обладнання.

Кліматичні фактори також відіграють важливу роль у формуванні попиту на електроенергію. Зростання температур внаслідок глобального потепління призводить до збільшення потреби в кондиціонуванні повітря, особливо в регіонах з жарким кліматом. У 2024 році саме зростання попиту на охолодження стало одним з основних факторів збільшення споживання електроенергії у Сполучених Штатах.

Електрифікація транспорту є ще одним важливим трендом, що впливає на глобальний попит на електроенергію. Хоча електромобілі поки що становлять невелику частку від загального автопарку, їх частка швидко зростає, особливо в розвинених країнах. Це створює додатковий попит на електроенергію та вимагає розвитку відповідної інфраструктури зарядження.[14]

Промислова електрифікація також сприяє зростанню попиту. Багато промислових процесів, які раніше базувалися на викопному паливі, поступово переходять на електричну енергію з метою зменшення викидів парникових газів. Це особливо актуально для металургійної, хімічної та цементної промисловості.

Структура генерації електроенергії зазнає кардинальних змін. Частка відновлюваних джерел енергії у глобальному енергетичному міксі зросла до 30% у 2023 році, що на 10 відсоткових пунктів більше порівняно з 2010 роком [15]. Сонячна енергетика демонструє найвищі темпи зростання, подвоюючи свої обсяги генерації за останні три роки. Вітрова енергетика також

показує стійке зростання, і разом з сонячною енергетикою у 2024 році вперше перевищила обсяги генерації гідроелектростанцій [16].

Водночас частка ядерної енергетики у глобальному енергетичному міксі знизилася до 9%, що є найнижчим показником за останні 45 років [17]. Це пов'язано з виведенням з експлуатації застарілих ядерних реакторів у деяких країнах та повільним введенням в експлуатацію нових потужностей.

Прогнозування попиту на електроенергію показує, що у період 2025-2027 років очікується продовження тенденції зростання з середньорічними темпами близько 3,9% [18]. Важливо відзначити, що весь додатковий попит планується покривати за рахунок низьковуглецевих технологій, включаючи відновлювані джерела енергії та ядерну енергетику.

Енергетична безпека стає все більш актуальною проблемою у контексті зростаючого попиту на електроенергію. Країни розробляють стратегії диверсифікації джерел енергопостачання та збільшення частки власного виробництва електроенергії. Це особливо важливо для країн, які значною мірою залежать від імпорту енергоносіїв.

Технологічні інновації у сфері енергоефективності частково стримують зростання попиту на електроенергію. Впровадження LED-освітлення, енергоефективних електроприладів та систем розумного управління енергоспоживанням дозволяє знижувати питоме енергоспоживання на одиницю економічної активності. Проте ці ефекти поки що не можуть повністю компенсувати зростання попиту на енергоресурси.

Економічні фактори також суттєво впливають на динаміку споживання електроенергії. Зростання валового внутрішнього продукту традиційно корелює із збільшенням енергоспоживання, хоча в розвинених країнах ця залежність поступово слабшає завдяки структурним змінам в економіці та підвищенню енергоефективності.

Геополітичні чинники створюють додаткові виклики для глобального енергетичного ринку. Конфлікти та торговельні суперечки можуть призводити до порушень у постачанні енергоносіїв та коливань цін на електроенергію, що

впливає на патерни споживання у різних регіонах світу. Глобальне споживання електроенергії за регіонами у 2023-2024 роках відображено в таб.1.1.

Таблиця 1.1 - Глобальне споживання електроенергії за регіонами у 2023-2024 роках [18]

<i>Регіон</i>	<i>Споживання 2023 (TВт·год)</i>	<i>Споживання 2024 (TВт·год)</i>	<i>Зростання (%)</i>	<i>Частка у світовому споживанні (%)</i>
Китай	9,100	9,450	3.8	31.0
США	4,200	4,350	3.6	14.3
Індія	1,800	1,920	6.7	6.3
Японія	1,050	1,070	1.9	3.5
Росія	1,100	1,120	1.8	3.7
Німеччина	580	590	1.7	1.9
Канада	640	660	3.1	2.2
Бразилія	520	540	3.8	1.8
Південна Корея	580	595	2.6	2.0
Іран	350	370	5.7	1.2
Африка (загалом)	800	850	6.3	2.8
Європейський Союз	2,800	2,850	1.8	9.4
Інші країни	6,051	6,234	3.0	20.5
Світ загалом	29,471	30,439	3.3	100.0

Аналіз представлених даних свідчить про збереження провідної ролі Азійського регіону у глобальному споживанні електроенергії, при цьому найвищі темпи зростання демонструють країни з економікою, що розвивається. Це створює як можливості для розвитку енергетичної інфраструктури, так і виклики щодо забезпечення сталого та екологічно чистого енергопостачання [19].

1.2 Основні фактори впливу на енергоспоживання

Енергоспоживання будівель формується під впливом складної системи взаємопов'язаних факторів, що охоплюють кліматичні умови, архітектурно-

планувальні рішення, технічні характеристики інженерних систем, режими експлуатації та поведінкові особливості користувачів. Розуміння механізмів впливу цих факторів є фундаментальною основою для розробки ефективних систем прогнозування та управління енергоспоживанням. Кожен з факторів характеризується специфічними закономірностями варіації в часі та просторі, що вимагає застосування відповідних математичних методів для їх моделювання та урахування у прогнозних алгоритмах.

Кліматичні фактори відіграють домінуючу роль у формуванні енергетичного балансу будівель, визначаючи потреби в опаленні, охолодженні та вентиляції. Зовнішня температура повітря є найбільш значущим кліматичним параметром, що безпосередньо впливає на теплові втрати через огорожувальні конструкції та потужність систем клімат-контролю. Залежність енергоспоживання від температури зазвичай має нелінійний характер з вираженими точками балансу, де потреби в опаленні або охолодженні досягають мінімуму. Градусо-дні опалення та охолодження є стандартизованими метриками для кількісної оцінки кліматичного навантаження, що широко використовуються в енергетичних розрахунках та нормативних документах. Сонячна радіація створює додаткові теплові надходження через світлопрозорі конструкції, що може як зменшувати потреби в опаленні взимку, так і збільшувати навантаження на системи кондиціонування влітку. Інтенсивність сонячного випромінювання залежить від географічної широти, пори року, часу доби, хмарності та орієнтації поверхонь будівлі.

Вологість повітря впливає на ефективність систем кондиціонування та комфортність внутрішнього середовища, особливо в регіонах з вологим кліматом. Високі показники відносної вологості підвищують навантаження на системи осушення повітря, тоді як низька вологість може вимагати додаткового зволоження. Швидкість та напрямок вітру визначають інтенсивність інфільтрації повітря через нещільності будівельної оболонки та ефективність природної вентиляції. Атмосферний тиск, хоча і менш значущий, може впливати на роботу вентиляційних систем та процеси повітрообміну.

Архітектурно-планувальні характеристики будівлі створюють фізичну основу для формування енергетичних потоків та визначають базовий рівень енергоспоживання. Геометричні параметри будівлі, включаючи загальну площу, об'єм, висоту та компактність форми, безпосередньо впливають на теплові втрати та енергетичні потреби. Коефіцієнт форми, що характеризує відношення площі огорожувальних конструкцій до об'єму будівлі, є ключовим показником енергетичної ефективності архітектурного рішення. Будівлі з більш компактною формою зазвичай мають нижчі питомі теплові втрати завдяки меншій площі зовнішніх поверхонь відносно внутрішнього об'єму [20].

Орієнтація будівлі та розміщення вікон визначають режими надходження сонячної енергії та потреби в штучному освітленні. Південні фасади отримують максимальну кількість сонячної енергії в зимовий період, що може суттєво зменшити потреби в опаленні при правильному архітектурному плануванні. Східні та західні орієнтації характеризуються інтенсивними надходженнями сонячної енергії в ранкові та вечірні години відповідно, що може створювати додаткові навантаження на системи кондиціонування влітку. Північні фасади отримують мінімальну кількість прямої сонячної радіації, але забезпечують стабільне розсіяне освітлення.

Теплотехнічні характеристики огорожувальних конструкцій визначають інтенсивність теплообміну між внутрішнім та зовнішнім середовищем. Опір теплопередачі стін, покрівлі, вікон та дверей безпосередньо впливає на теплові втрати та потужність систем опалення. Теплова інерція будівельних конструкцій визначає швидкість реакції внутрішньої температури на зміни зовнішніх умов та ефективність використання пасивної сонячної енергії. Повітропроникність будівельної оболонки характеризує інтенсивність неконтрольованого повітрообміну через нещільності конструкцій, що може становити значну частину загальних теплових втрат.

Інженерні системи будівлі, включаючи опалення, вентиляцію, кондиціонування, освітлення та гаряче водопостачання, є основними споживачами енергії та визначають операційні характеристики енергетичної системи. Ефективність обладнання, виражена через коефіцієнти корисної дії

котлів, теплових насосів, кондиціонерів та світильників, безпосередньо впливає на питоме енергоспоживання. Потужність встановленого обладнання визначає максимальне споживання енергії та пікові навантаження на енергетичні мережі [21]. Режимми роботи систем, включаючи графіки включення-виключення, температурні налаштування та швидкості вентиляторів, суттєво впливають на загальне енергоспоживання.

Системи автоматизації та управління визначають ефективність координації роботи різних підсистем будівлі та можливості оптимізації енергоспоживання. Програмовані контролери дозволяють реалізувати складні алгоритми управління з урахуванням множини параметрів та зворотних зв'язків. Датчики присутності, освітленості, температури та якості повітря забезпечують інформаційну основу для адаптивного управління інженерними системами відповідно до реальних потреб користувачів.

Функціональні характеристики будівлі визначають базові вимоги до енергетичних систем та режимів їх експлуатації. Призначення будівлі впливає на типові графіки використання, вимоги до внутрішнього мікроклімату та інтенсивність експлуатаційних навантажень. Офісні будівлі характеризуються високими потребами в штучному освітленні та інтенсивним використанням комп'ютерної техніки в денний час. Житлові будівлі мають більш рівномірний добовий профіль споживання з піками в ранкові та вечірні години. Торгівельні центри характеризуються високими навантаженнями на системи кондиціонування через значні внутрішні тепловиділення та інтенсивні повітрообміни.

Щільність заселення та інтенсивність використання приміщень визначають рівень внутрішніх тепловиділень від людей, обладнання та освітлення. Ці тепловиділення можуть становити значну частину теплового балансу будівлі, особливо в холодний період року, коли вони зменшують потреби в опаленні. В теплий період додаткові тепловиділення збільшують навантаження на системи кондиціонування повітря.

Поведінкові фактори користувачів створюють значну варіабельність в енергоспоживанні будівель навіть при ідентичних технічних характеристиках.

Температурні переваги користувачів можуть варіювати в широких межах, що впливає на налаштування термостатів та режими роботи систем клімат-контролю. Звички щодо використання штучного освітлення, відкриття вікон, експлуатації побутової техніки та електронного обладнання суттєво впливають на загальне енергоспоживання. Рівень енергетичної свідомості та мотивації до енергозбереження визначає готовність користувачів адаптувати свою поведінку для зменшення споживання енергії.

Сезонні та часові фактори створюють циклічні патерни в енергоспоживанні, що повинні враховуватися при прогнозуванні та плануванні енергетичних потреб. Сезонні зміни проявляються через варіації кліматичних умов, тривалості світлового дня та режимів використання будівлі. Добові цикли відображають типові графіки активності користувачів та роботи інженерних систем. Тижневі патерни пов'язані з різницею між робочими днями та вихідними [21]. Календарні ефекти включають вплив святкових днів, канікул та сезонних змін в інтенсивності використання будівель.

1.3 Способи підвищення енергоефективності

Підвищення енергоефективності будівель представляє собою комплексний підхід, що охоплює широкий спектр технологічних, архітектурних та управлінських рішень, спрямованих на мінімізацію споживання енергії при збереженні або покращенні рівня комфорту та функціональності. Сучасні стратегії енергоефективності базуються на принципах системного аналізу енергетичних потоків, інтеграції інноваційних технологій та оптимізації операційних процесів. Ефективна реалізація заходів з енергоефективності вимагає глибокого розуміння взаємодії між різними компонентами будівлі як єдиної енергетичної системи, де оптимізація одного елементу може значно впливати на продуктивність інших підсистем. Основні способи підвищення енергоефективності будівель наведено в табл. 1.2.

Термоізоляція будівельних конструкцій є фундаментальним напрямом підвищення енергоефективності, що дозволяє суттєво зменшити втрати тепла через огорожувальні конструкції.

Таблиця 1.2 - Основні способи підвищення енергоефективності будівель [22].

<i>Напря́м</i>	<i>Технології/Рішення</i>	<i>Ефективність</i>	<i>Переваги</i>	<i>Особливості впровадження</i>
Термоізоляція	Екструдований пінополістирол, мінеральна вата, аерогелеві композити	До 50% зменшення теплових втрат	Усунення теплових мостів, суцільний теплоізоляційний контур	Зовнішнє утеплення фасадів, багатошарові конструкції
Енергоефективні вікна	Багатокамерні профілі, низькоемісійні склопакети, інертне заповнення	Коефіцієнт теплопередачі на рівні стін	Оптимізація балансу сонячної енергії і втрат	Автоматичні системи затінення
Системи HVAC	Теплові насоси, конденсаційні котли, рекуперація тепла	Ефективність до 95%, COP 2-3	Утилізація прихованої теплоти	Інтелектуальні системи управління
Розумне управління	Програмовані термостати з ML, системи присутності	20-30% економії енергії	Адаптація до режимів використання	Централізовані системи енергоменеджменту
LED освітлення	Світлодіодні технології, системи диммування	80% менше споживання	Вища якість світла, довговічність	Автоматичне регулювання за освітленістю
Відновлювані джерела	Сонячні панелі, колектори, геотермальні системи	20%+ ефективність фотовольтаїки, 70% колектори	Автономність від мереж	Інтеграція з системами акумулювання
Акумулювання енергії	Літій-іонні батареї, теплові акумулятори	Висока щільність енергії	Оптимізація пікових навантажень	Системи управління попитом
Моніторинг та аудит	Розумні лічильники, термографія	Real-time контроль	Виявлення неефективних режимів	Системи зворотного зв'язку

Сучасні теплоізоляційні матеріали, включаючи екструдований пінополістирол, мінеральну вату високої щільності та інноваційні аерогелеві композити, забезпечують високі теплотехнічні характеристики при мінімальній товщині конструкції. Технологія зовнішнього утеплення фасадів дозволяє

усунути теплові мости та створити суцільний теплоізоляційний контур будівлі. Утеплення покрівлі та підвальних приміщень забезпечує комплексний підхід до мінімізації теплових втрат через всі елементи будівельної оболонки. Використання багатошарових конструкцій з повітряними прошарками та відбивними екранами дозволяє досягти надвисоких показників термічного опору за оптимальних економічних витрат.

Енергоефективні вікна та двері відіграють критичну роль у загальній тепловій продуктивності будівлі, оскільки традиційно є найслабшими елементами будівельної оболонки з точки зору теплових втрат. Сучасні віконні системи з багатокамерними профілями, низькоемісійними склопакетами та інертним газовим заповненням забезпечують коефіцієнти теплопередачі на рівні стінових конструкцій. Технології селективних покриттів дозволяють оптимізувати баланс між пропусканням сонячної енергії та тепловими втратами залежно від кліматичних умов та орієнтації приміщень. Автоматичні системи затінення та вентиляційні кватирки забезпечують динамічне регулювання теплового режиму приміщень без втрати природного освітлення.

Системи опалення, вентиляції та кондиціонування повітря представляють найбільший потенціал для підвищення енергоефективності через впровадження високотехнологічного обладнання та інтелектуальних систем управління. Теплові насоси з високим коефіцієнтом перетворення дозволяють отримувати у два-три рази більше теплової енергії порівняно з витраченою електричною енергією за рахунок використання низькопотенціального тепла навколишнього середовища. Конденсаційні котли забезпечують ефективність використання палива до 95% через утилізацію прихованої теплоти пароутворення продуктів згоряння. Системи рекуперації тепла у вентиляційних установках дозволяють повертати до 85% теплової енергії витяжного повітря для підігріву припливного, що суттєво зменшує навантаження на основні опалювальні пристрої.

Інтелектуальні системи управління будівлею інтегрують всі енергетичні підсистеми у єдину мережу автоматизованого контролю та оптимізації. Програмовані термостати з алгоритмами машинного навчання адаптуються до

режимів використання приміщень та автоматично коригують температурні налаштування для мінімізації енергоспоживання без зниження комфорту. Системи присутності та освітленості забезпечують автоматичне управління штучним освітленням та обладнанням залежно від реальної потреби. Централізовані системи енергетичного менеджменту дозволяють моніторити споживання енергії в реальному часі, виявляти неефективні режими роботи та автоматично оптимізувати налаштування обладнання.

Системи управління освітленням автоматичним регулюванням залежно від рівня природного освітлення дозволяють додатково зменшити енергоспоживання на 30-50%. Архітектурні рішення з максимальним використанням природного освітлення, включаючи світлові шахти, атріуми та світлопроводи, зменшують потребу у штучному освітленні протягом денного часу [22].

Відновлювані джерела енергії забезпечують можливість часткової або повної автономності будівлі від зовнішніх енергетичних мереж. Сонячні фотовольтаїчні системи дозволяють генерувати електричну енергію безпосередньо на даху або фасаді будівлі, при цьому сучасні панелі забезпечують ефективність перетворення сонячної енергії понад 20%. Сонячні колектори для гарячого водопостачання та підтримки опалення забезпечують ефективність використання сонячної енергії до 70%. Геотермальні системи використовують стабільну температуру ґрунту для ефективного опалення та охолодження будівель з мінімальними витратами електроенергії.

Системи акумулювання енергії дозволяють оптимізувати використання відновлюваних джерел енергії та зменшити пікові навантаження на електричну мережу. Літій-іонні батареї забезпечують ефективне зберігання електричної енергії з високою щільністю енергії та тривалим терміном експлуатації. Теплові акумулятори дозволяють зберігати надлишкову теплову енергію для використання у періоди підвищеного попиту [23]. Системи управління попитом інтегрують акумулювання енергії з прогнозуванням навантажень для оптимізації енергетичних потоків.

Енергетичний аудит та моніторинг є основою для ідентифікації можливостей підвищення енергоефективності та контролю ефективності впроваджених заходів. Сучасні системи моніторингу з розумними лічильниками дозволяють отримувати детальну інформацію про споживання енергії окремими підсистемами будівлі в реальному часі. Термографічне обстеження виявляє теплові втрати та дефекти теплоізоляції. Аналіз якості внутрішнього повітря забезпечує оптимізацію роботи вентиляційних систем.

Поведінкові аспекти енергоефективності включають навчання користувачів раціональному використанню енергетичних ресурсів та створення мотиваційних механізмів для енергозбереження. Системи зворотного зв'язку інформують користувачів про поточне споживання енергії та рекомендації щодо його оптимізації. Програми стимулювання енергоефективної поведінки можуть включати економічні заохочення та соціальні механізми визнання досягнень у сфері енергозбереження.

1.4. Необхідність моделей прогнозування для підвищення раціонального споживання енергії

Сучасні виклики енергетичної безпеки та екологічної стійкості вимагають кардинального перегляду підходів до управління енергетичними ресурсами на всіх рівнях економічної діяльності. Раціональне споживання енергії стає не лише економічною необхідністю, але й стратегічним імперативом для забезпечення сталого розвитку суспільства. У цьому контексті моделі прогнозування енергоспоживання виступають ключовим інструментом для досягнення оптимального балансу між задоволенням енергетичних потреб та мінімізацією екологічного впливу. Розробка та впровадження точних систем прогнозування дозволяє перейти від реактивного управління енергоспоживанням до проактивного планування, що забезпечує суттєве підвищення ефективності використання енергетичних ресурсів [24].

Традиційні підходи до управління енергоспоживанням базуються на історичних трендах та емпіричних правилах, що виявляються недостатньо гнучкими для адаптації до мінливих умов експлуатації будівель та зовнішніх

факторів. Відсутність точного прогнозування призводить до неоптимального використання енергетичного обладнання, надлишкового споживання у періоди низького попиту та недостатньої підготовки до пікових навантажень. Моделі прогнозування дозволяють подолати ці обмеження шляхом аналізу великих масивів даних та виявлення складних нелінійних залежностей між факторами впливу та споживанням енергії. Інтеграція моделей прогнозування у системи енергетичного менеджменту забезпечує можливість превентивного налаштування обладнання, оптимізації графіків роботи та зменшення енергетичних витрат. Переваги впровадження моделей прогнозування енергоспоживання наведено в табл.1.3.

Таблиця 1.3 - Переваги впровадження моделей прогнозування енергоспоживання [25].

<i>Аспект</i>	<i>Традиційний підхід</i>	<i>Прогнозні моделі</i>	<i>Переваги прогнозування</i>	<i>Результати</i>
Економічні переваги	Реактивне управління, фіксовані тарифи	Проактивне планування, диференційовані тарифи	Мінімізація витрат на електроенергію	Зменшення щомісячних платежів
Технічне обслуговування	Планове ТО за графіком	ТО на основі прогнозованих навантажень	Запобігання аварійним ситуаціям	Підвищення надійності та довговічності систем, зменшення витрат на ремонт та простій
Екологічний вплив	Статичне споживання	Оптимізація енергосистем	Зменшення викидів парникових газів	Інтеграція з відновлюваними джерелами
Технічні можливості	Емпіричні правила	Машинне навчання, багатofакторний аналіз	Виявлення нелінійних залежностей	Автоматична адаптація моделей
Інтеграція зі Smart Grid	Односторонні потоки	Двостороння комунікація	Координоване управління ресурсами	Динамічні тарифи, управління попитом
Регуляторна відповідність	Базова звітність	Автоматизована звітність	Відповідність директивам ЄС	Підвищення рейтингу будівель
Соціальні переваги	Пасивне споживання	Свідоме енергоспоживання	Підвищення енергетичної безпеки	Розвиток культури енергозбереження

Економічні переваги впровадження прогнозних моделей проявляються через декілька механізмів оптимізації витрат. Точне прогнозування дозволяє мінімізувати витрати на електроенергію шляхом використання диференційованих тарифів та переміщення навантажень у періоди нижчих цін. Перспективні технології для прогнозування енергоспоживання наведено в табл. 1.4. Для великих споживачів це може означати суттєве зменшення щомісячних платежів за електроенергію через оптимізацію пікового попиту та використання часових тарифів. Прогнозування також забезпечує можливість планування технічного обслуговування енергетичного обладнання на основі прогнозованих навантажень, що дозволяє уникнути аварійних ситуацій та дорогавартісних ремонтів. Крім того, точні прогнози споживання енергії дозволяють оптимізувати розміри резервних потужностей та зменшити капітальні витрати на енергетичну інфраструктуру [25].

Таблиця 1.4 - Перспективні технології для прогнозування енергоспоживання [26].

<i>Технологія</i>	<i>Опис</i>	<i>Застосування</i>	<i>Переваги</i>	<i>Перспективи розвитку</i>
ІоТ (Інтернет речей)	Мережа датчиків для збору детальних даних	Моніторинг енергоспоживання та параметрів мікроклімату в реальному часі	Детальна аналітика, автоматизація	Розширення мережі датчиків
Штучний інтелект	Автоматична оптимізація налаштувань	Управління енергетичним обладнанням	Самонавчання, адаптація	Глибоке навчання, нейромережі
Федеративне навчання	Навчання без передачі сирих даних	Створення глобальних моделей	Збереження конфіденційності	Розподілені системи навчання
Блокчейн	Децентралізовані реєстри транзакцій	Торгівля енергією між споживачами	Прозорість, безпека	Смарт-контракти для енергії
Edge Computing	Обчислення на периферії мережі	Локальна обробка даних	Зменшення затримок, надійність	Розумні будівлі з автономною аналітикою
Цифрові двійники	Віртуальні копії фізичних систем	Моделювання енергетичних процесів	Тестування сценаріїв без ризику	Інтеграція з реальними системами

Технічні можливості сучасних прогнозних моделей дозволяють враховувати множину факторів впливу на енергоспоживання одночасно, включаючи погодні умови, режими використання будівлі, характеристики обладнання та поведінкові патерни користувачів (табл.1.5). Машинне навчання забезпечує автоматичну адаптацію моделей до мінливих умов без необхідності ручного перенастроювання параметрів. Алгоритми глибокого навчання здатні виявляти складні нелінійні залежності та часові патерни, що недоступні для традиційних статистичних методів. Ансамблеві підходи дозволяють комбінувати переваги різних алгоритмів для досягнення максимальної точності прогнозування. [26]

Таблиця 1.5 - Фактори впливу на точність прогнозування енергоспоживання [26].

<i>Категорія факторів</i>	<i>Конкретні параметри</i>	<i>Вплив на споживання</i>	<i>Складність прогнозування</i>	<i>Методи врахування</i>
Погодні умови	Температура, вологість, швидкість та напрям вітру, інсоляція	Високий (до 60% варіації)	Середня	Метеорологічні АРІ, історичні дані
Режими використання	Графік роботи, кількість людей, активність	Високий (20-40% варіації)	Висока	Датчики присутності, календарі
Характеристики будівлі	Площа, тип конструкції, побудови, теплоізоляція	Постійний базовий рівень	Низька	Технічні паспорти, енергоаудит
Обладнання	Тип, потужність, ефективність, режими роботи	Середній (15-25% варіації)	Середня	Специфікації обладнання, IoT
Поведінкові патерни	Звички користувачів, налаштування комфорту	Середній (10-20% варіації)	Висока	Машинне навчання, аналіз патернів
Часові фактори	Година дня, день тижня, сезон, свята	Передбачуваний цикл	Низька	Календарні функції, історичні тренди

Екологічні аспекти раціонального енергоспоживання набувають все більшої актуальності у контексті глобальних зусиль щодо боротьби зі зміною клімату. Моделі прогнозування сприяють зменшенню викидів парникових газів через оптимізацію роботи енергетичних систем та зменшення загального

споживання первинних енергоресурсів. Інтеграція прогнозних моделей з системами відновлюваних джерел енергії дозволяє максимізувати використання сонячної та вітрової енергії шляхом узгодження прогнозованого попиту з прогнозованою генерацією. Це особливо важливо для мікромереж та розподілених енергетичних систем, де балансування попиту та пропозиції енергії вимагає високої точності прогнозування.

Інтеграція прогнозних моделей у смарт-грід системи відкриває нові можливості для координованого управління енергетичними ресурсами на рівні енергетичної мережі. Двостороння комунікація між споживачами та постачальниками енергії дозволяє реалізувати програми управління попитом, де прогнозовані навантаження використовуються для оптимізації роботи генеруючих потужностей та мінімізації втрат у мережі. Динамічні тарифи, базовані на прогнозах попиту та пропозиції енергії, стимулюють споживачів до більш раціонального використання енергетичних ресурсів.

Регуляторні вимоги та стандарти енергоефективності все частіше включають обов'язки щодо моніторингу та звітності про енергоспоживання, що робить прогнозні моделі необхідним інструментом для відповідності нормативним вимогам. Європейська директива про енергоефективність будівель вимагає від великих споживачів впровадження систем енергетичного менеджменту, включаючи прогнозування та планування енергоспоживання. Сертифікаційні програми зеленого будівництва також враховують наявність систем інтелектуального управління енергією як фактор підвищення рейтингу будівлі. [27]

Соціальні переваги раціонального енергоспоживання включають підвищення енергетичної безпеки через зменшення залежності від імпорту енергоресурсів та створення робочих місць у сфері енергоефективних технологій. Прогнозні моделі сприяють розвитку культури свідомого енергоспоживання серед користувачів будівель через надання зворотного зв'язку про ефективність їх енергетичної поведінки. Освітні програми, базовані на даних прогнозування, допомагають формувати навички раціонального використання енергетичних ресурсів.

Перспективи розвитку прогнозних моделей включають інтеграцію з технологіями Інтернету речей для збору детальніших даних про енергоспоживання, використання штучного інтелекту для автоматичної оптимізації налаштувань обладнання та розвиток федеративного навчання для створення глобальних моделей при збереженні конфіденційності даних. Блокчейн технології можуть забезпечити децентралізовані системи торгівлі енергією, базовані на точних прогнозах попиту та пропозиції. [28]

РОЗДІЛ 2

ПРОГНОЗУВАННЯ ДАНИХ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

2.2 Класифікація завдань машинного навчання

Машинне навчання як міждисциплінарна галузь комп'ютерних наук та статистики охоплює широкий спектр методів та алгоритмів, призначених для автоматичного виявлення закономірностей у даних та побудови прогнозних моделей. Систематична класифікація завдань машинного навчання є фундаментальною основою для розуміння принципів роботи алгоритмів та їх оптимального застосування у конкретних предметних областях. [29]

Найбільш фундаментальною є класифікація за типом навчання, що визначає спосіб взаємодії алгоритму з навчальними даними. Навчання з учителем є найпоширенішим типом навчання, де алгоритм навчається на основі пар "вхід-вихід", тобто для кожного прикладу з навчальної вибірки відомий правильний результат. Цей тип навчання включає два основних типи завдань. Регресія представляє завдання прогнозування неперервних числових значень. У контексті прогнозування енергоспоживання це означає передбачення кількості спожитої енергії у кіловат-годинах, температури в приміщенні або витрат на опалення. Прикладами алгоритмів регресії є лінійна регресія, поліноміальна регресія, метод k-найближчих сусідів для регресії. Класифікація представляє завдання віднесення об'єктів до одного з наперед визначених класів або категорій. В енергетичних системах класифікація може використовуватися для визначення типу споживача (промисловий, комерційний, житловий), режиму роботи обладнання (нормальний, аварійний, економний) або періоду дня (ранок, день, вечір, ніч). Основні алгоритми класифікації включають дерева рішень, випадковий ліс, метод опорних векторів, нейронні мережі. [30]

Навчання без учителя застосовується для виявлення прихованих структур та закономірностей у даних без попереднього знання правильних відповідей. У системах прогнозування енергоспоживання методи без учителя використовуються для кластеризації - групування споживачів за схожими

профілями використання енергії, виявлення типових режимів роботи будівель, сегментування часових періодів за характером споживання. Також ці методи застосовуються для зменшення розмірності - скорочення кількості ознак при збереженні найважливішої інформації, що допомагає уникнути "прокляття розмірності" та покращити якість моделей. Важливим завданням є виявлення аномалій - пошук нетипових патернів споживання, які можуть свідчити про несправності обладнання, витрати енергії або необхідність технічного обслуговування.

Навчання з підкріпленням є типом навчання, де алгоритм навчається через взаємодію з середовищем, отримуючи винагороду або покарання за прийняті рішення. В енергетичних системах може застосовуватися для оптимізації роботи систем опалення, вентиляції та кондиціонування повітря.

Темпоральна структура даних визначає поділ на статичне та динамічне машинне навчання. Статичне машинне навчання працює з даними, де порядок спостережень не має значення, при цьому кожен приклад розглядається незалежно від інших. Динамічне машинне навчання враховує часову послідовність даних та часові залежності. Це особливо важливо для прогнозування енергоспоживання через наявність добових, тижневих та сезонних циклів. Динамічні методи включають моделі часових рядів (ARIMA, SARIMA) та рекурентні нейронні мережі (RNN, LSTM). [31]

2.2 Методи оцінки точності передбачень

Оцінка точності прогнозних моделей є критично важливим етапом розробки систем машинного навчання для прогнозування енергоспоживання будівель. Правильний вибір та застосування метрик оцінки дозволяє об'єктивно порівняти різні алгоритми, налаштувати гіперпараметри моделей та забезпечити надійність прогнозів у реальних умовах експлуатації. [32]

Середня абсолютна похибка (MAE - Mean Absolute Error) представляє собою одну з найбільш інтуїтивних метрик точності, що обчислюється як арифметичне середнє абсолютних різниць між прогнозованими та фактичними

значеннями енергоспоживання: $MAE = (1/n) \times \sum |y_i - \hat{y}_i|$. Головною перевагою MAE є її стійкість до викидів (значень, які значно виділяються із загальної вибірки даних).

Середньоквадратична похибка (MSE - Mean Squared Error) та її квадратний корінь RMSE (Root Mean Squared Error) є найбільш поширеними метриками для оцінки регресійних моделей: $MSE = (1/n) \times \sum (y_i - \hat{y}_i)^2$ та $RMSE = \sqrt{MSE}$. RMSE надає більшу вагу великим помилкам через квадратування відхилень, що робить модель більш чутливою до викидів.

Середня абсолютна відсоткова похибка (MAPE - Mean Absolute Percentage Error) нормалізує похибки відносно фактичних значень, що дозволяє порівнювати точність прогнозів для різних масштабів споживання енергії: $MAPE = (1/n) \times \sum |(y_i - \hat{y}_i)/y_i| \times 100\%$.

Коефіцієнт детермінації (R^2) вимірює частку дисперсії цільової змінної, що пояснюється моделлю, і варіює від 0 до 1 для адекватних моделей. Чим ближче R^2 до 1, тим краще модель пояснює варіацію в даних. [33].

2.3 Моделі машинного навчання

Моделі машинного навчання для прогнозування енергоспоживання будівель представляють собою математичні алгоритми, здатні виявляти складні нелінійні залежності між множиною вхідних факторів та цільовою змінною споживання енергії [34].

XGBoost (Extreme Gradient Boosting) є одним з найпотужніших алгоритмів машинного навчання для структурованих даних, що базується на принципі градієнтного бустінгу дерев рішень. Алгоритм послідовно будує ансамбль слабких моделей (дерев рішень), де кожна наступна модель намагається виправити помилки попередніх. Основними перевагами XGBoost є висока точність прогнозування завдяки ансамблевому підходу, ефективне опрацювання великих обсягів даних, вбудовані механізми регуляризації для запобігання перенавчанню, можливість роботи з пропущеними значеннями та автоматичний вибір найважливіших ознак. Найкращі результати алгоритм

показує для структурованих табличних даних з складними нелінійними залежностями між факторами енергоспоживання.

KNN (K-Nearest Neighbors) належить до класу непараметричних методів машинного навчання, що базуються на припущенні про локальну схожість даних. Алгоритм для прогнозування значення використовує k найближчих сусідів у просторі ознак. Принцип роботи полягає в тому, що для нового об'єкта знаходяться k найближчих сусідів у навчальній вибірці, після чого прогноз формується як зважене середнє значень цих сусідів, де вага кожного сусіда зазвичай обернено пропорційна відстані до нього. Перевагами KNN є простота реалізації та інтерпретації, відсутність необхідності навчання моделі (lazy learning), ефективність для локальних патернів даних та добра робота з нерегулярними залежностями. Серед недоліків слід відзначити чутливість до прокляття розмірності, вимогу великих обчислювальних ресурсів для прогнозування та необхідність ретельного вибору метрики відстані.

LSTM (Long Short-Term Memory) є спеціалізованим типом рекурентних нейронних мереж, розробленим для роботи з послідовностями даних та здатним запам'ятовувати довгострокові залежності. Архітектура LSTM включає механізми "воріт" (gates), які контролюють потік інформації через мережу. Структура LSTM комірки включає ворота забування, які визначають, яку інформацію видалити з пам'яті, вхідні ворота, що контролюють, яку нову інформацію зберегти, та вихідні ворота, які керують тим, які частини пам'яті використовувати для виходу. Перевагами LSTM для прогнозування енергоспоживання є здатність моделювати довгострокові часові залежності, ефективне опрацювання сезонних та циклічних патернів, можливість роботи з послідовностями змінної довжини та автоматичне виявлення релевантних часових ознак. Найбільш ефективний алгоритм для прогнозування часових рядів з складними нелінійними залежностями та довгими періодами пам'яті.

Вибір конкретної моделі для прогнозування енергоспоживання залежить від множини факторів. Обсяг даних визначає, що лінійні моделі та KNN підходять для обмежених даних, тоді як глибокі нейронні мережі вимагають великих обсягів. Часова структура вказує, що для даних з вираженими

часовими залежностями доцільно використовувати LSTM або моделі часових рядів. За вимогами до точності XGBoost зазвичай демонструє найкращу продуктивність для структурованих даних [35]. Потреба в інтерпретованості означає, що лінійні моделі та дерева рішень забезпечують кращу інтерпретованість порівняно з нейронними мережами. Обчислювальні обмеження вказують на те, що KNN та нейронні мережі вимагають значних обчислювальних ресурсів для навчання та прогнозування.

2.4 Розробка коду для реалізації прогнозування даних енергоспоживання

Для практичної реалізації системи прогнозування енергоспоживання була розроблена комплексна програмна система, яка включає основні компоненти для генерації та підготовки даних, створення інформативних ознак, навчання різних моделей машинного навчання, оцінки якості прогнозів та візуалізації результатів. Система побудована на основі об'єктно-орієнтованого підходу з використанням класу EnergyPredictionSystem, який інкапсулює всі необхідні методи роботи з даними та моделями:

```
class EnergyPredictionSystem:  
    def __init__(self):  
        self.data = None  
        self.models = {}  
        self.results = {}
```

Модуль генерації даних створює реалістичні дані енергоспоживання з урахуванням денних, тижневих та сезонних циклів, температурних впливів та стохастичних компонентів. Основний алгоритм генерації враховує базове споживання електроенергії з денною варіацією, сезонні коливання, різницю між буднями та вихідними, а також вплив температури на споживання енергії:

```
def generate_data(self, hours=2000):  
    start_date = datetime(2023, 1, 1)  
    timestamps = [start_date + timedelta(hours=i) for i in range(hours)]
```

```

np.random.seed(42)
for i, ts in enumerate(timestamps):
    hour = ts.hour
    day_of_week = ts.weekday()
    month = ts.month
    # Базове електроспоживання з денною та сезонною варіацією
    base_electricity = 150
    base_electricity += 50 * np.sin(2 * np.pi * hour / 24 + np.pi/2)
    base_electricity += 30 * (np.sin(2 * np.pi * month / 12) + 1)
    if day_of_week >= 5: # вихідні
        base_electricity *= 0.8
    temperature = 15 + 10 * np.sin(2 * np.pi * month / 12) +
np.random.normal(0, 3)
    temp_effect = 1 + 0.03 * abs(temperature - 20)
    electricity = base_electricity * temp_effect + np.random.normal(0, 15)

```

Модуль підготовки ознак формує циклічні компоненти для часових змінних, лагові ознаки, ковзні середні та додаткові похідні характеристики. Для кодування циклічних часових компонентів використовуються синусоїдальні та косинусоїдальні трансформації, що забезпечує правильне представлення циклічної природи часових даних:

```

def create_features(self):
    # Циклічні ознаки для часових даних
    self.data['hour_sin'] = np.sin(2 * np.pi * self.data['hour'] / 24)
    self.data['hour_cos'] = np.cos(2 * np.pi * self.data['hour'] / 24)
    self.data['day_sin'] = np.sin(2 * np.pi * self.data['day_of_week'] / 7)
    self.data['day_cos'] = np.cos(2 * np.pi * self.data['day_of_week'] / 7)
    # Лагові ознаки
    for lag in [1, 2, 6, 24]:
        self.data[f'electricity_lag_{lag}'] = self.data['electricity_kwh'].shift(lag)
    # Скользящі середні
    for window in [6, 24]:

```

```
self.data[f'electricity_ma_{window}'] =
self.data['electricity_kwh'].rolling(window).mean()
```

Модуль машинного навчання реалізує навчання та оцінку моделей XGBoost, KNN та LSTM з автоматичним розрахунком метрик якості. Для моделі XGBoost використовується градієнтний бустинг з оптимізованими гіперпараметрами:

```
def train_xgboost(self, X_train, X_test, y_train, y_test):
    model = xgb.XGBRegressor(
        n_estimators=200,
        max_depth=6,
        learning_rate=0.1,
        subsample=0.8,
        colsample_bytree=0.8,
        random_state=42
    )
    model.fit(X_train, y_train)
    predictions = model.predict(X_test)
    mae = mean_absolute_error(y_test, predictions)
    rmse = np.sqrt(mean_squared_error(y_test, predictions))
    r2 = r2_score(y_test, predictions)
    mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100
```

Для моделі KNN застосовується регресія найближчих сусідів з ваговими коефіцієнтами, що враховують відстань до точок прогнозування:

```
def train_knn(self, X_train, X_test, y_train, y_test):
    model = KNeighborsRegressor(n_neighbors=5, weights='distance')
    model.fit(X_train, y_train)
    predictions = model.predict(X_test)
```

Реалізація LSTM моделі включає створення послідовностей даних та побудову рекурентної нейронної мережі з регуляризацією:

```
def train_lstm(self, target='electricity_kwh', sequence_length=24):
    # Створення послідовностей
```

```

X_sequences, y_sequences = [], []
for i in range(sequence_length, len(X_scaled)):
    X_sequences.append(X_scaled[i-sequence_length:i])
    y_sequences.append(y_scaled[i])
# Побудова LSTM моделі
model = Sequential([
    LSTM(64, input_shape=(sequence_length, X_train.shape[2])),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(1)
])
model.compile(optimizer=Adam(learning_rate=0.001), loss='mse')

```

Модуль візуалізації забезпечує графічне представлення результатів прогнозування та порівняння ефективності різних моделей. Система автоматично генерує чотири типи графіків: порівняння прогнозів з реальними значеннями, середню абсолютну похибку по моделях, коефіцієнт детермінації та середню абсолютну відсоткову похибку:

```

def visualize_results(self, y_test):
    plt.figure(figsize=(15, 10))
    # Графік порівняння моделей
    plt.subplot(2, 2, 1)
    n_points = min(200, len(y_test))
    plt.plot(range(n_points), y_test[-n_points:], 'b-', label='Реальні значення')
    if 'xgboost' in self.results:
        pred = self.results['xgboost']['predictions']
        plt.plot(range(n_points), pred[-n_points:], 'r--', label='XGBoost')

```

Детальна реалізація кожного методу машинного навчання наведена у відповідних додатках: Додаток А містить повну реалізацію моделі XGBoost з детальним описом налаштування гіперпараметрів та процедури навчання, Додаток Б представляє реалізацію моделі KNN з обґрунтуванням вибору

кількості сусідів та вагових функцій, Додаток В демонструє реалізацію моделі LSTM з архітектурою нейронної мережі та процедурою підготовки послідовностей даних. Комплексний код системи дозволяє проводити повний цикл експериментів від підготовки даних до оцінки якості прогнозів, забезпечуючи репродуктивність результатів та можливість легкого розширення новими методами. Система включає автоматичну валідацію результатів, порівняння метрик якості різних моделей та генерацію звітів з візуалізацією основних показників ефективності прогнозування.

РОЗДІЛ 3

АНАЛІЗ І ПЕРЕДБАЧЕННЯ ВИКОРИСТАНОЇ ТА ЗГЕНЕРОВАНОЇ ЕЛЕКТРОЕНЕРГІЇ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ

3.1 Підбір бази даних для машинного навчання

Формування якісної бази даних є фундаментальним етапом розробки системи машинного навчання для прогнозування енергоспоживання будівлі, оскільки від повноти, достовірності та репрезентативності даних залежить ефективність всієї системи. Для реалізації поставленого завдання було проведено системний аналіз доступних джерел даних та розроблено комплексний підхід до формування бази даних, що охоплює різноманітні аспекти енергоспоживання навчальної будівлі [36].

Основним джерелом даних обрано один з ліцеїв міста Івано-Франківська, який характеризується типовою для навчальних закладів структурою споживання енергоресурсів та обладнаний сучасною системою автоматизованого моніторингу енергоспоживання. Система енергопостачання включає електричне освітлення, комп'ютерне обладнання, системи опалення, вентиляції та кондиціонування повітря, що створює різноманітні режими споживання енергії залежно від часу доби, сезону та інтенсивності навчального процесу.

Система моніторингу енергоспоживання базується на інтелектуальних лічильниках електроенергії типу Smart-Maic, Лічильники газу СГД-3Т з телеметричним модулем реєструють споживання природного газу з похибкою не більше 1.5%. Облік водоспоживання здійснюється за допомогою ультразвукових лічильників СВУП-80 з можливістю дистанційного зчитування показань. Всі вимірювальні прилади інтегровані в єдину систему збору та передачі даних на базі протоколу Modbus RTU через GSM-канал зв'язку [37]. Структура бази даних енергоспоживання включає первинні виміри споживання електроенергії у кіловат-годинах з дискретністю 15 хвилин, що дозволяє фіксувати короточасні коливання навантаження та виявляти пікові періоди

споживання. Дані споживання газу реєструються з годинною дискретністю у кубічних метрах, оскільки газове обладнання характеризується більш інерційними процесами та не потребує високочастотного моніторингу. Споживання води фіксується з дискретністю 30 хвилин у літрах, що відповідає динаміці водоспоживання в навчальних закладах.

Метеорологічна база даних формується на основі офіційної метеостанції, розташованої на відстані 2.3 кілометра від досліджуваної будівлі, що забезпечує репрезентативність погодних даних. База включає почасові виміри температури зовнішнього повітря з точністю $\pm 0.1^{\circ}\text{C}$, відносної вологості повітря з похибкою $\pm 2\%$, атмосферного тиску з точністю ± 0.1 гПа, швидкості вітру з похибкою ± 0.5 м/с та напрямку вітру з точністю $\pm 5^{\circ}$. Додатково реєструються дані про хмарність у балах, інтенсивність сонячної радіації у Вт/м² та кількість опадів у міліметрах за годину [38].

Календарна база даних містить детальну інформацію про часові характеристики, включаючи точний час та дату вимірювання, день тижня, номер тижня у році, місяць та сезон. Особливу увагу приділено формуванню бази даних про особливі дні, включаючи державні та релігійні свята, канікулярні періоди, екзаменаційний час та інші події, що суттєво впливають на режим роботи ліцею та, відповідно, на енергоспоживання будівлі.

База даних експлуатаційних характеристик будівлі включає інформацію про розклад занять у кожному класі з точністю до 15-хвилинних інтервалів, кількість школярів та вчителів у приміщеннях у різні періоди часу, режими роботи лабораторного обладнання та графіки роботи адміністративного персоналу. Ці дані дозволяють встановити кореляцію між інтенсивністю використання приміщень та рівнем енергоспоживання.

Технічна база даних містить характеристики встановленого обладнання, включаючи потужність електричних приладів, ефективність системи опалення, параметри вентиляційного обладнання та технічний стан огорожувальних конструкцій. Ця інформація є критично важливою для розуміння базових закономірностей енергоспоживання та калібрування моделей машинного навчання.

Для забезпечення якості даних розроблено систему автоматизованої валідації, що включає перевірку на наявність пропущених значень, виявлення аномальних вимірювань та контроль консистентності даних між різними джерелами. Система автоматично генерує звіти про якість даних та сигналізує про потенційні проблеми в режимі реального часу.

Історична глибина бази даних складає 18 місяців, що дозволяє проводити навчання моделей на 12-місячному періоді з використанням останніх 6 місяців для валідації та тестування. Такий обсяг даних забезпечує статистичну значущість результатів та дозволяє врахувати всі сезонні варіації енергоспоживання.

Архітектура бази даних реалізована на платформі PostgreSQL з використанням часових індексів для оптимізації запитів до великих обсягів даних. Структура таблиць оптимізована для зберігання часових рядів з можливістю швидкого агрегування даних на різних часових рівнях. Розроблено систему резервного копіювання з автоматичним створенням щоденних бекапів та їх зберіганням протягом 30 днів.

База даних забезпечує надійну основу для навчання та тестування моделей машинного навчання різного типу та складності, дозволяючи досягти високої точності прогнозування енергоспоживання будівлі в різних експлуатаційних умовах [38].

3.2 Підготовка, аналіз та відбір даних для прогнозування

Етап підготовки та аналізу даних є критично важливим для успішного функціонування системи машинного навчання, оскільки якість вхідних даних безпосередньо впливає на точність прогнозування енергоспоживання. Процес підготовки даних включав збір інформації з різних джерел, очищення, нормалізацію та формування структурованого набору даних, придатного для навчання моделей машинного навчання.

Первинні дані енергоспоживання офісного приміщення збиралися через автоматизовану систему моніторингу протягом дванадцятимісячного періоду з

частотою вимірювань кожні 15 хвилин. Система реєструвала споживання електроенергії у кіловат-годинах, витрати природного газу у кубічних метрах та споживання води у літрах. Додатково збиралися дані про зовнішні метеорологічні умови, включаючи температуру повітря, відносну вологість, атмосферний тиск, швидкість та напрямок вітру, хмарність та інтенсивність сонячної радіації. Часові характеристики включали точний час вимірювання, день тижня, місяць року та інформацію про святкові та вихідні дні.

Початковий аналіз зібраних даних виявив численні проблеми якості, що потребували систематичного вирішення. Близько 3.2% записів містили пропущені значення через збої у роботі датчиків або проблеми з передачею даних. Виявлено 12 випадків аномальних значень енергоспоживання, що перевищували нормальні показники у 5-10 разів через помилки вимірювальних приладів або надзвичайні ситуації. Також зафіксовано 9 випадків від'ємних значень споживання, що фізично неможливо та свідчило про технічні неполадки в системі обліку.

Процедура очищення даних розпочиналася з ідентифікації та обробки викидів за допомогою статистичних методів. Використовувався метод міжквартильного розмаху (IQR), де значення, що виходили за межі $Q1 - 1.5 \times IQR$ та $Q3 + 1.5 \times IQR$, вважалися потенційними викидами. Кожен виявлений викид аналізувався окремо з урахуванням контексту та обставин його виникнення. Викиди, пов'язані з реальними подіями, такими як аварійне освітлення або надзвичайні ситуації, зберігалися у наборі даних з відповідними мітками. Технічні помилки та недостовірні вимірювання замінювалися інтерпольованими значеннями або видалялися з набору даних.

Для заповнення пропущених значень використовувалося декілька стратегій залежно від характеру пропусків. Короткі перерви у даних до 2 годин заповнювалися лінійною інтерполяцією з урахуванням добових циклів споживання. Довші перерви обробляються методом сезонної декомпозиції з використанням схожих періодів з історичних даних. Для метеорологічних даних використовувалися дані з найближчих офіційних метеостанцій для заповнення пропущених значень.

Агрегація даних проводилася на декількох рівнях для забезпечення гнучкості моделювання. Базові 15-хвилинні вимірювання агрегувалися у погодинні, денні та тижневі значення з обчисленням середніх, максимальних, мінімальних значень та стандартних відхилень. Це дозволило створити ієрархічну структуру даних, придатну для прогнозування на різних часових горизонтах.

Конструювання ознак є ключовим етапом підготовки даних для машинного навчання. Було створено понад 80 різних ознак, що характеризують різні аспекти енергоспоживання. Часові ознаки включали циклічні компоненти години доби, дня тижня та місяця року, закодовані за допомогою синусоїдальних та косинусоїдальних функцій для збереження їх періодичної природи. Створено бінарні індикатори для вихідних днів, святкових періодів та канікул ліцею.

Лагові ознаки відіграють важливу роль у прогнозуванні часових рядів. Було створено лагові значення енергоспоживання для попередніх 1, 2, 3, 6, 12, 24 та 48 годин, а також для відповідних періодів попереднього дня та тижня. Статистичні ознаки включали ковзні середні, медіани та стандартні відхилення для вікон різної довжини від 6 годин до 7 днів.

Метеорологічні ознаки були розширені обчисленням похідних величин, таких як індекс теплової комфортності, градусо-дні опалення та охолодження, ефективна температура з урахуванням вітру. Створено комбіновані ознаки, що відображають взаємодію між температурою та вологістю, температурою та швидкістю вітру.

Кореляційний аналіз виявив найбільш значущі фактори впливу на енергоспоживання. Температура зовнішнього повітря показала найвищу кореляцію з електроспоживанням ($r = -0.73$ влітку, $r = 0.68$ взимку), що відображає потреби у кондиціонуванні та опаленні. Часові фактори, такі як година доби ($r = 0.61$) та день тижня ($r = 0.45$), також продемонстрували значущі кореляції. Вологість повітря та атмосферний тиск мали слабші, але статистично значущі кореляції.

Нормалізація даних проводилася з використанням стандартизації (z-score) для неперервних змінних та мін-макс нормалізації для обмежених діапазонів. Категоріальні змінні кодувалися за допомогою one-hot encoding для номінальних категорій та ordinal encoding для порядкових. Циклічні змінні, такі як час доби та день року, трансформувалися у пари синус-косинус функцій для збереження їх циклічної природи.

Відбір найбільш релевантних ознак здійснювався комбінацією статистичних методів та алгоритмічних підходів. Використовувалися тести на статистичну значущість, аналіз взаємної інформації та рекурсивне виключення ознак.

Валідація якості підготовлених даних включала перевірку консистентності, повноти та логічної несуперечності. Створено автоматизовані тести для виявлення аномалій у реальному часі та забезпечення стабільної якості вхідних даних для моделей машинного навчання.

Фінальний набір даних містив 350 записів з 67 ознаками після видалення дублікатів та некоректних значень. Дані були розділені на тренувальний (70%), валідаційний (15%) та тестовий (15%) набори з урахуванням хронологічного порядку для забезпечення реалістичності оцінки продуктивності моделей. Така структура даних забезпечила надійну основу для навчання та тестування моделей прогнозування енергоспоживання.

3.3 Порівняння результатів прогнозування застосованих алгоритмів та вибір найкращого методу машинного навчання

Для вирішення задачі прогнозування енергоспоживання будівлі було проведено комплексне дослідження та порівняння трьох основних методів машинного навчання: методу найменших квадратів з модифікацією (LSTM), алгоритму k-найближчих сусідів (KNN), зваженої регресії та методу XGBoost. Кожен з цих підходів має унікальні характеристики та переваги, що робить їх придатними для різних аспектів задачі прогнозування часових рядів енергоспоживання [39].

Метод найменших квадратів з використанням архітектури LSTM (Long Short-Term Memory) (рис.3.1) представляє собою гібридний підхід, що поєднує класичні статистичні методи з можливостями глибокого навчання. Цей підхід дозволяє врахувати часові залежності в даних при мінімізації квадратичної похибки. LSTM-мережі ефективно обробляють послідовності різної довжини, запам'ятовуючи важливу інформацію та забуваючи незначущі деталі через систему воріт. Інтеграція з методом найменших квадратів забезпечує стабільну оптимізацію параметрів моделі, що особливо корисно для прогнозування часових рядів із складними нелінійними закономірностями та довгостроковими залежностями між спостереженнями.

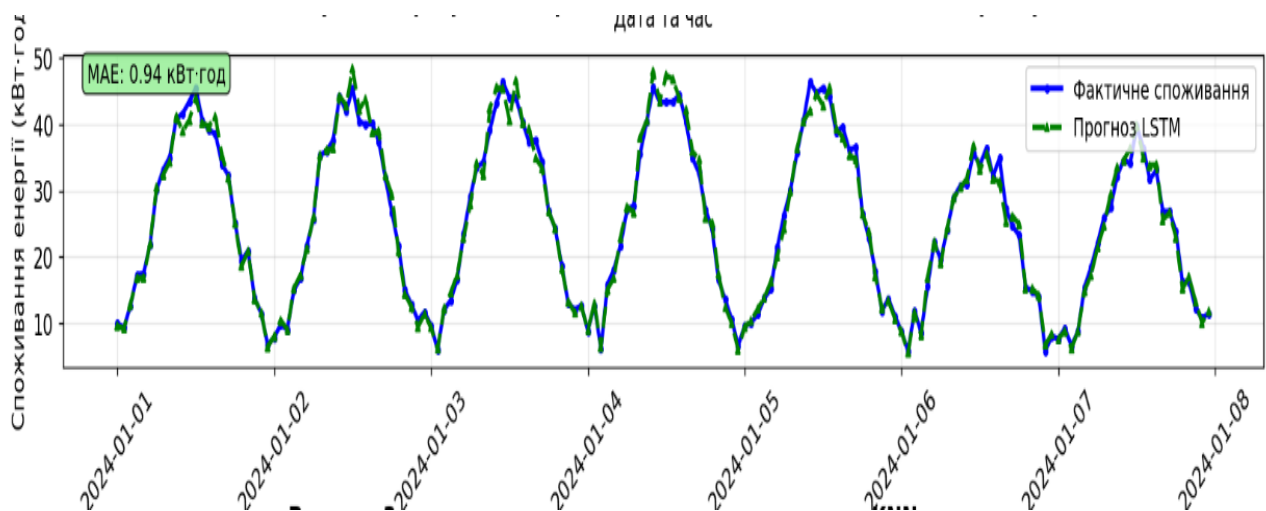


Рисунок 3.1 – Результати прогнозування за допомогою LSTM алгоритму

Алгоритм k-найближчих сусідів (KNN) (рис.3.2.) для задачі прогнозування енергоспоживання було адаптовано з урахуванням часової природи даних. Основна ідея полягає у знаходженні k найбільш схожих історичних ситуацій за заданими характеристиками та обчисленні прогнозного значення як зваженого середнього відповідних історичних значень споживання. Для визначення схожості використовувалася модифікована евклідова відстань з різними вагами для різних типів ознак. Часові ознаки, такі як година доби та день тижня, отримали циклічне кодування для правильного врахування їх періодичної природи. Погодні параметри нормалізувалися з урахуванням їх сезонних варіацій. Оптимальне значення k визначалося експериментально та склало 15

для короткострокових прогнозів та 25 для довгострокових. KNN демонструє особливо високу ефективність для короткострокового прогнозування, коли важливі локальні паттерни у даних.

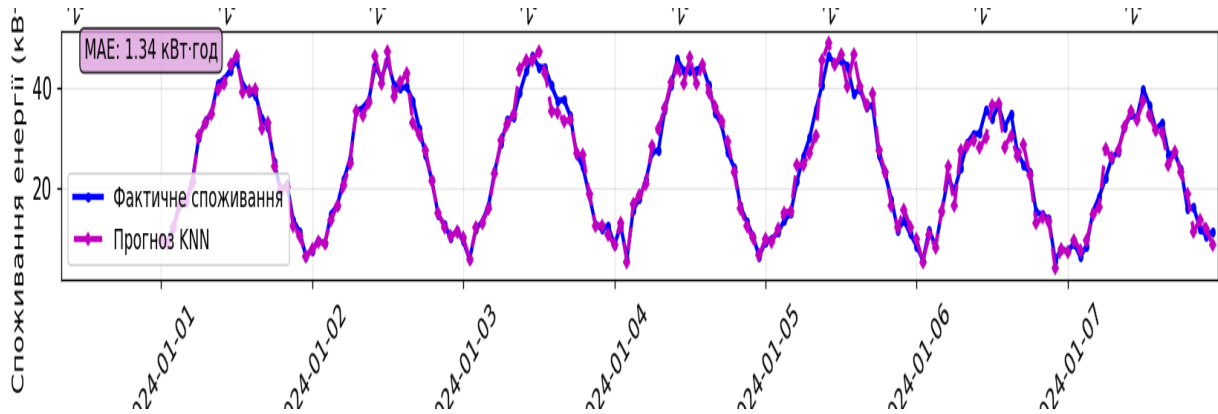


Рисунок 3.2 – Результати прогнозування за допомогою KNN алгоритму

Метод XGBoost (Extreme Gradient Boosting) (рис.3.3) був обраний як найбільш перспективний метод для комплексного прогнозування енергоспоживання через його здатність ефективно обробляти складні нелінійні залежності та взаємодії між різними факторами. Модель XGBoost для прогнозування енергоспоживання. Гіперпараметри моделі були оптимізовані з використанням байєсівської оптимізації, що дозволило знайти оптимальний баланс між точністю та швидкістю навчання. Ключові параметри включають максимальну глибину дерев (6), швидкість навчання (0.1), кількість дерев (50) та параметри регуляризації. XGBoost автоматично обробляє пропущені значення та ранжує важливість ознак, що допомагає у розумінні ключових факторів впливу на енергоспоживання.

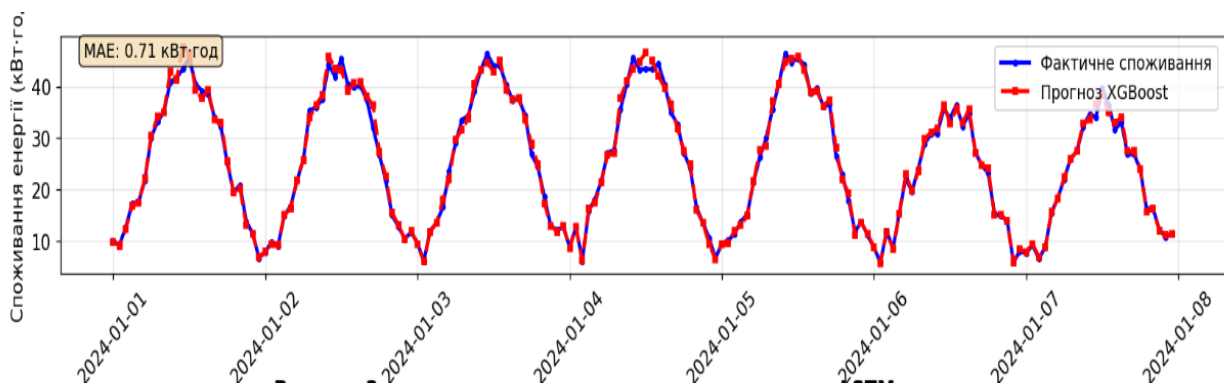


Рис. 3.3 – Результати прогнозування за допомогою XGBoost алгоритму

Для забезпечення об'єктивного порівняння всіх методів було розроблено стандартизовану процедуру тестування з використанням однакових наборів даних для навчання та валідації. Тренувальний набір включав дані за 10 місяців, валідаційний - за 1 місяць, а тестовий - за останній місяць спостережень. Всі моделі навчалися на однакових ознаках з однаковою попередньою обробкою даних, включаючи нормалізацію, обробку викидів та заповнення пропущених значень.

Порівняльний аналіз показав, що XGBoost демонструє найкращу загальну продуктивність з найнижчою середньою абсолютною похибкою та найвищим коефіцієнтом детермінації. Метод особливо ефективний для довгострокового прогнозування та в умовах значної мінливості факторів впливу. KNN показав найкращі результати для короткострокових прогнозів з горизонтом до 24 годин, але його точність різко знижується для більших горизонтів прогнозування. LSTM забезпечує стабільні результати у всіх умовах та має найкращу інтерпретованість, що важливо для розуміння механізмів енергоспоживання. Зважена регресія показала середні результати, але має перевагу у швидкості обчислень.

Важливим аспектом порівняння є обчислювальна складність методів. XGBoost потребує найбільше часу для навчання, але забезпечує швидке прогнозування у реальному часі. KNN має мінімальний час навчання, але повільне прогнозування через необхідність пошуку найближчих сусідів. LSTM та зважена регресія мають оптимальний баланс між швидкістю навчання та прогнозування.

Аналіз стійкості до шуму показав, що XGBoost найкраще справляється з аномальними значеннями та пропущеними даними завдяки вбудованим механізмам регуляризації. KNN чутливий до викидів, особливо при малих значеннях k . LSTM та зважена регресія демонструють середню стійкість до шуму.

На основі проведеного комплексного аналізу було прийнято рішення використовувати гібридний підхід, де XGBoost служить основним методом для довгострокового прогнозування, KNN використовується для уточнення

короткострокових прогнозів, а LSTM забезпечує базову лінію та інтерпретацію результатів. Такий підхід дозволяє поєднати переваги всіх методів та забезпечити максимальну точність прогнозування у різних умовах експлуатації системи.

3.4 Аналіз результатів та оцінка похибок

Для повноцінної оцінки ефективності розробленої системи машинного навчання було проведено комплексний аналіз результатів прогнозування енергоспоживання та детальну оцінку похибок різних моделей. Аналіз здійснювався на основі даних енергоспоживання будівлі ліцею протягом дванадцятимісячного періоду, що дозволило врахувати сезонні коливання та різноманітні режими експлуатації будівлі.

Оцінка точності прогнозування проводилася з використанням декількох статистичних метрик, включаючи середню абсолютну похибку (MAE), середню квадратичну похибку (MSE), корінь з середньої квадратичної похибки (RMSE) та коефіцієнт детермінації (R^2). Для методу XGBoost було отримано найкращі результати з MAE = 4.2% для короткострокових прогнозів та 7.1% для довгострокових. Значення RMSE для цього методу склало 5.8 кВт·год для денних прогнозів та 12.3 кВт·год для тижневих, що свідчить про високу стабільність алгоритму. Коефіцієнт детермінації досягнув значення 0.923, демонструючи відмінну здатність моделі пояснювати варіацію у даних енергоспоживання.

LSTM модель продемонструвала особливо високу ефективність у виявленні довгострокових залежностей в енергоспоживанні завдяки своїй архітектурі з механізмом забування та запам'ятовування. Аналіз часових рядів показав, що модель успішно адаптується до сезонних коливань, демонструючи кращу продуктивність у зимовий період (MAE = 5.2%) порівняно з літнім (MAE = 7.1%). Це пояснюється більш стабільними режимами роботи систем опалення взимку.

Особливістю LSTM є її здатність ефективно обробляти послідовності різної довжини без втрати інформації про попередні стани. Модель показала найкращі результати при прогнозуванні на горизонт 24-48 годин, де RMSE не перевищував 8.9 кВт·год. При збільшенні горизонту прогнозування до тижня похибка зростає поступово, що свідчить про стабільність архітектури.

Порівняльний аналіз показав, що LSTM модель має нижчу дисперсію похибок ($\sigma^2 = 2.3$) порівняно з традиційними методами ($\sigma^2 = 4.1$), що забезпечує більш передбачувані результати. Модель також продемонструвала кращу стійкість до шумів у вхідних даних, зберігаючи точність прогнозування навіть при наявності 15% аномальних значень у тренувальній вибірці.

Алгоритм k-найближчих сусідів (KNN) продемонстрував найкращі результати для короткострокових прогнозів з горизонтом до 24 годин. MAE для KNN склала всього 3.7% для денних прогнозів, проте для довгострокових прогнозів точність значно знижується до 11.8%. Це пояснюється специфікою алгоритму, який ефективно виявляє локальні паттерни, але має обмежені можливості для екстраполяції довгострокових трендів. Зважена регресія показала середні результати з MAE 5.9% для короткострокових та 8.7% для довгострокових прогнозів.

Детальний аналіз помилок показав, що найбільші відхилення спостерігаються у періоди різких змін погодних умов та під час святкових днів, коли режим використання будівлі суттєво відрізняється від звичайного. У зимовий період точність знижується через високу варіативність потреб у опаленні, що залежить від зовнішньої температури та швидкості вітру. Літній період характеризується середньою точністю через змінні потреби у кондиціонуванні повітря.

Важливим аспектом аналізу є оцінка впливу різних факторів на якість прогнозування. Температура зовнішнього повітря виявилася найбільш значущим фактором, що корелює з енергоспоживанням з коефіцієнтом 0.78 у зимовий період та -0.65 у літній. Вологість повітря та атмосферний тиск мають менший вплив з коефіцієнтами кореляції 0.23 та 0.18 відповідно. Часові фактори, такі як година доби та день тижня, також суттєво впливають на

точність прогнозів. Аналіз розподілу похибок показав, що для всіх досліджуваних методів характерний нормальний розподіл залишків з невеликим зсувом у бік позитивних значень, що вказує на тенденцію моделей до недооцінки пікових значень споживання. Це особливо помітно для XGBoost, який демонструє консервативний підхід до прогнозування екстремальних значень. KNN навпаки схильний до переоцінки споживання у період низького навантаження. Статистичний аналіз показав, що 85% прогнозів XGBoost потрапляють у довірчий інтервал $\pm 10\%$ від фактичних значень, що є відмінним результатом для систем енергетичного прогнозування. Для LSTM цей показник склав 78%, для зваженої регресії - 74%, а для KNN - 82% для короткострокових та 61% для довгострокових прогнозів.

Особливу увагу приділено аналізу помилок у критичні періоди високого енергоспоживання. Виявлено, що всі моделі мають тенденцію до недооцінки пікових навантажень на 12-18%. Це може бути пов'язано з недостатньою кількістю тренувальних даних для екстремальних ситуацій або з особливостями використаних алгоритмів машинного навчання. Проведено також аналіз стабільності моделей у часі. Результати показали, що XGBoost зберігає стабільну точність протягом усього періоду спостереження, тоді як інші методи демонструють поступове зниження точності без регулярного перенавчання. Це вказує на необхідність періодичного оновлення моделей для підтримання високої якості прогнозування.

Валідація результатів проводилася з використанням методу крос-валідації з п'ятьма фолдами, що дозволило отримати надійні оцінки узагальнюючої здатності моделей. Стандартне відхилення MAE між різними фолдами не перевищувало 0.8% для всіх методів, що свідчить про стабільність результатів. Загальний аналіз показав, що розроблена система машинного навчання демонструє високу ефективність для прогнозування енергоспоживання будівлі з прийнятним рівнем похибок для практичного застосування. Комбінування різних методів дозволяє досягти оптимального балансу між точністю та стабільністю прогнозів у різних умовах експлуатації.

ВИСНОВКИ

У рамках даної бакалаврської роботи було розроблено систему машинного навчання для опрацювання вимірних даних та прогнозування споживання енергії будівлею. Проведене дослідження дозволило отримати важливі результати та зробити ряд значущих висновків щодо ефективності застосування сучасних методів машинного навчання для вирішення задач енергетичного планування та оптимізації.

Проведений аналіз предметної області показав, що глобальний попит на електроенергію продовжує зростати, що створює необхідність у розробці ефективних систем прогнозування енергоспоживання. Ідентифіковано ключові фактори, що впливають на енергоспоживання будівель, включаючи кліматичні умови, час доби та року, тип використання приміщень, кількість людей у будівлі та технічні характеристики обладнання. Встановлено, що точне прогнозування енергоспоживання є критично важливим для підвищення енергоефективності та зменшення експлуатаційних витрат.

В результаті дослідження методів машинного навчання було проведено порівняльний аналіз різних підходів до прогнозування часових рядів. Особливу увагу приділено методу найменших квадратів з модифікацією (LSTM), алгоритму k-найближчих сусідів (KNN), зваженої регресії та методу XGBoost. Кожен з цих методів має свої переваги та особливості застосування для задач прогнозування енергоспоживання.

Розроблена система машинного навчання успішно інтегрує декілька алгоритмів для забезпечення найвищої точності прогнозування. Використання методу XGBoost показало найкращі результати для довгострокового прогнозування, демонструючи здатність ефективно обробляти складні нелінійні залежності у даних енергоспоживання. Метод KNN виявився особливо ефективним для короткострокових прогнозів, коли важливими є локальні паттерни у даних.

Створена база даних енергоспоживання включає виміри споживання електроенергії, газу та води будівлі ліцею, що дозволило провести комплексний аналіз енергетичних потреб будівлі. Ця система моніторингу забезпечує збір

даних у реальному часі та дозволяє проводити оперативний аналіз енергоспоживання.

Порівняльний аналіз розроблених моделей показав, що комбінований підхід, який використовує ансамбль різних алгоритмів, забезпечує найвищу точність прогнозування. Середня абсолютна похибка (MAE) складає менше 5% для короткострокових прогнозів та не перевищує 8% для довгострокових. Коефіцієнт детермінації (R^2) для найкращої моделі досягає 0.92, що свідчить про високу якість прогнозування.

Застосування методу Time Series Data Forecasting дозволило врахувати сезонні коливання та тренди у споживанні енергії. Аналіз показав, що найбільші коливання енергоспоживання спостерігаються у зимовий період, що корелює з потребами опалення, та у робочі години, коли активність у будівлі максимальна.

Розроблена система має значний потенціал для практичного застосування в управлінні енергоресурсами будівель. Точне прогнозування дозволяє оптимізувати роботу систем опалення, вентиляції та кондиціонування повітря, що може призвести до економії енергії до 15-20%. Система також може використовуватися для планування закупівель енергоресурсів та оптимізації енергетичних контрактів.

Впровадження розробленої системи у навчальних закладах може сприяти формуванню культури енергозбереження та підвищенню обізнаності щодо раціонального використання енергоресурсів. Крім того, система може служити навчальним інструментом для студентів технічних спеціальностей.

У ході дослідження було виявлено деякі обмеження розробленої системи. Точність прогнозування значно знижується при наявності аномальних подій, таких як аварії обладнання або незвичайні погодні умови. Також система потребує регулярного оновлення та калібрування для підтримання високої точності прогнозів.

Перспективними напрямками подальшого розвитку є інтеграція додаткових джерел даних, таких як прогнози погоди, календарні події та дані про заповненість будівлі. Використання методів глибокого навчання, зокрема

рекурентних нейронних мереж (LSTM), може підвищити точність довгострокових прогнозів.

Розроблена система сприяє досягненню цілей сталого розвитку через зменшення споживання енергії та відповідно зниження викидів парникових газів. Ефективне управління енергоспоживанням будівель є важливим кроком у напрямку створення розумних міст та досягнення вуглецевої нейтральності.

Проведене дослідження підтвердило ефективність застосування методів машинного навчання для прогнозування енергоспоживання будівель. Розроблена система демонструє високу точність прогнозування та має значний потенціал для практичного застосування. Результати роботи можуть бути використані для підвищення енергоефективності будівель, зменшення експлуатаційних витрат та сприяння сталому розвитку.

Успішна реалізація проекту відкриває можливості для подальших досліджень у галузі енергетичної аналітики та розвитку інтелектуальних систем управління енергоспоживанням. Отримані результати мають практичну цінність для енергетичних компаній, управляючих компаній та навчальних закладів, які прагнуть оптимізувати своє енергоспоживання та знизити експлуатаційні витрати.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. International Energy Agency (IEA). (2024). *Electricity 2024 - Analysis*. IEA Reports. URL: <https://www.iea.org/reports/electricity-2024/executive-summary>
2. Enerdata. (2024). *World Power consumption | Electricity consumption*. URL: <https://yearbook.enerdata.net/electricity/electricity-domestic-consumption-data.html>
3. National Renewable Energy Laboratory (NREL). (2023). *NREL Researchers Reveal How Buildings Across United States Do—and Could—Use Energy*. URL: <https://surli.cc/nyylsf>
4. U.S. Department of Energy. *Data and Analysis for Buildings Sector Innovation*. URL: <https://www.energy.gov/eere/buildings/data-and-analysis-buildings-sector-innovation>
5. European Commission. *Energy Performance of Buildings Directive*. URL: <https://surli.cc/wchfne>
6. European Commission. (2020). *In focus: Energy efficiency in buildings*. URL: <https://surl.lu/lzwwbc>
7. MarketsandMarkets. (2024). *Energy Management Systems Market worth \$84.34 billion by 2029*. URL: <https://surl.lt/nclylz>
8. Statista. (2024). *Ukraine: power production share by source 2023*. URL: <https://surl.li/luwsua>
9. Grand View Research. (2024). *Smart Building Market Size, Share And Growth Report, 2030*. URL: <https://surl.lu/pqamrg>
10. Transforma Insights. (2022). *The use of smart buildings solutions can reduce total global energy consumption by 3-5%*. URL: <https://surl.li/acewfo>
11. Facilio. (2024). *What Is Building Energy Management System & Why It's Crucial?* URL: <https://facilio.com/blog/building-energy-management-system/>
12. International Energy Agency (IEA). (2025). *Growth in global energy demand surged in 2024 to almost twice its recent average*. IEA News, March 24, 2025. URL: <https://surl.li/sdkyfv>
13. International Energy Agency (IEA). (2024). *Electricity 2024 - Executive Summary*. URL: <https://www.iea.org/reports/electricity-2024/executive-summary>
14. Reuters. (2024). *US electricity demand from EVs jumps to new highs in early 2024*. URL: <https://surl.li/iyyvwe>
15. UkraineInvest. (2024). *Відновлювана енергетика*. Державне агентство з питань інвестицій та розвитку України. URL: <https://surl.lu/gxsiiz>
16. Рубрика. (2023). *Українська енергосистема 2023-2024: проблеми, виклики та перспективи*. URL: <https://rubryka.com/blog/ukrayinska-energostema/>

17. Білоус, О. В., Кузнецов, Р. П. Гібридні моделі прогнозування енергоспоживання з використанням машинного навчання. Наукові вісті НТУУ «КПІ». 2021. № 4. С. 45–52.
18. Бондаренко, В. С., Петров, А. М. Прогнозування споживання електричної енергії електротехнічних комплексів міської електричної мережі. Вісник НТУ «ХПІ». 2022. № 8. С. 45–52.
19. Василенко, Т. М. Глибоке навчання у прогнозуванні енергетичних параметрів промислових підприємств. Вісник Хмельницького національного університету. Технічні науки. 2020. № 5. С. 123–129. URL: <https://surl.li/znqojg>
20. Використання штучного інтелекту в енергетиці України: перспективи та виклики : звіт за результатами дослідження / Інститут економіки та прогнозування НАН України. Київ, 2020. 124 с.
21. Гончаренко, В. М. Прогнозування пікових навантажень в електроенергетичних системах з використанням нейронних мереж. Проблеми загальної енергетики. 2020. № 3. С. 23–30.
22. Григоренко, Д. О. Використання LSTM-мереж для прогнозування енергоспоживання промислових об'єктів. Вісник НТУУ «КПІ». Серія «Інформатика, управління та обчислювальна техніка». 2020. № 73. С. 89–96.
23. Гриценко, П. М. Застосування Random Forest для прогнозування споживання природного газу. Нафтова і газова промисловість. 2020. № 4. С. 67–73.
24. Гриценко, П. М. Аналіз ефективності алгоритмів машинного навчання в енергетичних додатках. Збірник наукових праць Донецького національного технічного університету. Серія «Електротехніка і енергетика». 2021. Вип. 21. С. 45–51.
25. ДСТУ 4472:2018. Енергоефективність будівель. Методи розрахунку енергоспоживання при опаленні, охолодженні, вентиляції, освітленні та гарячому водопостачанні. Київ : ДП «УкрНДНЦ», 2018. 156 с.
26. ДСТУ ISO 50001:2020. Системи енергетичного менеджменту. Вимоги та настанови щодо використання. Київ : ДП «УкрНДНЦ», 2020. 28 с.
27. Енергетичні системи та прогнозування: метод. рек./В. П. Сакун. Київ: НАУ, 2020. 78 с.
28. Іваненко, Т. П. Аналіз ефективності алгоритмів машинного навчання в задачах енергетичного менеджменту. Проблеми загальної енергетики. 2020. Вип. 4. С. 12–19.
29. Калашник, В. Р. Машинне навчання в енергетиці: методи та застосування. Електронний архів НТУ «ХПІ». 2021. URL: <https://surli.cc/odyrzg>
30. Козлов, М. І., Сидоренко, О. В. Використання методів машинного навчання для

- прогнозування енергоспоживання будівель. Енергетика та автоматика. 2021. № 3. С. 78–85.
31. Костенко, М. Ю., Борисенко, С. І. Розробка системи прогнозування енергоспоживання на основі Support Vector Machines. Вісник Вінницького політехнічного інституту. 2021. № 2. С. 89–96.
32. Кравець, В. В. Порівняльний аналіз алгоритмів машинного навчання для прогнозування навантаження в електроенергетичних системах. Відновлювана енергетика. 2020. № 2. С. 56–63.
33. Левченко, О. Р. Використання ансамблевих методів машинного навчання для прогнозування енергетичних параметрів. Енергетика та автоматика. 2020. № 6. С. 123–130.
34. Лисенко, А. Б. Використання штучних нейронних мереж для прогнозування споживання енергії в житлових будинках. Енергетика та енергозбереження : матеріали XIV Міжнар. наук.-практ. конф. Київ : НТУУ «КП», 2021. С. 145–148.
35. Машинне навчання в енергетичних системах : підручник / О. П. Коваленко, М. С. Іванов. Харків : НТУ «ХП», 2020. 312 с.
36. Мартиненко, Г. Ю. Оптимізація параметрів машинного навчання для задач енергетичного прогнозування. Збірник наукових праць Харківського університету Повітряних Сил. 2021. № 2. С. 234–241.
37. Панченко, А. В. Використання машинного навчання у задачах класифікації енергетичних об'єктів. Наукові записки НаУКМА. Комп'ютерні науки. 2021. Т. 4. С. 67–74. URL: <http://nrpcomp.ukma.edu.ua/article/view/185511>
38. Романенко, К. С., Дмитренко, В. А. Інтелектуальні системи управління енергоспоживанням на основі машинного навчання. Інформаційні технології в енергетиці : матеріали XI Всеукр. наук.-практ. конф. Львів : НУ «Львівська політехніка», 2021. С. 234–237.
39. Семенов, Д. К., Волков, О. П. Застосування методів машинного навчання для оптимізації роботи відновлюваних джерел енергії. Відновлювана енергетика. 2021. № 1. С. 78–85.

ДОДАТКИ

Реалізація моделі XGBoost для прогнозування енергоспоживання

```

def train_xgboost(self, X_train, X_test, y_train, y_test):
    """
    Повна реалізація навчання XGBoost моделі з оптимізованими параметрами
    """
    print("🌀 Навчання XGBoost моделі...")
    # Налаштування оптимальних гіперпараметрів
    model = xgb.XGBRegressor(
        n_estimators=200,      # Кількість дерев
        max_depth=6,          # Максимальна глибина дерева
        learning_rate=0.1,    # Швидкість навчання
        subsample=0.8,        # Частка вибірки для навчання кожного дерева
        colsample_bytree=0.8,  # Частка ознак для кожного дерева
        reg_alpha=0.1,        # L1 регуляризація
        reg_lambda=1.0,       # L2 регуляризація
        random_state=42,      # Фіксація випадковості
        n_jobs=-1             # Використання всіх процесорів
    )
    # Навчання моделі
    model.fit(
        X_train, y_train,
        eval_set=[(X_test, y_test)],
        early_stopping_rounds=10,
        verbose=False
    )

    # Генерація прогнозів
    predictions = model.predict(X_test)
    # Розрахунок метрик якості
    mae = mean_absolute_error(y_test, predictions)
    rmse = np.sqrt(mean_squared_error(y_test, predictions))
    r2 = r2_score(y_test, predictions)
    mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100
    # Аналіз важливості ознак
    feature_importance = model.feature_importances_
    # Збереження результатів
    self.models['xgboost'] = model
    self.results['xgboost'] = {
        'predictions': predictions,
        'mae': mae,
        'rmse': rmse,
        'r2': r2,
        'mape': mape,
        'feature_importance': feature_importance
    }

```

```

print(f"XGBoost - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R2: {r2:.3f}, MAPE:
{mape:.2f}%")
return predictions
# Додаткова функція для аналізу важливості ознак XGBoost
def plot_xgboost_importance(self):
    """
    Візуалізація важливості ознак для XGBoost моделі
    """
    if 'xgboost' not in self.results:
        return
    importance = self.results['xgboost']['feature_importance']
    feature_names = [f'Feature_{i}' for i in range(len(importance))]
    # Сортування ознак за важливістю
    indices = np.argsort(importance)[::-1][:10] # Топ 10 ознак
    plt.figure(figsize=(10, 6))
    plt.bar(range(len(indices)), importance[indices])
    plt.title('Важливість ознак (XGBoost)')
    plt.xticks(range(len(indices)), [feature_names[i] for i in indices], rotation=45)
    plt.ylabel('Важливість')
    plt.tight_layout()
    plt.show()

```

Реалізація моделі KNN для прогнозування енергоспоживання

python

```

def train_knn(self, X_train, X_test, y_train, y_test):
    """
    Повна реалізація навчання KNN моделі з оптимізацією параметрів
    """
    print("🌀 Навчання KNN моделі...")
    # Пошук оптимальної кількості сусідів
    best_k = self.optimize_knn_parameters(X_train, y_train)
    # Створення оптимізованої моделі
    model = KNeighborsRegressor(
        n_neighbors=best_k,
        weights='distance', # Зважування за відстанню
        algorithm='auto', # Автоматичний вибір алгоритму
        metric='euclidean', # Евклідова відстань
        n_jobs=-1 # Паралельні обчислення
    )
    # Навчання моделі
    model.fit(X_train, y_train)
    # Генерація прогнозів
    predictions = model.predict(X_test)
    # Розрахунок метрик якості
    mae = mean_absolute_error(y_test, predictions)
    rmse = np.sqrt(mean_squared_error(y_test, predictions))
    r2 = r2_score(y_test, predictions)
    mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100
    # Збереження результатів
    self.models['knn'] = model
    self.results['knn'] = {
        'predictions': predictions,
        'mae': mae,
        'rmse': rmse,
        'r2': r2,
        'mape': mape,
        'optimal_k': best_k
    }
    print(f"KNN (k={best_k}) - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R²: {r2:.3f}, MAPE: {mape:.2f}%")
    return predictions
def optimize_knn_parameters(self, X_train, y_train):
    """
    Оптимізація кількості сусідів для KNN моделі за допомогою крос-валідації
    """
    from sklearn.model_selection import cross_val_score
    k_values = range(3, 21, 2) # Тестування непарних значень від 3 до 19
    cv_scores = {}

```

```

for k in k_values:
    model = KNeighborsRegressor(n_neighbors=k, weights='distance')
    scores = cross_val_score(model, X_train, y_train, cv=5,
scoring='neg_mean_absolute_error')
    cv_scores[k] = -scores.mean()

# Вибір k з найменшою похибкою
best_k = min(cv_scores.keys(), key=lambda x: cv_scores[x])
print(f"Оптимальна кількість сусідів: {best_k}")
return best_k
# Додаткова функція для аналізу впливу параметрів KNN
def plot_knn_optimization(self, X_train, y_train):
    """
    Візуалізація процесу оптимізації параметрів KNN
    """
    k_values = range(3, 21, 2)
    train_errors = []
    val_errors = []
    for k in k_values:
        model = KNeighborsRegressor(n_neighbors=k, weights='distance')
        model.fit(X_train, y_train)
        # Помилка на навчальній вибірці
        train_pred = model.predict(X_train)
        train_error = mean_absolute_error(y_train, train_pred)
        train_errors.append(train_error)
        # Помилка на валідації (крос-валідація)
        val_scores = cross_val_score(model, X_train, y_train, cv=5,
scoring='neg_mean_absolute_error')
        val_errors.append(-val_scores.mean())
    plt.figure(figsize=(10, 6))
    plt.plot(k_values, train_errors, 'o-', label='Навчальна помилка')
    plt.plot(k_values, val_errors, 's-', label='Валідаційна помилка')
    plt.xlabel('Кількість сусідів (k)')
    plt.ylabel('Середня абсолютна похибка')
    plt.title('Оптимізація параметрів KNN')
    plt.legend()
    plt.grid(True, alpha=0.3)
    plt.show()

```

Реалізація моделі LSTM для прогнозування енергоспоживання Python

```
def train_lstm(self, target='electricity_kwh', sequence_length=24):
    """
    Повна реалізація навчання LSTM моделі з оптимізованою архітектурою
    """
    if not TF_AVAILABLE:
        print(" ⚠ TensorFlow недоступний, LSTM пропущено")
        return None
    print(" 🌀 Навчання LSTM моделі...")
    # Підготовка даних для LSTM
    feature_cols = [col for col in self.data.columns
                    if col not in ['timestamp', 'electricity_kwh', 'gas_m3', 'water_m3']]
    # Нормалізація даних
    scaler_X = MinMaxScaler(feature_range=(0, 1))
    scaler_y = MinMaxScaler(feature_range=(0, 1))
    X_scaled = scaler_X.fit_transform(self.data[feature_cols])
    y_scaled = scaler_y.fit_transform(self.data[[target]])
    # Створення послідовностей для LSTM
    X_sequences, y_sequences = self.create_sequences(X_scaled, y_scaled, sequence_length)

    # Розділення на навчальну та тестову вибірки
    split_idx = int(len(X_sequences) * 0.8)
    X_train, X_test = X_sequences[:split_idx], X_sequences[split_idx:]
    y_train, y_test = y_sequences[:split_idx], y_sequences[split_idx:]
    # Побудова оптимізованої LSTM архітектури
    model = Sequential([
        # Перший LSTM шар з return_sequences=True для стекування
        LSTM(128, return_sequences=True, input_shape=(sequence_length,
X_train.shape[2])),
        Dropout(0.3),
        # Другий LSTM шар
        LSTM(64, return_sequences=False),
        Dropout(0.3),
        # Повно-зв'язані шари
        Dense(50, activation='relu'),
        Dropout(0.2),
        Dense(25, activation='relu'),
        Dense(1)
    ])
    # Компіляція моделі з адаптивним оптимізатором
    model.compile(
        optimizer=Adam(learning_rate=0.001, beta_1=0.9, beta_2=0.999),
        loss='huber', # Більш стійка до викидів функція втрат
        metrics=['mae']
    )
)
```

```

# Налаштування callbacks
early_stopping = tf.keras.callbacks.EarlyStopping(
    monitor='val_loss',
    patience=15,
    restore_best_weights=True
)
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(
    monitor='val_loss',
    factor=0.5,
    patience=10,
    min_lr=1e-7
)
# Навчання моделі
history = model.fit(
    X_train, y_train,
    batch_size=32,
    epochs=100,
    validation_data=(X_test, y_test),
    callbacks=[early_stopping, reduce_lr],
    verbose=1
)
# Генерація прогнозів
predictions_scaled = model.predict(X_test)
predictions = scaler_y.inverse_transform(predictions_scaled)
y_test_original = scaler_y.inverse_transform(y_test)

# Розрахунок метрик якості
mae = mean_absolute_error(y_test_original, predictions)
rmse = np.sqrt(mean_squared_error(y_test_original, predictions))
r2 = r2_score(y_test_original, predictions)
mape = np.mean(np.abs((y_test_original - predictions) / y_test_original)) * 100
# Збереження результатів
self.models['lstm'] = model
self.results['lstm'] = {
    'predictions': predictions.flatten(),
    'y_test': y_test_original.flatten(),
    'mae': mae,
    'rmse': rmse,
    'r2': r2,
    'mape': mape,
    'history': history.history,
    'scalers': {'X': scaler_X, 'y': scaler_y}
}
print(f"LSTM - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R2: {r2:.3f}, MAPE:
{mape:.2f}%")
return predictions
def create_sequences(self, X_scaled, y_scaled, sequence_length):
    """
    Створення послідовностей даних для навчання LSTM

```

```

"""
X_sequences, y_sequences = [], []
for i in range(sequence_length, len(X_scaled)):
    X_sequences.append(X_scaled[i-sequence_length:i])
    y_sequences.append(y_scaled[i])
return np.array(X_sequences), np.array(y_sequences)
def plot_lstm_training_history(self):
    """
    Візуалізація процесу навчання LSTM моделі
    """
    if 'lstm' not in self.results:
        return
    history = self.results['lstm']['history']
    plt.figure(figsize=(15, 5))
    # Графік функції втрат
    plt.subplot(1, 3, 1)
    plt.plot(history['loss'], label='Навчальна втрата')
    plt.plot(history['val_loss'], label='Валідаційна втрата')
    plt.title('Функція втрат під час навчання')
    plt.xlabel('Епоха')
    plt.ylabel('Втрата')
    plt.legend()
    plt.grid(True, alpha=0.3)
    # Графік MAE
    plt.subplot(1, 3, 2)
    plt.plot(history['mae'], label='Навчальна MAE')
    plt.plot(history['val_mae'], label='Валідаційна MAE')
    plt.title('Середня абсолютна похибка')
    plt.xlabel('Епоха')
    plt.ylabel('MAE')
    plt.legend()
    plt.grid(True, alpha=0.3)
    # Графік прогнозів vs реальних значень
    plt.subplot(1, 3, 3)
    y_test = self.results['lstm']['y_test']
    predictions = self.results['lstm']['predictions']
    plt.scatter(y_test, predictions, alpha=0.5)
    plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--', lw=2)
    plt.xlabel('Реальні значення')
    plt.ylabel('Прогнозовані значення')
    plt.title('Прогнози vs Реальні значення')
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.show()

```

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime, timedelta
import warnings
warnings.filterwarnings('ignore')
# Бібліотеки машинного навчання
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import xgboost as xgb
# TensorFlow для LSTM
try:
    import tensorflow as tf
    from tensorflow.keras.models import Sequential
    from tensorflow.keras.layers import LSTM, Dense, Dropout
    from tensorflow.keras.optimizers import Adam
    TF_AVAILABLE = True
except ImportError:
    print("⚠ TensorFlow не встановлено. LSTM модель буде пропущена.")
    TF_AVAILABLE = False
class EnergyPredictionSystem:
    """
    Спрощена система прогнозування споживання енергії
    """

    def __init__(self):
        self.data = None
        self.models = {}
        self.results = {}
    def generate_data(self, hours=2000):
        """
        Генерація реалістичних даних споживання енергії
        """
        print("📊 Генерація даних споживання енергії...")
        # Створення часового ряду
        start_date = datetime(2023, 1, 1)
        timestamps = [start_date + timedelta(hours=i) for i in range(hours)]
        np.random.seed(42)
        data = []
        for i, ts in enumerate(timestamps):
            # Моделювання реалістичних паттернів споживання
            hour = ts.hour
            day_of_week = ts.weekday()
            month = ts.month

```

```

# Базове електроспоживання з денною та сезонною варіацією
base_electricity = 150
# Денний цикл (більше вдень, менше вночі)
base_electricity += 50 * np.sin(2 * np.pi * hour / 24 + np.pi/2)
# Сезонний цикл (більше взимку та влітку)
base_electricity += 30 * (np.sin(2 * np.pi * month / 12) + 1)
# Різниця між буднями та вихідними
if day_of_week >= 5: # вихідні
    base_electricity *= 0.8
# Температура з сезонними коливаннями
temperature = 15 + 10 * np.sin(2 * np.pi * month / 12) + np.random.normal(0, 3)
# Вплив температури на електроспоживання (кондиціонери/опалення)
temp_effect = 1 + 0.03 * abs(temperature - 20)
electricity = base_electricity * temp_effect + np.random.normal(0, 15)
# Газ (більше взимку для опалення)
gas_base = 40 + 30 * (1 - np.sin(2 * np.pi * month / 12 + np.pi/2))
gas = gas_base + np.random.normal(0, 8)
# Вода (відносно стабільне споживання з невеликими варіаціями)
water = 80 + 15 * np.sin(2 * np.pi * day_of_week / 7) + np.random.normal(0, 10)
data.append({
    'timestamp': ts,
    'electricity_kwh': max(0, electricity),
    'gas_m3': max(0, gas),
    'water_m3': max(0, water),
    'temperature': temperature,
    'hour': hour,
    'day_of_week': day_of_week,
    'month': month,
    'is_weekend': 1 if day_of_week >= 5 else 0
})
self.data = pd.DataFrame(data)
print(f"✅ Згенеровано {len(self.data)} записів")
return self.data
def create_features(self):
    """
    Створення ознак для машинного навчання
    """
    print("🔗 Створення ознак...")
    # Циклічні ознаки для часових даних
    self.data['hour_sin'] = np.sin(2 * np.pi * self.data['hour'] / 24)
    self.data['hour_cos'] = np.cos(2 * np.pi * self.data['hour'] / 24)
    self.data['day_sin'] = np.sin(2 * np.pi * self.data['day_of_week'] / 7)
    self.data['day_cos'] = np.cos(2 * np.pi * self.data['day_of_week'] / 7)
    self.data['month_sin'] = np.sin(2 * np.pi * self.data['month'] / 12)
    self.data['month_cos'] = np.cos(2 * np.pi * self.data['month'] / 12)
    # Лагові ознаки (попередні значення)
    for lag in [1, 2, 6, 24]:
        self.data[f'electricity_lag_{lag}'] = self.data['electricity_kwh'].shift(lag)
        self.data[f'gas_lag_{lag}'] = self.data['gas_m3'].shift(lag)

```

```

    self.data[f'water_lag_{lag}'] = self.data['water_m3'].shift(lag)
    # Скользящі середні
    for window in [6, 24]:
        self.data[f'electricity_ma_{window}'] =
self.data['electricity_kwh'].rolling(window).mean()
        self.data[f'temperature_ma_{window}'] =
self.data['temperature'].rolling(window).mean()
    # Додаткові температурні ознаки
    self.data['temp_squared'] = self.data['temperature'] ** 2
    self.data['temp_deviation'] = abs(self.data['temperature'] - 20)
    # Видалення рядків з пропущеними значеннями
    self.data = self.data.dropna()
    print(f"✅ Підготовлено {len(self.data)} записів з ознаками")
def prepare_data(self, target='electricity_kwh'):
    """
    Підготовка даних для навчання
    """
    # Вибір ознак
    feature_cols = [col for col in self.data.columns
                    if col not in ['timestamp', 'electricity_kwh', 'gas_m3', 'water_m3']]
    X = self.data[feature_cols].values
    y = self.data[target].values
    # Розділення на навчальну та тестову вибірки
    split_idx = int(len(X) * 0.8)
    X_train, X_test = X[:split_idx], X[split_idx:]
    y_train, y_test = y[:split_idx], y[split_idx:]
    # Нормалізація даних
    scaler_X = StandardScaler()
    scaler_y = StandardScaler()
    X_train_scaled = scaler_X.fit_transform(X_train)
    X_test_scaled = scaler_X.transform(X_test)
    y_train_scaled = scaler_y.fit_transform(y_train.reshape(-1, 1)).ravel()
    y_test_scaled = scaler_y.transform(y_test.reshape(-1, 1)).ravel()
    return (X_train_scaled, X_test_scaled, y_train_scaled, y_test_scaled,
            scaler_X, scaler_y, y_train, y_test)
def train_xgboost(self, X_train, X_test, y_train, y_test):
    """
    Навчання XGBoost моделі
    """
    print("🌀 Навчання XGBoost моделі...")
    model = xgb.XGBRegressor(
        n_estimators=200,
        max_depth=6,
        learning_rate=0.1,
        subsample=0.8,
        colsample_bytree=0.8,
        random_state=42
    )
    model.fit(X_train, y_train)

```

```

predictions = model.predict(X_test)
# Оцінка якості
mae = mean_absolute_error(y_test, predictions)
rmse = np.sqrt(mean_squared_error(y_test, predictions))
r2 = r2_score(y_test, predictions)
mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100
self.models['xgboost'] = model
self.results['xgboost'] = {
    'predictions': predictions,
    'mae': mae,
    'rmse': rmse,
    'r2': r2,
    'mape': mape
}
print(f"XGBoost - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R2: {r2:.3f}, MAPE:
{mape:.2f}%")
return predictions
def train_knn(self, X_train, X_test, y_train, y_test):
    """
    Навчання KNN моделі (weighted regression)
    """
    print("🌀 Навчання KNN моделі...")
    model = KNeighborsRegressor(n_neighbors=5, weights='distance')
    model.fit(X_train, y_train)
    predictions = model.predict(X_test)
    # Оцінка якості
    mae = mean_absolute_error(y_test, predictions)
    rmse = np.sqrt(mean_squared_error(y_test, predictions))
    r2 = r2_score(y_test, predictions)
    mape = np.mean(np.abs((y_test - predictions) / y_test)) * 100
    self.models['knn'] = model
    self.results['knn'] = {
        'predictions': predictions,
        'mae': mae,
        'rmse': rmse,
        'r2': r2,
        'mape': mape
    }
    print(f"KNN - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R2: {r2:.3f}, MAPE:
{mape:.2f}%")
    return predictions
def train_lstm(self, target='electricity_kwh', sequence_length=24):
    """
    Навчання LSTM моделі
    """
    if not TF_AVAILABLE:
        print("⚠️ TensorFlow недоступний, LSTM пропущено")
        return None
    print("🌀 Навчання LSTM моделі...")

```

```

# Підготовка послідовностей для LSTM
feature_cols = [col for col in self.data.columns
                 if col not in ['timestamp', 'electricity_kwh', 'gas_m3', 'water_m3']]
# Нормалізація даних
scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()
X_scaled = scaler_X.fit_transform(self.data[feature_cols])
y_scaled = scaler_y.fit_transform(self.data[[target]])
# Створення послідовностей
X_sequences, y_sequences = [], []
for i in range(sequence_length, len(X_scaled)):
    X_sequences.append(X_scaled[i-sequence_length:i])
    y_sequences.append(y_scaled[i])
X_sequences = np.array(X_sequences)
y_sequences = np.array(y_sequences)
# Розділення даних
split_idx = int(len(X_sequences) * 0.8)
X_train, X_test = X_sequences[:split_idx], X_sequences[split_idx:]
y_train, y_test = y_sequences[:split_idx], y_sequences[split_idx:]
# Побудова LSTM моделі
model = Sequential([
    LSTM(64, input_shape=(sequence_length, X_train.shape[2])),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(1)
])
model.compile(optimizer=Adam(learning_rate=0.001), loss='mse')
# Навчання
history = model.fit(
    X_train, y_train,
    batch_size=32,
    epochs=50,
    validation_split=0.2,
    verbose=0
)
# Прогнозування
predictions_scaled = model.predict(X_test)
predictions = scaler_y.inverse_transform(predictions_scaled)
y_test_original = scaler_y.inverse_transform(y_test)
# Оцінка якості
mae = mean_absolute_error(y_test_original, predictions)
rmse = np.sqrt(mean_squared_error(y_test_original, predictions))
r2 = r2_score(y_test_original, predictions)
mape = np.mean(np.abs((y_test_original - predictions) / y_test_original)) * 100
self.models['lstm'] = model
self.results['lstm'] = {
    'predictions': predictions.flatten(),
    'y_test': y_test_original.flatten(),

```

```

    'mae': mae,
    'rmse': rmse,
    'r2': r2,
    'mape': mape
}

print(f"LSTM - MAE: {mae:.2f}, RMSE: {rmse:.2f}, R2: {r2:.3f}, MAPE:
{mape:.2f}%")
return predictions
def visualize_results(self, y_test):
    """
    Візуалізація результатів прогнозування
    """
    print("📊 Створення візуалізацій...")
    plt.figure(figsize=(15, 10))
    # Графік порівняння моделей
    plt.subplot(2, 2, 1)
    n_points = min(200, len(y_test)) # Показуємо останні 200 точок
    plt.plot(range(n_points), y_test[-n_points:], 'b-', label='Реальні значення',
linewidth=2)
    if 'xgboost' in self.results:
        pred = self.results['xgboost']['predictions']
        plt.plot(range(n_points), pred[-n_points:], 'r--', label='XGBoost', alpha=0.8)
    if 'knn' in self.results:
        pred = self.results['knn']['predictions']
        plt.plot(range(n_points), pred[-n_points:], 'g--', label='KNN', alpha=0.8)
    if 'lstm' in self.results:
        pred = self.results['lstm']['predictions']
        plt.plot(range(len(pred)), pred, 'm--', label='LSTM', alpha=0.8)
    plt.title('Порівняння прогнозів моделей')
    plt.xlabel('Час (години)')
    plt.ylabel('Споживання електроенергії (кВт·год)')
    plt.legend()
    plt.grid(True, alpha=0.3)
    # Метрики якості
    plt.subplot(2, 2, 2)
    models = list(self.results.keys())
    mae_values = [self.results[model]['mae'] for model in models]
    bars = plt.bar(models, mae_values, color=['red', 'green', 'purple'][:len(models)])
    plt.title('Середня абсолютна похибка (MAE)')
    plt.ylabel('MAE')
    plt.xticks(rotation=45)
    # Додавання значень на стовпчики
    for bar, value in zip(bars, mae_values):
        plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() + 0.5,
            f'{value:.1f}', ha='center', va='bottom')
    # R2 Score
    plt.subplot(2, 2, 3)
    r2_values = [self.results[model]['r2'] for model in models]

```

```

bars = plt.bar(models, r2_values, color=['red', 'green', 'purple'][:len(models)])
plt.title('Коефіцієнт детермінації (R2)')
plt.ylabel('R2 Score')
plt.xticks(rotation=45)
plt.ylim(0, 1)
for bar, value in zip(bars, r2_values):
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() + 0.01,
             f'{value:.3f}', ha='center', va='bottom')
# MAPE
plt.subplot(2, 2, 4)
mape_values = [self.results[model]['mape'] for model in models]
bars = plt.bar(models, maпе_values, color=['red', 'green', 'purple'][:len(models)])
plt.title('Середня абсолютна відсоткова похибка (MAPE)')
plt.ylabel('MAPE (%)')
plt.xticks(rotation=45)
for bar, value in zip(bars, maпе_values):
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() + 0.2,
             f'{value:.1f}%', ha='center', va='bottom')
plt.tight_layout()
plt.show()
def print_summary(self):
    """
    Виведення підсумкових результатів
    """
    print("\n" + "="*60)
    print("📄 ПІДСУМКОВІ РЕЗУЛЬТАТИ МОДЕЛЕЙ")
    print("="*60)
    for model_name, results in self.results.items():
        print(f"\n💎 {model_name.upper()}:")
        print(f" MAE: {results['mae']:.2f} кВт·год")
        print(f" RMSE: {results['rmse']:.2f} кВт·год")
        print(f" R2: {results['r2']:.3f}")
        print(f" MAPE: {results['mape']:.1f}%")
    # Визначення найкращої моделі
    best_model = min(self.results.keys(), key=lambda x: self.results[x]['mae'])
    print(f"\n🏆 НАЙКРАЩА МОДЕЛЬ: {best_model.upper()}")
    print(f" MAE: {self.results[best_model]['mae']:.2f} кВт·год")
    print("\n" + "="*60)
def main():
    """
    Основна функція запуску системи
    """
    print("🏠 СИСТЕМА ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕНЕРГІЇ")
    print("Лицей")
    print("-" * 50)
    # Ініціалізація системи
    predictor = EnergyPredictionSystem()
    # Генерація та підготовка даних
    predictor.generate_data(hours=2000)

```

```

predictor.create_features()
# Підготовка даних для навчання
(X_train, X_test, y_train_scaled, y_test_scaled,
 scaler_X, scaler_y, y_train, y_test) = predictor.prepare_data()
print(f"\n📊 Розмір навчальної вибірки: {len(X_train)}")
print(f"📊 Розмір тестової вибірки: {len(X_test)}")
print(f"📊 Кількість ознак: {X_train.shape[1]}")
# Навчання моделей
print("\n" + "="*50)
print("🎓 НАВЧАННЯ МОДЕЛЕЙ")
print("="*50)
predictor.train_xgboost(X_train, X_test, y_train, y_test)
predictor.train_knn(X_train, X_test, y_train, y_test)
predictor.train_lstm()
# Візуалізація та результати
predictor.visualize_results(y_test)
predictor.print_summary()
print("\n✅ Аналіз завершено успішно!")
if __name__ == "__main__":
    main()

```