МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ І ПРИКЛАДНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЛУХВЕРЧИК СЕРГІЙ АНДРІЙОВИЧ

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій,
д-р. техн. наук, професор
_____Наталія ВЕСЕЛОВСЬКА
«____»_____ 2025 р.

ОПТИМІЗАЦІЯ ЛОГІСТИЧНИХ ПРОЦЕСІВ НА ОСНОВІ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ВЕЛИКИХ ДАНИХ

Спеціальність 122 Комп’ютерні науки

Кваліфікаційна (магістерська) робота

Науковий керівник:
Надія ПОТАПОВА,
доцент кафедри інформаційних технологій,
к. е. н., доцент
_____
(підпис)

Оцінка: _____ / _____ / _____
(бали/за шкалою ЄКТС/за національною шкалою)

Голова ЕК: _____
(підпис)

Вінниця – 2025

АНОТАЦІЯ

Лухверчик С.А. Оптимізація логістичних процесів на основі інтелектуального аналізу великих даних. Спеціальність 122 «Комп'ютерні науки». Освітня програма Комп'ютерна обробка даних (Data Science). Донецький національний університет імені Василя Стуса, Вінниця, 2025.

У магістерській роботі досліджено методи аналізу великих даних та їх застосування для оптимізації логістичних процесів. Розглянуто кластеризацію точок доставки, прогнозування попиту та побудову маршрутів. Розроблено програмне забезпечення, що автоматично очищує дані, виконує групування, будує аналітичні моделі й формує оптимізовані маршрути.

Магістерська робота складається зі вступу, трьох розділів, висновків і додатків. У вступі обґрунтовується актуальність теми, визначається мета та завдання дослідження. Перший розділ містить огляд сучасних підходів Big Data-аналітики у логістиці. У другому сформовано постановку задачі та наведено моделі кластеризації, прогнозування попиту й оптимізації маршрутів. У третьому розділі розроблено програмне забезпечення, що очищує дані, виконує кластеризацію, будує аналітичні моделі та формує оптимізовані маршрути.

Система є універсальною та придатною до роботи з різними наборами даних без зміни архітектури. Отримані результати підтверджують необхідність використання Big Data у зменшенні витрат і скороченні відстаней доставки.

Ключові слова: Big Data, кластеризація, прогнозування, оптимізація маршрутів, логістика,

81 с., 15 рис., 50 джерел.

ABSTRACT

Lukhverchyk S.A. Optimization of Logistics Processes Based On Big Data Analytics. Specialization 122 “Computer Science.” Educational program Computer Data Processing (Data Science). Vasyl’ Stus Donetsk National University, Vinnytsia, 2025.

The thesis examines big data analysis methods applied to logistics optimization. The study includes delivery-point clustering, demand forecasting, and route optimization. A software system was developed to clean data automatically, cluster logistic points, build analytical models, and generate optimized delivery routes.

The thesis examines big data analysis methods applied to logistics optimization. The first chapter reviews modern Big Data approaches in logistics. The second chapter defines the problem and presents models for clustering, demand forecasting, and route optimization. The third chapter introduces software that cleans data, clusters delivery points, builds analytical models, and generates optimized routes.

The solution is universal and can be used with different datasets without architectural changes. The results show that Big Data analytics helps reduce transportation costs and delivery distances.

Keywords: Big Data, clustering, forecasting, route optimization, logistics,

81 p., 15 fig., 50 sources.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Big Data – великі обсяги різнорідних даних.

ML (Machine Learning) – методи машинного навчання.

GIS – геоінформаційна система.

VRP (Vehicle Routing Problem) – задача маршрутизації транспортних засобів.

K-Means – алгоритм кластеризації за центроїдами.

DBSCAN – кластеризація на основі щільності.

Haversine distance – відстань між координатами на поверхні Землі.

Euclidean distance – прямолінійна відстань між точками.

TMS – система управління транспортом.

WMS – система управління складом.

ERP – система планування ресурсів підприємства.

CSV – текстовий формат табличних даних.

Centroid – центр кластера.

Demand Forecasting – прогнозування попиту.

Rolling Mean – ковзне середнє.

Preprocessing – підготовка та очищення даних.

Outlier – аномальне значення.

Route Optimization – оптимізація маршруту доставки.

NNA (Nearest Neighbor Algorithm) – алгоритм найближчого сусіда.

Naive Routing – ненавчений базовий метод побудови маршруту.

Pipeline – послідовність етапів обробки даних.

Pandas – бібліотека для роботи з даними.

Sklearn – бібліотека аналітичних та кластеризаційних методів.

Matplotlib — бібліотека для побудови графіків.

Dataset — набір даних для аналізу.

ВСТУП

Актуальність теми дослідження. У сучасних умовах цифрової трансформації логістика є одним із ключових елементів забезпечення стабільного функціонування економіки, промисловості та торгівлі. Логістичні системи охоплюють складні процеси планування, управління, транспортування, складування, контролю запасів і розподілу ресурсів, що потребує високої точності, оперативності та адаптивності. Стрімке зростання обсягів даних, які генеруються у логістичних операціях – від GPS-трекінгу транспортних засобів до сенсорних вимірювань на складах, транзакцій клієнтів та даних з інформаційних систем – створює необхідність у впровадженні ефективних методів інтелектуального аналізу інформації [1, 8].

У таких умовах традиційні методи обробки не здатні забезпечити достатню швидкість аналізу та виявлення прихованих закономірностей. Тому на перший план виходять технології Big Data, які поєднують машинне навчання, оптимізаційні алгоритми та інструменти прогнозної аналітики, дозволяючи будувати адаптивні, гнучкі та ефективні логістичні рішення.

Підприємства дедалі більше потребують прозорих, масштабованих та автоматизованих механізмів управління логістичними потоками [9]. Методи інтелектуального аналізу даних дозволяють не тільки описувати поточний стан логістичних процесів, а й прогнозувати майбутні події, мінімізувати витрати, підвищувати точність планування та оптимізувати маршрути доставки. У глобальній економіці такі підходи стають одним з основних факторів конкурентоспроможності.

Метою даної роботи є дослідження та систематизація методів інтелектуального аналізу великих даних для оптимізації логістичних процесів, а також удосконалення методики їх практичного застосування у транспортно-логістичних системах.

В межах даного дослідження були поставлені **наступні завдання:**

1. Проаналізувати сучасні підходи, методи та інструменти Big Data-аналітики у сфері логістики.

2. Дослідити методи кластеризації, прогнозування та оптимізації, придатні для роботи з великими масивами логістичних даних.

3. Визначити критерії ефективності аналітичних моделей у логістичних системах (витрати, час доставки, завантаженість транспорту, точність прогнозів).

4. Розробити узагальнену схему інтеграції аналітичних модулів у логістичну інфраструктуру підприємства;

5. Розробити програмне рішення, здатне виконувати кластеризацію, аналіз попиту та маршрутизацію на основі великих даних.

Об’єкт дослідження: процеси обробки та аналізу великих масивів даних у транспортно-розподільчих логістичних системах.

Предмет дослідження: методи інтелектуального аналізу великих даних та алгоритми оптимізації, що застосовуються для підвищення ефективності логістичних процесів (кластеризація, прогнозування попиту, оптимізація маршрутів).

У процесі дослідження були використані **методи**: аналізу літературних джерел, порівнянь, системного аналізу, машинного навчання, тестування.

Наукова новизна полягає у систематизації та узагальненні підходів до застосування методів Big Data-аналітики у логістиці, які інтегрують кластеризацію, прогнозні моделі та маршрутні оптимізації у єдину аналітичну систему. Набуло удосконалення методичного підходу до створення універсального програмного модуля, що може бути адаптований під різні типи логістичних даних та сценарії їх використання.

Структура магістерської роботи складається зі вступу, трьох розділів, висновків і додатків. У вступі обґрунтовується актуальність теми, визначається

мета та завдання дослідження. Перший розділ містить огляд сучасних підходів Big Data-аналітики у логістиці. У другому сформовано постановку задачі та наведено моделі кластеризації, прогнозування попиту й оптимізації маршрутів. У третьому розділі розроблено програмне забезпечення, що очищує дані, виконує кластеризацію, будує аналітичні моделі та формує оптимізовані маршрути.

Практичне значення дослідження полягає у застосуванні побудованих аналітичних модулів у системах управління логістикою з метою отримання стійкості логістичних мереж до зовнішніх впливів шляхом підвищення точності планування маршрутів та прогнозування попиту, скорочення транспортних витрат та зменшення часу доставки. Розроблене програмне рішення може бути інтегрованим у цифрові логістичні системи для автоматизації аналізу даних та підтримки прийняття управлінських рішень.

Апробація результатів дослідження. Результати даного дослідження було апробовано в доповіді «Аналітичні методи обробки великих даних у логістичних системах» на VI Всеукраїнській науково-практичній конференції «Комп’ютерні технології обробки даних» (м. Вінниця, 5 грудня 2025 року).

РОЗДІЛ 1
ТЕОРЕТИЧНІ ЗАСАДИ ПІДХОДІВ ДО ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ ПРОЦЕСІВ

1.1. Поняття та сучасні підходи до інтелектуального аналізу даних

Поняття «великі дані» (Big Data) виникло на межі XX–XXI століть у зв'язку з лавиноподібним зростанням обсягів цифрової інформації, яку генерують підприємства, споживачі, сенсори, соціальні мережі, мобільні пристрої та інформаційні системи. У науковій літературі термін Big Data описується як сукупність методів, технологій та інструментів для збору, зберігання, обробки й аналізу структурованих, напівструктурованих і неструктурованих даних надвеликих обсягів і швидкостей. Класичне визначення, що закріпилося у 2010-х роках, спирається на концепцію трьох "V": volume (обсяг), velocity (швидкість), variety (різноманітність), до яких сучасні дослідники додають ще veracity (достовірність) та value (цінність) [5].

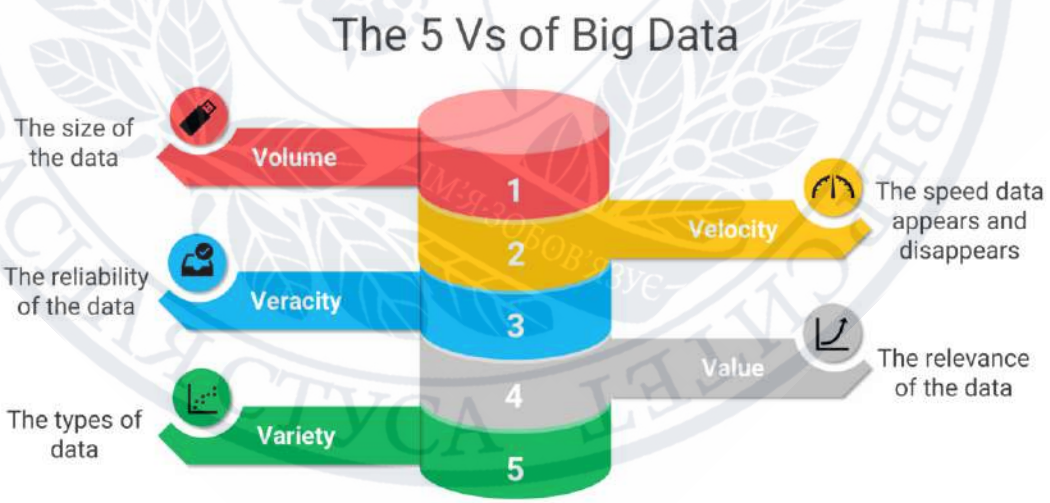

Рисунок 1.1 – Основні характеристики великих даних за моделлю 5V (Volume, Velocity, Variety, Veracity, Value)

У логістичних системах великі дані формуються з численних джерел: телематичних систем транспортних засобів, RFID-міток і сенсорів складів [11],

систем управління ланцюгами постачань (SCM), онлайн-транзакцій, ERP і CRM-рішень, метеорологічних сервісів, геолокаційних модулів тощо. У результаті компанії отримують гетерогенні інформаційні потоки, які потребують синхронізації, очищення та глибокого аналізу для прийняття ефективних управлінських рішень.

Інтелектуальний аналіз даних (Data Mining) – це процес виявлення неочевидних закономірностей, трендів і взаємозв'язків у великих обсягах даних із використанням алгоритмів машинного навчання, статистики, штучного інтелекту та баз даних. Основні етапи Data Mining включають: підготовку даних, відбір ознак, навчання моделей, валідацію результатів та інтерпретацію отриманих знань.

Серед ключових методів інтелектуального аналізу даних у логістиці можна виокремити:

- кластеризацію, яка дозволяє групувати маршрути, клієнтів або склади за схожими характеристиками (наприклад, обсягом перевезень чи географічним розташуванням);
- класифікацію, що використовується для прогнозування категорійних подій (наприклад, виявлення ризикових поставок або відмов у доставці);
- регресійний аналіз, застосовуваний для оцінки впливу факторів на витрати або час доставки;
- асоціативний аналіз, який виявляє кореляції між замовленнями, сезонністю, погодними умовами тощо;
- аналіз часових рядів, який дозволяє передбачати попит, навантаження транспортної мережі чи рівень запасів.

З точки зору логістики, методи інтелектуального аналізу спрямовані на оптимізацію ключових операційних процесів: планування маршрутів, управління запасами, розподіл ресурсів, моніторинг транспортних потоків та контроль постачань. Сукупність цих аспектів формує концепцію «розумної логістики» (Smart Logistics), яка базується на інтеграції інформаційних технологій, IoT-рішень та

аналітики великих даних для отримання обґрунтованих управлінських рішень у режимі реального часу. Таким чином, великі дані в логістичних системах виступають стратегічним ресурсом, а методи їх інтелектуального аналізу – інструментом перетворення цього ресурсу на конкурентну перевагу бізнесу та підвищення ефективності ланцюгів постачань.

У межах такого підходу важливо розуміти, які саме методи й технологічні інструменти застосовуються під час роботи з великими даними. Розвиток Big Data-технологій спричинив формування цілої системи сучасних підходів до обробки й аналізу великих масивів інформації, які умовно можна поділити на традиційні аналітичні, машинно-навчальні та гібридні. Традиційні аналітичні підходи охоплюють статистичні методи, регресійне моделювання, факторний та кореляційний аналіз. Такі методи відзначаються високою інтерпретованістю результатів і придатністю для класичних логістичних задач, проте мають обмеження при роботі з великими обсягами неструктурованих даних або в умовах високої динаміки, характерної для сучасних логістичних середовищ.

Методи машинного навчання (Machine Learning), навпаки, здатні адаптуватися до змін середовища та обробляти великі масиви даних [13]. У логістиці найбільшого поширення набули:

- алгоритми кластеризації (K-means, DBSCAN) для сегментації клієнтів чи маршрутів;
- методи дерев рішень і випадкових лісів (Decision Tree, Random Forest) для прогнозування ризиків або відмов у доставці;
- нейронні мережі (ANN, LSTM, CNN) для обробки часових рядів і просторових даних, зокрема для прогнозування попиту або оптимізації маршрутів;
- методи градієнтного підсилення (XGBoost, LightGBM), які демонструють високу точність при моделюванні багатофакторних залежностей;
- алгоритми підкріплювального навчання (Reinforcement Learning), що дозволяють системі самостійно навчатися на основі досвіду, знаходячи оптимальні

стратегії маршрутизації чи розподілу ресурсів.

У практичних системах логістичної аналітики все частіше застосовуються гібридні моделі, які комбінують кілька підходів для досягнення кращих результатів. Наприклад, поєднання кластеризації з нейронними мережами дозволяє одночасно враховувати географічні особливості маршрутів і динаміку попиту.

У контексті обробки великих обсягів інформації важливе значення має технологічна інфраструктура [12]. Такі платформи, як Hadoop, Spark, Flink або Kafka, забезпечують паралельну обробку потоків даних і можливість роботи в режимі реального часу. Саме завдяки цим інструментам інтелектуальний аналіз даних став невід'ємною частиною сучасних логістичних рішень [4].

Варто відзначити, що останні наукові публікації (зокрема, дослідження European Transport Research Review, IEEE Transactions on Intelligent Transportation Systems та Springer Logistics) фіксують тенденцію переходу від реактивних до прогнозно-аналітичних систем управління. Це означає, що логістичні компанії поступово переходять від простого моніторингу до побудови моделей, які прогнозують проблеми та пропонують оптимальні рішення до їх виникнення.

1.2. Методи збору, зберігання та підготовки даних у логістичних системах

Успішне застосування методів Big Data у логістиці неможливе без налагодженого процесу збору, очищення, трансформації та зберігання даних. Джерела логістичних даних надзвичайно різноманітні: сенсорні системи транспортних засобів, GPS-модулі, системи ERP/CRM, електронні документи, веб-платформи, електронна комерція, платіжні сервіси та навіть соціальні медіа.

Збір даних у логістичних системах здійснюється за допомогою IoT-пристроїв, RFID-технологій, API-інтерфейсів і потокових сервісів (наприклад, Apache Kafka). Ці інструменти забезпечують безперервний потік інформації, який надходить у центральні сховища для подальшої обробки.

Зберігання великих даних зазвичай реалізується у вигляді розподілених систем: Hadoop Distributed File System (HDFS), NoSQL-бази даних (MongoDB, Cassandra, HBase) або хмарні сервіси (AWS S3, Google BigQuery, Azure Synapse). Такі рішення дозволяють масштабувати обчислення, зберігати терабайти інформації та забезпечувати доступ до даних у реальному часі [3].

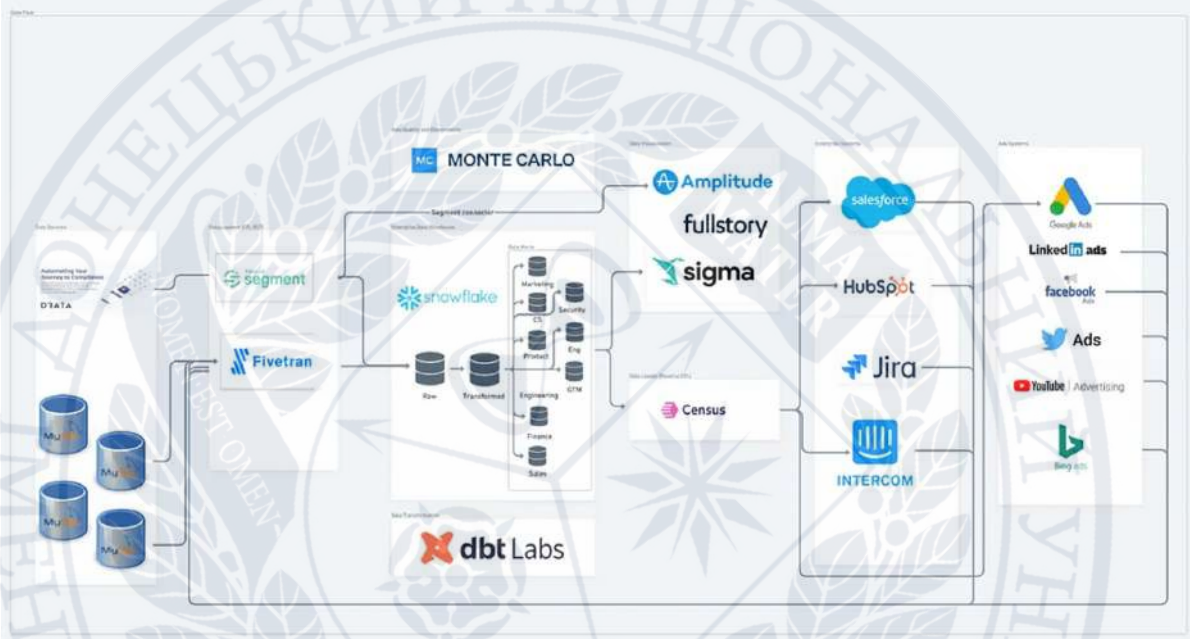

Рисунок 1.2 – Архітектура потокової обробки логістичних даних у реальному часі

Підготовка даних (Data Preprocessing) включає етапи очищення, нормалізації, інтеграції та відбору ознак. Логістичні дані часто містять пропуски, дублікати або похибки, спричинені людським фактором чи технічними збоями. Для їх обробки використовуються методи детекції викидів, заповнення пропусків статистичними або нейромережевими алгоритмами, а також інструменти для автоматичного контролю якості даних.

Окремим напрямом є потокова обробка даних (stream processing), яка дозволяє аналізувати інформацію у режимі реального часу. Це критично важливо для моніторингу транспортних засобів, прогнозування заторів чи оцінки часу

прибуття вантажу. Для цього використовуються фреймворки Apache Spark Streaming, Flink або Storm.

Рисунок 1.3 – Основні фреймворків потокової обробки даних (Spark, Kafka, Storm, Flink, Samza)

Ефективна інфраструктура збору та зберігання даних є базовою умовою для подальшого застосування методів інтелектуального аналізу. Саме від якості та повноти підготовлених даних залежить точність моделей оптимізації логістичних процесів, коректність прогнозів та здатність системи відображати реальні операційні закономірності.

На основі таких підготовлених даних формуються аналітичні моделі, які сьогодні відіграють центральну роль в оптимізації логістичних систем [14], забезпечуючи перехід від інтуїтивного прийняття рішень до системного, даних-орієнтованого управління. У цьому контексті виділяють кілька типів аналітики – описову, діагностичну, прогнозну та оптимізаційну [2], кожна з яких виконує окрему функцію у формуванні ефективної логістичної стратегії.

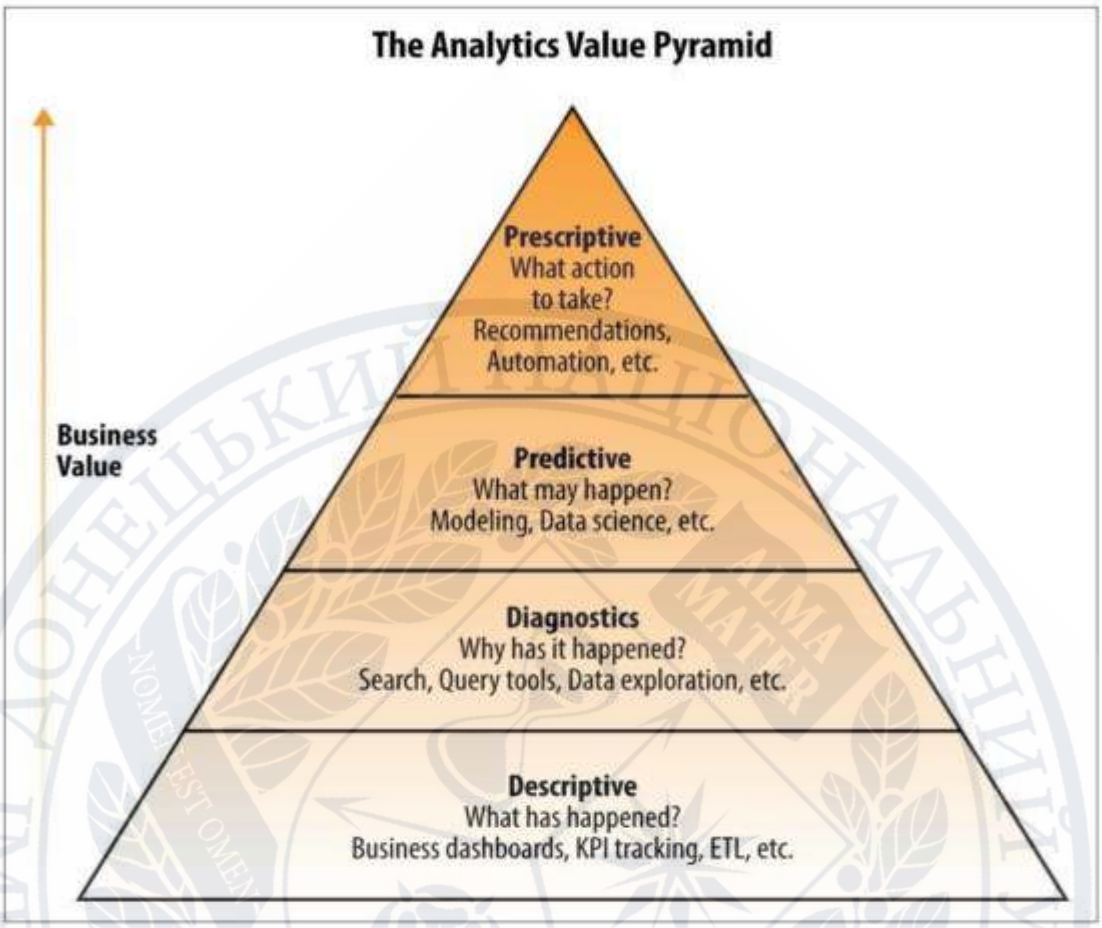

Рисунок 1.4 – Рівні аналітики в логістичних системах (Descriptive, Diagnostic, Predictive, Prescriptive)

Описова аналітика (Descriptive Analytics) використовується для візуалізації поточного стану логістичних процесів: відслідковування місцезнаходження вантажів, аналіз завантаженості складів, визначення продуктивності транспорту.

Діагностична аналітика (Diagnostic Analytics) спрямована на пошук причин відхилень, наприклад, чому певний маршрут має систематичні затримки або збільшені витрати.

Прогнозна аналітика (Predictive Analytics) дозволяє на основі історичних даних передбачати майбутні події: зміни попиту, затори, сезонні коливання, технічні несправності.

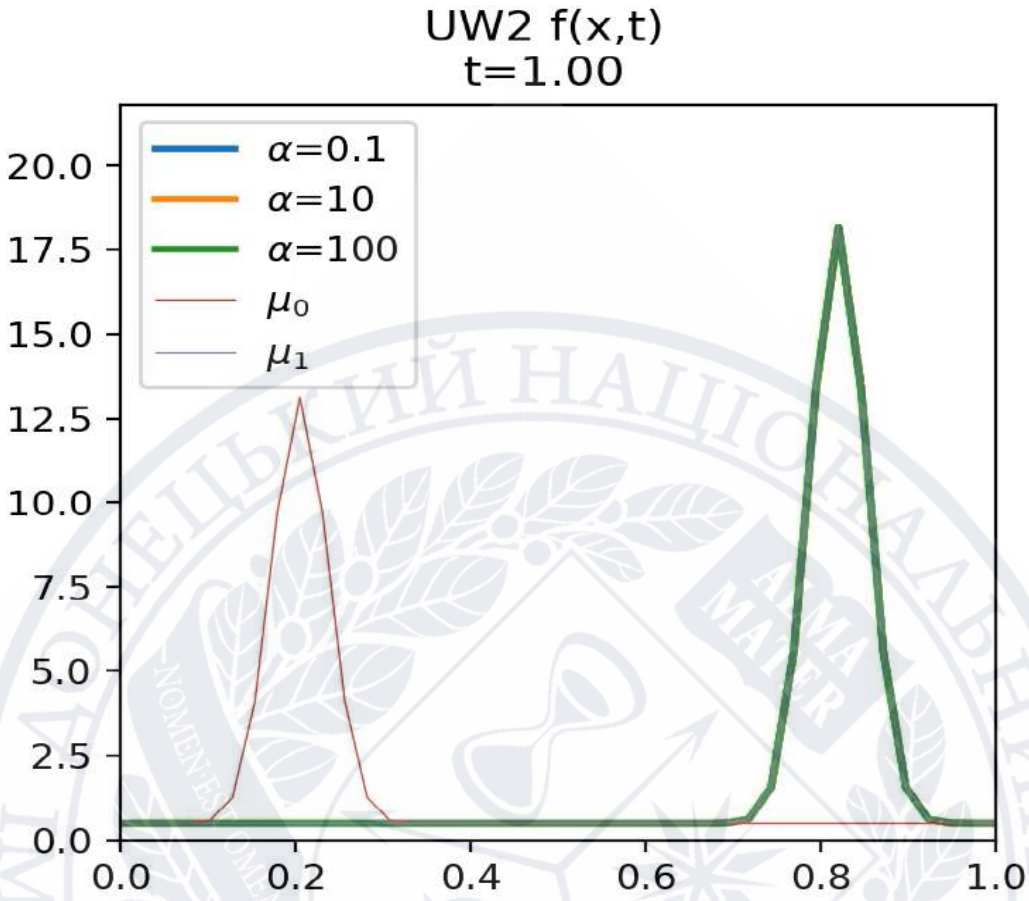

Рисунок 1.5 – Концептуальна схема data-driven екосистеми логістичних процесів

Оптимізаційна аналітика (Prescriptive Analytics) генерує рекомендації щодо найкращих рішень: визначення оптимальних маршрутів, графіків доставки, складання змін для водіїв, управління запасами.

Застосування таких моделей дозволяє скоротити логістичні витрати, підвищити точність прогнозів, оптимізувати використання транспорту та зменшити вуглецевий слід. Наприклад, використання алгоритмів прогнозування попиту дозволяє скоротити надлишкові запаси, тоді як оптимізаційні моделі маршрутизації (Vehicle Routing Problem, VRP) знижують витрати на паливо й час у дорозі.

Впровадження аналітичних моделей у логістичних системах вимагає інтеграції з корпоративними IT-рішеннями: ERP, SCM, CRM, а також із

геоінформаційними системами (GIS). Це створює єдине інформаційне середовище, у якому аналітика перетворюється на основу стратегічного управління.

Таким чином, сучасні логістичні системи стають даними керованими екосистемами (data-driven ecosystems), у яких методи інтелектуального аналізу відіграють вирішальну роль у забезпеченні ефективності, гнучкості та стійкості процесів.

Використання технологій Big Data у логістиці відкриває нові горизонти для підвищення ефективності ланцюгів постачань, оптимізації маршрутів, управління запасами та прогнозування попиту. Основна відмінність логістичних систем від інших сфер застосування великих даних полягає у високій динамічності процесів та неперервності потоків інформації. Кожна операція – від замовлення товару до його доставки – генерує цифровий слід, який можна аналізувати для прийняття рішень у реальному часі.

Завдяки об’єднанню даних із сенсорів транспортних засобів, складських систем, клієнтських платформ і зовнішніх джерел (метеодані, трафік, соціальні мережі) логістичні компанії отримують повну картину свого функціонування. Аналітичні моделі, побудовані на цих даних, дозволяють виявляти закономірності, які раніше були недосяжні для людського аналізу. Застосування Big Data у логістиці сприяє вирішенню таких практичних завдань:

- Оптимізація маршрутів перевезення з урахуванням часу доби, заторів і погодних умов.
- Прогнозування попиту для зменшення надлишкових запасів на складах.
- Оцінка ризиків постачання та виявлення потенційних затримок.
- Персоналізація сервісів для клієнтів шляхом аналізу поведінкових даних.

Таким чином, Big Data стає не лише технічним інструментом, а й стратегічним фактором розвитку логістичних компаній [15], які прагнуть до цифрової трансформації.

Одним із ключових напрямів інтелектуального аналізу логістичних даних є кластеризація – процес автоматичного групування об’єктів (клієнтів, маршрутів, складів, товарів) за певною схожістю. Вона допомагає структурувати складні системи, де велика кількість елементів має спільні властивості. У логістиці це може бути поділ клієнтів за географічним розташуванням, обсягом замовлень чи частотою доставки.

Найпоширеніші методи кластеризації:

- K-Means – для поділу на попередньо задану кількість кластерів;
- DBSCAN – для виявлення груп довільної форми у щільних областях даних;
- Hierarchical Clustering – для побудови ієрархій постачальників чи маршрутів.

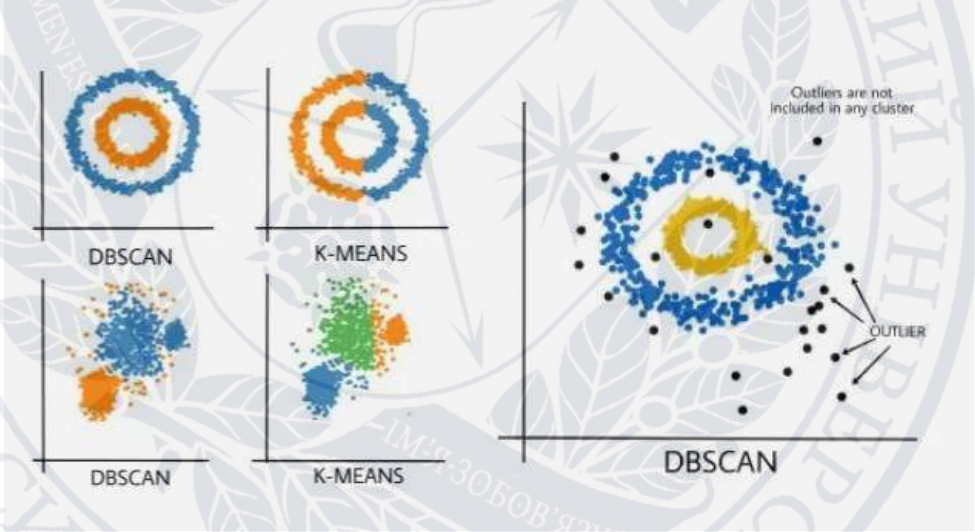

Рисунок 1.6 – Порівняння результатів кластеризації даних методами K-Means та DBSCAN

Наприклад, застосування алгоритму K-Means до історичних даних доставки дозволяє сформувати географічні зони оптимального обслуговування, що зменшує транспортні витрати.

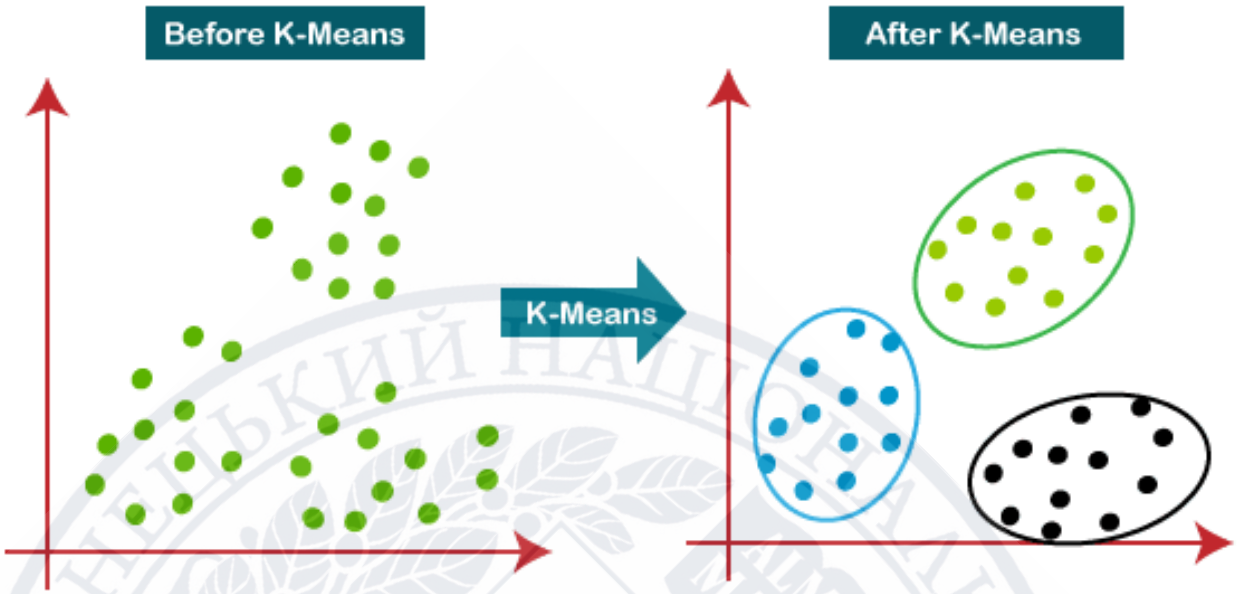

Рисунок 1.7 – Результат кластеризації даних методом K-Means до та після навчання

Інший важливий метод – класифікація, тобто розподіл об’єктів за категоріями на основі навчальних прикладів. У логістиці це може бути прогнозування:

- чи встигне доставка у визначений термін (класи “вчасно / із запізненням”);
- чи є певна партія ризиковою (класи “стандартна / проблемна”);
- які маршрути є оптимальними для конкретного типу товару.

До популярних алгоритмів класифікації належать Decision Tree, Random Forest, Support Vector Machine (SVM), Naive Bayes і Gradient Boosting. Вони дозволяють створювати моделі, що приймають рішення на основі десятків факторів, автоматично виявляючи закономірності між вхідними параметрами та результатами.

Поєднання кластеризації й класифікації створює умови для динамічної оптимізації ланцюгів постачань, де кожен елемент системи аналізується в контексті свого кластеру, а рішення приймаються на основі класифікаційних моделей ризику та ефективності.

1.3. Моделі прогнозування попиту та оптимізації маршрутів

Прогнозування попиту є фундаментальним завданням у логістичному менеджменті. Від точності прогнозу залежить ефективність планування виробництва, обсяги запасів і витрати на транспортування. Методи інтелектуального аналізу даних дозволяють створювати моделі, які враховують як історичні дані, так і зовнішні фактори – сезонність, поведінку споживачів, соціальні тенденції, погодні умови. Найбільш ефективними виявилися такі підходи:

- Регресійні моделі (Linear, Ridge, Lasso) – базові методи прогнозування кількісних показників.
- Моделі часових рядів (ARIMA, SARIMA, Prophet) – враховують сезонність і тренди.
- Нейронні мережі (LSTM, GRU) – здатні виявляти складні нелінійні залежності та враховувати попередній контекст.
- Комбіновані ансамблеві моделі – поєднують кілька методів для підвищення точності прогнозів.

Щодо оптимізації маршрутів, то це завдання описується як Vehicle Routing Problem (VRP) – класична NP-складна задача, у якій потрібно знайти найкоротший або найефективніший шлях для групи транспортних засобів із урахуванням обмежень (часових вікон, ваги вантажу, черговості точок доставки тощо) [7].

Сучасні аналітичні моделі використовують:

- Евристичні алгоритми (генетичні алгоритми, алгоритм мурашиних колоній, метод імітації відпалу);
- Стохастичні моделі для врахування непередбачуваних факторів (погодні умови, затори);
- Методи машинного навчання, які прогнозують оптимальні маршрути на основі історії перевезень.

Використання прогнозних моделей у поєднанні з оптимізаційними дозволяє логістичним компаніям скорочувати операційні витрати, підвищувати швидкість

доставки та зменшувати викиди CO_2 за рахунок оптимального завантаження транспорту.

Технології машинного навчання (Machine Learning, ML) поступово перетворюються на основу сучасного управління логістичними процесами. Їх застосування охоплює практично всі рівні – від планування закупівель до контролю доставки. Зокрема, це:

1. Прогнозне управління (Predictive Supply Chain): ML-моделі аналізують історичні дані й прогнозують можливі збої у постачанні, зміни попиту або порушення графіку доставки.

2. Автоматичне управління запасами: Алгоритми машинного навчання можуть самостійно розраховувати оптимальні рівні запасів на складах з урахуванням коливань попиту та часу поповнення.

3. Динамічне ціноутворення: Застосовується для оптимізації тарифів на доставку залежно від попиту, маршруту, ваги та інших змінних параметрів.

4. Оптимізація складів і транспортних потоків: ML дозволяє створювати моделі, які адаптують маршрути транспорту в реальному часі, враховуючи затори, поломки або зміну умов на дорозі.

Основні алгоритми, що використовуються в управлінні ланцюгами постачань [6]:

- Random Forest, XGBoost – для оцінки ризиків і прогнозування затримок;
- Neural Networks (CNN, LSTM) – для часових і просторових задач;
- Reinforcement Learning (RL) – для навчання системи ухвалювати послідовні рішення з урахуванням відгуку середовища (наприклад, маршрутизація в реальному часі).

Впровадження ML у логістичні процеси формує нову парадигму “розумних ланцюгів постачань” (Smart Supply Chains), у яких системи не лише реагують на події, а й передбачають їх, пропонуючи оптимальні дії. Такі моделі забезпечують здатність логістичних мереж адаптуватися до динамічних змін, підвищують

точність планування та мінімізують вплив зовнішніх ризиків. Проте ефективність цих рішень напряму залежить від якості застосованих методів Big Data-аналітики та їх здатності коректно обробляти різнорідні інформаційні потоки.

У сучасній логістиці ці потоки є надзвичайно інтенсивними та різноманітними – від GPS-сигналів транспортних засобів до даних ERP-систем, сенсорів складів, замовлень клієнтів і транзакційних операцій. Тому для побудови дійсно “розумних” систем недостатньо мати лише інструменти збору та зберігання даних; потрібен глибокий аналітичний аналіз, що дозволяє перетворювати великі масиви інформації на знання, а знання – на ефективні управлінські рішення. Саме порівняння різних підходів Big Data-аналітики дає змогу визначити, які моделі забезпечують найкращий баланс між точністю, швидкістю, інтерпретованістю та масштабованістю у логістичних умовах. Саме тому порівняльна оцінка різних методів інтелектуального аналізу даних є критично важливою для вибору оптимального підходу до оптимізації логістичних процесів.

Першою групою методів є класичні статистичні моделі, що ґрунтуються на лінійних або регресійних залежностях між змінними. Їхня перевага полягає у простоті реалізації, високій інтерпретованості та мінімальних вимогах до обчислювальних ресурсів. Регресійний аналіз, наприклад, широко використовується для прогнозування витрат на транспортування, визначення факторів, що впливають на час доставки, або оцінки економічної ефективності логістичних маршрутів.

Однак головним обмеженням статистичних підходів є їхня нездатність адекватно відображати нелінійні процеси, які характерні для сучасних динамічних систем постачань. Логістика залежить від безлічі взаємопов’язаних факторів – погодних умов, коливань попиту, вартості пального, сезонності, людського фактора – тому моделі, побудовані на припущенні лінійності, швидко втрачають точність при масштабуванні. Таким чином, традиційна аналітика залишається актуальною

лише для базових завдань або в комбінації з більш гнучкими методами машинного навчання.

Другою великою групою є методи кластеризації, які дозволяють структурувати дані та виявляти приховані закономірності. Вони є фундаментом для подальших етапів аналітики – прогнозування, оптимізації чи побудови моделей прийняття рішень. Алгоритми на кшталт K-Means, DBSCAN чи Agglomerative Clustering дають змогу розділяти об’єкти на групи за схожими характеристиками. У логістиці це може означати поділ клієнтів за типом попиту, маршрути – за частотою перевезень, а склади – за рівнем завантаженості.

Головною перевагою кластеризації є здатність працювати без попередньої розмітки даних, що особливо цінно у великих логістичних системах, де навчальні набори не завжди доступні. Однак цей підхід вимагає ретельного підбору кількості кластерів, масштабування ознак і достатнього обсягу якісних даних – інакше модель може створити формальні, але не інтерпретовані угрупування.

На практиці методи кластеризації часто використовуються для побудови сегментованих стратегій управління доставкою. Наприклад, клієнти, згруповані за географічним принципом, можуть обслуговуватись різними логістичними центрами; маршрути, кластеризовані за навантаженням, допомагають рівномірно розподілити транспортні ресурси. Таким чином, кластеризація створює основу для раціоналізації рішень у масштабних системах постачань.

Наступним важливим напрямом є класифікаційні методи, які дозволяють прогнозувати категорійні події. На відміну від кластеризації, класифікація передбачає наявність навчальної вибірки, тобто система вчиться на історичних прикладах, щоб передбачити результат для нових ситуацій. Серед найефективніших алгоритмів можна відзначити Decision Tree, Random Forest, Support Vector Machine (SVM) та Gradient Boosting.

У контексті логістики ці методи використовуються для:

- прогнозування ризику затримки доставки;

- виявлення підозрілих або неефективних маршрутів;
- оцінки надійності постачальників;
- автоматичного визначення пріоритетності замовлень.

Моделі на основі дерев рішень мають перевагу у високій інтерпретованості: менеджер може простежити логіку, за якою система дійшла певного висновку. Алгоритми Random Forest чи Gradient Boosting забезпечують вищу точність за рахунок ансамблевого підходу, коли кілька моделей працюють спільно, компенсуючи похибки одна одної. Проте їхнім недоліком є висока обчислювальна складність і потреба у великій кількості навчальних даних. У масштабних логістичних системах із мільйонами транзакцій такі моделі вимагають потужної інфраструктури Big Data.

Зі зростанням обсягів інформації традиційні методи поступаються місцем глибинним нейронним мережам (Deep Learning), які здатні моделювати складні нелінійні залежності. У логістиці нейронні мережі, зокрема LSTM (Long Short-Term Memory) і GRU (Gated Recurrent Unit), активно застосовуються для прогнозування попиту та оптимізації маршрутів. Їхня ключова перевага – здатність враховувати часову послідовність подій. Наприклад, мережа LSTM може аналізувати, як зміна сезонності чи поведінки споживачів впливає на обсяги замовлень протягом року.

Крім того, глибинні моделі ефективно використовуються для розпізнавання образів і геопросторової аналітики: вони здатні обробляти супутникові знімки для оцінки стану доріг або аналізу транспортної інфраструктури. Попри високу точність прогнозів, ці методи мають і суттєві обмеження: складність налаштування архітектури, потребу у великих навчальних наборах і високе енергоспоживання при тренуванні моделей. Проте у великих логістичних компаніях, де доступні потужні сервери та об’ємні дані, нейронні мережі демонструють безпрецедентну ефективність у підвищенні точності прогнозування та автоматизації операцій.

Окремий клас складають методи підкріплювального навчання (Reinforcement Learning, RL), які імітують принципи поведінкового навчання. У таких системах

“агент” приймає рішення у динамічному середовищі, отримуючи “нагороду” або “штраф” залежно від ефективності дії. У логістиці цей підхід знайшов застосування у маршрутизації в реальному часі, де алгоритм самостійно визначає найкращий шлях для транспортного засобу з урахуванням змінних умов.

Перевага RL-моделей полягає в їхній здатності самонавчатися, поступово покращуючи власну стратегію без необхідності постійного контролю з боку людини.

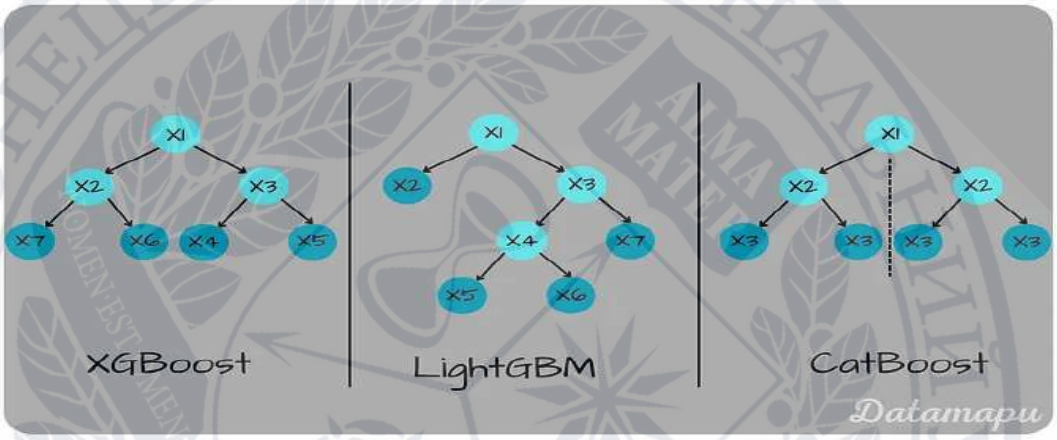

Рисунок 1.8 – Архітектура дерев рішень у моделях XGBoost, LightGBM та CatBoost

Разом із тим, RL-підхід вимагає значних ресурсів і складної побудови функції винагороди. Якщо система некоректно визначає мету, вона може навчитися ухвалювати формально “оптимальні”, але нефункціональні рішення. Тому застосування підкріплювального навчання у логістиці потребує ретельного моделювання середовища та інтеграції з реальними даними з транспортних систем.

У практиці сучасного бізнесу найкращі результати досягаються завдяки комбінуванню кількох методів Big Data-аналітики. Так звані ансамблеві моделі (наприклад, XGBoost, LightGBM, CatBoost) поєднують переваги різних алгоритмів, що дозволяє підвищити точність прогнозування й зменшити ризик переобучення [4]. У логістиці такі моделі використовуються для комплексного управління

ланцюгами постачань: наприклад, для поєднання прогнозів попиту з алгоритмами оптимізації маршрутів.

Інтеграційний підхід полягає у створенні єдиної аналітичної екосистеми, де різні моделі – кластеризаційні, класифікаційні, прогнозні та оптимізаційні – взаємодіють між собою через спільну платформу даних. Це дозволяє враховувати контекстні взаємозв'язки між процесами постачання, зберігання та транспортування, забезпечуючи адаптивність і самонавчання системи.

Порівняльний аналіз свідчить, що не існує універсального методу, придатного для всіх логістичних завдань.

- Якщо необхідна висока інтерпретованість і швидкість, доцільно застосовувати статистичні або деревоподібні моделі.
- Якщо головним критерієм є точність і здатність до самоадаптації, варто використовувати нейронні мережі або ансамблеві методи.
- Для динамічного управління в реальному часі оптимальним вибором стають моделі підкріплювального навчання.

На практиці найкращий результат дає поєднання різних підходів – наприклад, використання кластеризації для попередньої сегментації клієнтів, нейронних мереж для прогнозування попиту та алгоритмів RL для оперативної маршрутизації. Такий комплексний підхід дозволяє побудувати інтелектуальну систему управління ланцюгами постачань, здатну одночасно аналізувати, передбачати й адаптуватися до мінливих умов ринку.

У підсумку можна стверджувати, що ефективність Big Data-аналітики у логістиці визначається не лише вибором конкретного алгоритму, а передусім якістю даних, масштабованістю системи, інтеграцією моделей і здатністю організації впроваджувати отримані знання у практику. Інтелектуальний аналіз даних перестає бути допоміжним інструментом і стає центральним елементом цифрової логістичної стратегії, який визначає конкурентоспроможність підприємства в умовах глобальної економіки.

ВИСНОВКИ ДО РОЗДІЛУ 1

У першому розділі було розглянуто, як великі дані формують сучасний підхід до управління логістичними процесами. Проаналізовані джерела показали, що логістика сьогодні працює з великими, швидкими та різнорідними потоками інформації, тому якісна підготовка даних – їх очищення, узгодження та інтеграція – є необхідною передумовою для ефективного аналізу.

Вивчення методів аналітики, зокрема кластеризації, прогнозування та оптимізаційних моделей, підтвердило їхню ключову роль у підвищенні точності планування, скороченні витрат і побудові більш гнучких логістичних систем. З огляду на це, використання Big Data дозволяє підприємствам переходити від реактивного до проактивного управління.

Отже, проведений огляд заклав методологічну основу для формування задач другого розділу та подальшої розробки програмного рішення, спрямованого на оптимізацію маршрутів і підвищення ефективності логістичних операцій.

РОЗДІЛ 2
КОНЦЕПТУАЛЬНА МОДЕЛЬ ОПТИМІЗАЦІЇ ЛОГІСТИЧНОГО ПРОЦЕСУ

2.1. Аналіз предметної області та виявлення ключових логістичних проблем

Ефективне функціонування логістичних систем є критичним фактором для підприємств, що здійснюють транспортування товарів, управління складськими запасами та забезпечення своєчасного виконання замовлень. У сучасних умовах значна частина логістичних даних формується у вигляді великих масивів інформації, що надходять з різних джерел: систем управління перевезеннями, мобільних додатків кур'єрів, складських комплексів, GPS-трекерів, облікових систем підприємства тощо [16]. Це створює необхідність застосування методів аналітичного опрацювання даних для підтримки прийняття рішень у логістиці.

Аналіз предметної області показує, що логістичні компанії стикаються із низкою системних проблем, які значно ускладнюють планування поставок, маршрутизацію транспортних засобів та контроль за виконанням операцій. Нижче наведено основні проблеми, що потребують вирішення та можуть бути оптимізовані шляхом застосування методів аналітики та математичного моделювання.

Неефективність маршрутів перевезення. Однією з ключових проблем є побудова субоптимальних маршрутів доставки. Часто трапляється, що транспортні засоби виконують зайві кілометражі, обслуговують нераціонально сформовані групи клієнтів або рухаються за маршрутами, які не враховують:

- особливості дорожньої інфраструктури;
- часові обмеження доставки;
- реальне географічне розташування клієнтів;
- нерівномірність попиту у різні періоди;

– завантаженість транспортних засобів.

Неефективна маршрутизація призводить до перевитрат пального, збільшення часу доставки, зниження продуктивності логістичної мережі та підвищення операційних витрат.

Коливання попиту. Логістичні процеси значною мірою залежать від здатності підприємства прогнозувати обсяги майбутніх замовлень. Рівень попиту суттєво змінюється залежно від:

– сезонних факторів;
– маркетингових активностей постачальників;
– специфіки товарної групи;
– зовнішніх економічних умов;
– поведінкових змін споживачів.

Невірна оцінка попиту може спричинити надлишкові або недостатні запаси, що відповідно веде до збільшення витрат на зберігання або втрати клієнтів через недоступність товару. Тому створення моделей прогнозування на основі аналізу часових рядів є важливим інструментом підтримки логістичних рішень.

Затримки доставки. Затримки виконання замовлень негативно впливають на якість сервісу, формують додаткові витрати та знижують рівень задоволеності клієнтів. Причинами затримок найчастіше є:

– невраховані транспортні обмеження;
– неправильна оцінка часу обслуговування клієнтів;
– нерівномірне розташування точок доставки;
– непередбачені зміни у завантаженості доріг;
– недосконаліть планування маршрутів.

Аналіз історичних даних дозволяє виявити закономірності, що призводять до затримок, і використати їх для побудови більш надійних планів доставки.

Нерівномірне завантаження транспорту: У багатьох логістичних системах транспортні засоби використовуються нераціонально. Частина автомобілів може

працювати на межі завантаженості, тоді як інші – виконують маршрути з частково заповненим вантажним простором [17]. Це призводить до:

- перевитрат ресурсів;
- збільшення кількості рейсів;
- зниження продуктивності логістичної мережі;
- необхідності додаткового транспорту у пікові періоди.

Застосування методів кластеризації та оптимізації дозволяє формувати збалансовані групи доставок та рівномірно розподіляти навантаження на транспорт.

Розрізненість джерел даних. Логістичні дані надходять з великої кількості систем, що часто не інтегровані між собою:

- системи управління замовленнями;
- склади та облікові системи;
- транспортні модулі;
- системи моніторингу пересування;
- геоінформаційні сервіси.

Через це виникають труднощі з побудовою цілісної аналітичної моделі, а також збільшується ймовірність помилок через дублювання, втрату або некоректну обробку даних. Інтеграція даних у єдине інформаційне середовище є необхідною умовою для застосування математичних методів оптимізації.

Таким чином, аналіз предметної області свідчить, що ключові логістичні проблеми пов’язані з неефективним використанням транспортних ресурсів, відсутністю точних прогнозів попиту, нерівномірністю завантаження маршруту та розрізненістю джерел даних. Ці проблеми можуть бути вирішені шляхом побудови аналітичної системи, що використовує методи статистичного аналізу, математичного моделювання та оптимізації. Саме розробка такої системи і є основною метою даної роботи.

2.2. Формалізація задачі оптимізації на основі великих даних

Дослідивши ключові проблеми логістичної галузі, а саме: неефективність маршрутів, коливання попиту, затримки доставки, нерівномірне завантаження транспорту та розрізненість джерел даних, тепер можна сформувати точне бачення того, які саме процеси потребують математичного опису та подальшої оптимізації. Саме ці проблеми визначають структуру задачі, яка розв'язується у межах даної магістерської роботи: підвищення ефективності системи доставки шляхом оптимізації маршрутів і підтримки управлінських рішень через структурований аналіз великих масивів логістичних даних [3].

Оскільки логістичні системи оперують великими обсягами неоднорідної інформації, формалізація задачі оптимізації повинна включати: побудову маршрутів, прогнозування попиту, класифікацію або групування точок доставки та об'єднання даних з різних джерел у єдину структуру. У цьому підпункті наведено математичні моделі та формальні постановки, що лежать в основі розроблених подальших модулів [18]. Проблема побудови маршрутів доставки традиційно описується як задача маршрутизації транспортних засобів (Vehicle Routing Problem, VRP). У загальному випадку ця задача передбачає визначення найкоротших або найефективніших маршрутів для одного або декількох транспортних засобів, що повинні доставити товари певним клієнтам. Нехай маємо:

- множину точок доставки $C = \{c_1, c_2, \dots, c_n\}$;
- транспортний засіб або флот транспортних засобів із вантажопідйомністю Q;
- матрицю відстаней або часу пересування між точками $D = [d_{ij}]$;
- часові обмеження або часові вікна $T = \{t_i\}$ (за потреби).

Необхідно знайти такий маршрут або набір маршрутів $R = \{r_1, r_2, \dots, r_n\}$, які мінімізують цільову функцію:

$$min = \sum_{r \in R\, i} \sum_{j \in r} d_{ij} \qquad (1.1)$$

за умов, що кожна точка доставки має бути відвідана рівно один раз; жоден маршрут не перевищує вантажопідйомності Q; виконуються часові обмеження T (за потреби).

Це дозволяє математично сформулювати ключову задачу оптимізації: зменшити загальні витрати на доставку за рахунок пошуку найкращої схеми пересування транспортних засобів.

Для її вирішення, нам потрібно буде реалізувати в програмному вигляді формулу відстані між точками. Оскільки точки задані широтою та довготою, використовується формула Haversine:

$$d(p_1, p_2) = 2R\ arcsin\ (\sqrt{sin^2 \frac{\Delta\varphi}{2} + cos\varphi 1\ cos\varphi 2\ sin^2\ \frac{\Delta\lambda}{2}}), \quad (1.2)$$

де, R = 6371 - радіус Землі;

φ, λ – географічні координати.

Ця метрика дозволяє врахувати кривизну поверхні Землі, що важливо для коректного логістичного розрахунку [19].

Для розробки транспортних маршрутів можуть застосовуватися як прості (базові), так і більш оптимізовані алгоритми. Їх формалізація є важливою для подальшої реалізації у розділі 3. В цій роботі буде використано простий метод наївного формування маршруту та більш оптимальний підхід, жадібний алгоритм “найближчий сусід”.

Найпростішим варіантом є послідовне розбиття множини точок доставки на групи розміром k, що відповідає максимальному навантаженню одного транспортного засобу: $r_i = \{C(i-1)\ k\ +1, \ldots, Cik\}$. Сумарна довжина такого маршруту:

$$L_{naive}\ = \sum_{ri} \sum_{j\,=\,1}^{|ri|\,-\,1} d(c_j, c_j + 1) \quad (1.3)$$

Зазвичай цей метод застосовується в ситуаціях, коли немає обмеження по черговості доставки, або ж коли дані надходять дуже швидко і потрібне миттєве групування. Також коли виконується попередній аналіз перед оптимізацією. Наприклад: групування магазинів за регіонами, формування первинних маршрутних зон у службах доставки, базове планування у невеликих логістичних компаніях. Оптимальнішим є підхід, коли на кожному кроці обирається найближча ще не відвідана точка:

$$ct + 1 = arg \frac{min}{c \in U} d(ct, c), \tag{1.4}$$

де U – множина невідвіданих точок.

Переважно цей алгоритм «найближчого сусіда» широко застосовують у кур'єрських службах при локальній маршрутизації; оптимізації обходу об'єктів (наприклад, зняття показників електролічильників); робототехніці для обходу точок; навігаційних системах, де потрібна швидка побудова прийнятно ефективного маршруту.

- Перевага – швидкість роботи та простота реалізації.
- Недолік – алгоритм може давати локально оптимальні рішення, але не гарантує глобальної оптимальності.

Ця модель буде нами використовуватись як базова оптимізаційна стратегія для порівняння результатів [20].

Коливання попиту (ідентифіковані у розділі 2.1) створюють значну невизначеність при плануванні логістичних операцій. Для зменшення ризику дефіциту товарів або їхнього надлишку необхідно створити математичну модель прогнозування, яка здатна оцінити майбутні обсяги замовлень на основі історичних даних. Нехай маємо часовий ряд попиту:

$$y_t = кількість\ замовлень\ у\ момент\ часу\ t \tag{1.5}$$

Метою є побудова моделі:

$$\hat{y}_{t+1} = f(y_t, y_{t-1}, \ldots, y_{t-k}) \quad (1.6)$$

де функція f описує залежності між попередніми та майбутніми значеннями попиту. У межах даної роботи можуть застосовуватися такі методи: метод ковзного середнього, метод експоненціального згладжування, ARIMA, регресійні моделі. Це дозволяє підтримувати прийняття рішень щодо: прогнозування пікових навантажень, планування кількості транспорту, оцінки майбутнього завантаження маршрутів [20-23].

Оскільки у реальних логістичних системах точки доставки нерівномірно розподілені по території, виникає потреба в їхньому попередньому групуванні. Метою кластеризації є визначення регіональних груп клієнтів, для яких можуть бути побудовані окремі маршрути. Нехай:

- $X = \{x_1, x_2, \ldots, x_n\}$ – координати точок доставки;
- потрібно визначити k кластерів.

Завдання кластеризації формулюється як мінімізація суми квадратів відстаней до центроїдів:

$$min = \sum_{i=1}^{k} \sum_{x_j \in C_i} ||x_j - \mu_i||^2 \quad (1.7)$$

де μ_i – центр i-го кластера.

Цей етап спрощує задачу маршрутизації, оскільки: маршрути стають коротшими, навантаження між транспортними засобами розподіляється рівномірніше, а також підвищується швидкість планування.

Через розрізненість джерел даних (описану у 2.1) необхідно сформувати єдину структуру даних, придатну для подальшої обробки. Формально цю задачу можна визначити як побудову уніфікованої таблиці [24]:

$$D = \{(c_i, x_j, y_i, w_i, t_i, d_i, \ldots)\}, \quad (1.8)$$

де: c_i – ID замовлення

x_j , y_i – координати

w_i – вага

t_i – час надходження

d_i – час доставки

інші атрибути залежно від датасету.

Метою є забезпечення: відсутності пропусків, усунення дублювання, оректності форматів, узгодження часових та географічних параметрів.

Отже, з урахуванням результатів попереднього аналізу, ключовою задачею, яку необхідно вирішити в рамках магістерської роботи, є побудова аналітичної системи, що виконує інтеграцію логістичних даних, кластеризацію точок доставки, прогнозування попиту та оптимізацію маршрутів на основі математичних методів аналізу великих даних. Розробка такої системи дозволить:

- скоротити загальний кілометраж маршрутів;
- зменшити витрати на транспортування;
- підвищити рівномірність завантаження доставки;
- підвищити точність планування;
- знизити ризик затримок.

Усі зазначені задачі будуть реалізовані у розділі 3 шляхом створення програмного модуля для аналітичного опрацювання логістичних даних.

Проведений у попередніх підрозділах аналіз предметної області дозволив визначити ключові логістичні проблеми та сформулювати відповідні математичні задачі, що потребують аналітичного опрацювання великих даних. З огляду на це, постає необхідність визначити вимоги, яким повинні відповідати методи інтелектуального аналізу даних, щоб вони могли бути коректно застосовані у побудові інтегрованої логістичної системи. Оскільки дана робота є магістерським дослідженням, важливо не лише обрати конкретні алгоритми, але й обґрунтувати, які властивості мають бути притаманні цим методам, щоб забезпечити точність,

надійність та практичну цінність одержаних результатів.

Першою вимогою є здатність методів працювати з великими масивами логістичних даних, що мають різну природу: географічні координати, часові ряди попиту, вагові характеристики замовлень, інформацію про тривалість доставки та інші атрибути. Логістичні процеси генерують дані у значних обсягах, тому методи мають бути достатньо стійкими до шуму, а також здатними обробляти неповні або неоднорідні записи без суттєвого погіршення якості результатів. Це особливо важливо у випадках, коли дані надходять із різних систем, а інтеграція інформації здійснюється поступово [25].

Другою важливою вимогою є інтерпретованість результатів. У логістичних компаніях управлінські рішення приймаються людьми – логістами, аналітиками, операторами. Тому методи, що використовуються для кластеризації, прогнозування або оптимізації, повинні давати результати, які можуть бути пояснені у термінах логістичних показників: відстані, тривалості доставки, обсягів замовлень, рівня завантаження транспорту. Результати аналізу повинні бути не лише математично коректними, але й практично зрозумілими для прийняття рішень.

Наступною вимогою є стійкість моделей до коливань попиту. Як було показано, попит у логістиці має сезонний характер та піддається впливу зовнішніх факторів. Тому алгоритми аналізу часових рядів повинні вміти коректно працювати зі змінними трендами, зберігати стабільність прогнозу і не бути надто чутливими до короткострокових відхилень. Це необхідно для того, щоб система могла забезпечувати планування кількості транспорту, ресурсів та потенційного навантаження маршрутів [26].

Ще однією важливою вимогою є здатність методів підтримувати оптимізаційні процеси, пов’язані з маршрутизацією. Моделі мають забезпечувати коректне групування точок доставки, визначення їхньої географічної структури та формування маршрутів, що дозволяють мінімізувати кілометраж, час виконання доставки та відповідні витрати. Алгоритми мають бути достатньо швидкими, щоб

їх можна було застосувати у практичних умовах, де час на прийняття рішення часто обмежений. Окрім того, необхідно враховувати вимогу до узгодженості даних, що надходять до моделі. Через розрізненість джерел інформації методи аналізу повинні толерувати невеликі відмінності у форматах даних, коригувати аномалії, виявляти та усувати дублікати. Це забезпечує підготовку чистого аналітичного простору для побудови прогнозних та оптимізаційних моделей [27-30].

У підсумку необхідно зазначити, що методи інтелектуального аналізу даних, які застосовуються для вирішення задач, сформульованих у підрозділі 2.2, повинні відповідати ряду критеріїв: здатність працювати з великими масивами реальних логістичних даних, забезпечувати інтерпретованість результатів, підтримувати процеси прогнозування та оптимізації, бути стійкими до неповноти даних та мінливості попиту, а також забезпечувати прийнятну швидкість виконання. Саме ці вимоги визначають підхід до вибору методів і формують основу концептуальної схеми аналітичної системи, яку буде представлено у наступному підрозділі [31].

2.3. Концептуальна схема інтегрованої аналітичної системи логістики

На підставі результатів досліджень, наведених у підрозділах 2.1–2.3, сформовано необхідність побудови інтегрованої аналітичної системи, яка здатна забезпечити повний цикл опрацювання логістичних даних – від моменту їхнього надходження до отримання конкретних оптимізаційних рішень. Така система має бути достатньо гнучкою, щоб охоплювати різноманітні джерела інформації, підтримувати масштабування, працювати зі значними обсягами даних та забезпечувати формування результатів, придатних для практичного використання у логістичних процесах.

У логістиці дані відображають реальну діяльність компанії: надходження замовлень, їхню географічну прив’язку, часові характеристики, транспортні параметри, історію попиту, характеристику рейсів, обмеження роботи складів та інші властивості. Вони є неоднорідними, різновимірними та такими, що надходять

у різний час, що зумовлює складність їхньої обробки. Інтегрована аналітична система покликана забезпечити їх узгодження та перетворення у форму, придатну для подальшого моделювання. Першим логічним етапом є функціональний модуль збору даних. Він об'єднує джерела різного типу – таблиці замовлень, GPS-логи, дані складів, інформацію про виконані рейси, карти дорожньої інфраструктури, календарні особливості (свята, вихідні) та інші параметри. Оскільки у реальних умовах логістичні дані надходять нерегулярно, модуль збору виконує також роль фільтра та контролю вхідних потоків, забезпечуючи, щоб система опрацьовувала лише валідні дані.

Другим компонентом є модуль підготовки та очищення даних. Його роль полягає у виявленні пропусків, корекції аномалій, нормалізації значень параметрів та уніфікації форматів. Наявність цього етапу є критичною, оскільки математичні моделі чутливі до некоректних даних або відхилень у структурі. Крім того, підготовка даних дозволяє зменшити шум, що призводить до більшої точності кластеризації та підвищення стійкості моделей прогнозування. Підтримка якісного інформаційного середовища є основою для роботи наступних модулів системи.

Аналітичний модуль системи реалізує застосування математичних методів до підготовлених даних. На цьому етапі здійснюється кластеризація точок доставки відповідно до їхніх географічних координат, аналіз структури попиту, виявлення динаміки, аналіз часових залежностей, а також формування статистичних характеристик логістичних потоків. Завдяки цьому створюється узагальнене уявлення про типові логістичні патерни, нерівномірність завантаження, сезонність та інші системні особливості [31-33].

Оптимізаційний модуль є центральним етапом системи. Його призначення – побудова оптимальних маршрутів доставки, що мінімізують відстань, час у дорозі або сумарні логістичні витрати. З урахуванням результатів аналітики система може визначати межі кластерів доставки, оцінювати можливе навантаження на транспорт, погоджувати маршрути з прогнозованими піковими періодами, а також

коригувати розподіл замовлень між рейсами. Цей модуль може повторно використовувати результати аналітичного блоку для уточнення маршрутів, якщо оптимізаційний алгоритм виявить нерівномірний розподіл точок або потенційні логістичні конфлікти [34].

Завершальним етапом є модуль візуалізації результатів. Саме він перетворює аналітичні розрахунки та оптимізаційні рішення у графічну форму, яка є зрозумілою для логістів та керівників. Візуалізація дозволяє відобразити сформовані кластери, мережу оптимальних маршрутів, часові графіки попиту, рівень завантаження транспортних засобів, а також порівняння початкових і оптимізованих характеристик доставки. Завдяки цьому результати не лише математично коректні, але й доступні для оперативного прийняття рішень.

Нище наведено узагальнену концептуальну схему інтегрованої аналітичної системи логістики, яка відображає взаємозв’язки між основними її компонентами.

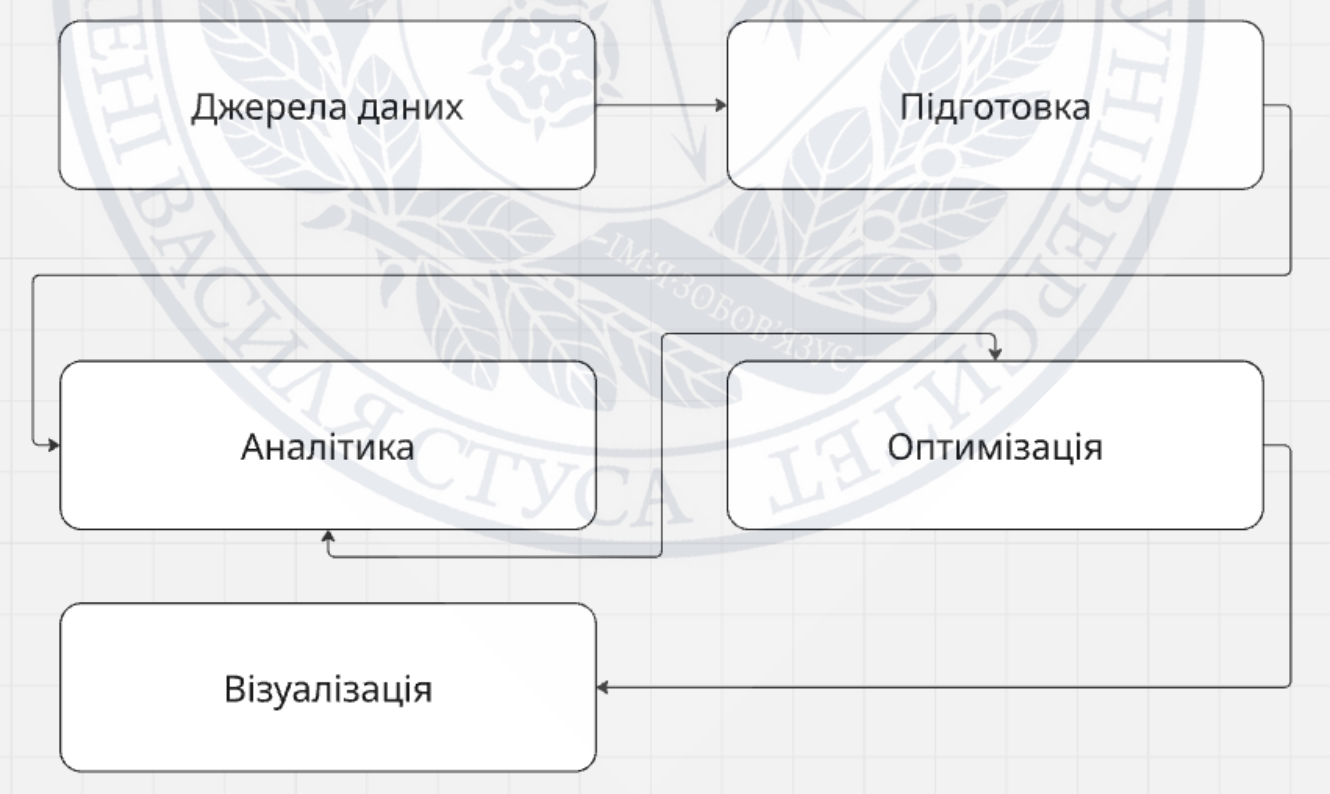

Рисунок 2.1 – Концептуальна схема інтегрованої аналітичної системи логістики

Загальна логіка побудови інтегрованої системи передбачає послідовність виконання ключових етапів, а також можливість повернення до попередніх модулів для коригування даних або повторного застосування аналітичних процедур. Таким чином, система є не статичною, а динамічною, що дозволяє адаптуватися до мінливих умов роботи логістичної мережі [35].

Аналіз логістичної предметної області, проведений у попередніх підрозділах, дав змогу визначити ключові проблеми, що виникають у процесі планування доставки, а також сформулювати математичні задачі, які повинна вирішувати інтегрована аналітична система. На основі цього необхідно обґрунтувати вибір конкретних аналітичних методів та алгоритмів, що застосовуватимуться у розробленій системі. Таке обґрунтування є важливою складовою магістерської роботи, оскільки від вибору методів залежить точність прогнозування, ефективність оптимізаційних процесів та загальна надійність результатів [36].

У логістиці дані характеризуються високою варіативністю: змінюються обсяги попиту, географія розташування клієнтів, конфігурації маршрутів, часові обмеження та інші параметри, що прямо впливають на прийняття рішень. Для роботи з такими даними доцільно застосовувати підходи, які забезпечують інтерпретованість результатів, стійкість до шуму та здатність працювати з великими масивами інформації. Саме ці критерії визначили вибір алгоритмів, які будуть використані в розділі 3 при створенні програмного забезпечення.

При вирішенні задачі групування точок доставки доцільним є застосування методів кластеризації, зокрема алгоритмів, що ґрунтуються на мінімізації внутрішньокластерної відстані. Алгоритм k-means є доцільним вибором, оскільки він добре працює з географічними координатами, дозволяє визначити компактні групи точок та забезпечує рівномірний розподіл навантаження між потенційними маршрутами. Його результати легко інтерпретуються та візуалізуються, що робить його практичним для використання в логістичних задачах.

Для прогнозування попиту оптимальним є застосування методів аналізу

часових рядів, які враховують наявність трендів, сезонних компонентів та випадкових коливань. Метод ARIMA або модифіковані варіанти експоненціального згладжування дозволяють здійснити короткостроковий прогноз на основі історичних даних. Вибір таких моделей обумовлений їхньою здатністю відображати динаміку замовлень та визначати характерні закономірності, що є важливим для ефективного планування логістичних ресурсів.

Для задачі оптимізації маршрутів обґрунтованим є застосування методів, заснованих на вирішенні задачі маршрутизації транспортних засобів. Через те, що ця задача є обчислювально складною (NP-складною), доцільно використовувати алгоритми, які забезпечують практично прийнятний час обчислень. Зокрема, бібліотека OR-Tools, заснована на класичних принципах комбінаторної оптимізації, дозволяє обчислювати маршрути з урахуванням відстаней, часових вікон, вантажопідйомності транспорту та інших логістичних обмежень. Використання такого підходу є виправданим з огляду на його стабільність, математичну строгість та практичну ефективність [37].

Окремої уваги потребує інтеграція цих методів у єдину систему. Вибір алгоритмів, які не потребують значних обчислювальних ресурсів, дозволяє реалізувати програмне забезпечення у доступних середовищах, зокрема в онлайн-інтерпретаторах Python [38]. Це забезпечує можливість відтворення результатів на будь-якому комп’ютері без встановлення спеціального програмного забезпечення, що є важливою вимогою для магістерської роботи та подальшої презентації.

Таким чином, вибір методів кластеризації, аналізу часових рядів та математичної оптимізації зумовлений специфікою логістичних даних, сформульованими у попередніх підрозділах вимогами та необхідністю забезпечення інтерпретованості й надійності результатів. Застосування цих підходів створює основу для побудови інтегрованої аналітичної системи, здатної підвищувати ефективність логістичних процесів шляхом об’єктивного використання великих масивів даних [39].

ВИСНОВКИ ДО РОЗДІЛУ 2

У другому розділі проведено формалізацію задачі оптимізації логістичних процесів на основі аналізу великих даних. На основі досліджених особливостей предметної області було визначено ключові проблеми сучасної логістики – нерівномірний розподіл точок доставки, коливання попиту, неузгодженість даних і потреба у скороченні транспортних витрат. Це дозволило сформувати чітке бачення того, які саме процеси підлягають оптимізації та які вимоги висуваються до моделей обробки даних.

У розділі обґрунтовано математичний опис задачі, що включає кластеризацію географічних точок, прогнозування попиту та пошук оптимальних маршрутів на основі відстаней і логістичних обмежень. Сформована концептуальна схема аналітичної системи демонструє взаємодію її модулів та визначає роль кожного етапу – від підготовки даних до побудови оптимізаційних моделей.

Також проаналізовано підходи й алгоритми, на яких ґрунтуватиметься програмна реалізація: методи кластеризації, математичні моделі прогнозування та алгоритми пошуку маршрутів. Обґрунтовано їх вибір з огляду на масштабність даних, просторову природу задачі та необхідність швидкого отримання результатів.

Таким чином, у розділі 2 сформовано методологічний та математичний фундамент майбутньої системи оптимізації. На його основі буде реалізовано програмне забезпечення, описане у третьому розділі, що дозволить практично застосувати обрані методи та перевірити їх ефективність для реальних логістичних даних.

РОЗДІЛ 3
РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ ПРОЦЕСІВ ЗАСОБАМИ BIG DATA-АНАЛІТИКИ

3.1. Загальна архітектура програмної системи

У попередньому розділі було сформовано концептуальну модель інтегрованої аналітичної системи логістики, яка визначає основні етапи опрацювання даних: від збору та підготовки до аналітики, оптимізації та візуалізації результатів. У рамках даної магістерської роботи ця концептуальна схема конкретизується у вигляді програмної системи, що реалізує зазначені етапи у формі послідовно взаємодіючих програмних модулів. Метою даного підрозділу є опис загальної архітектури програмного забезпечення, яке буде розроблено та реалізовано у наступних підрозділах.

Розроблюване програмне забезпечення орієнтоване на аналітичну обробку логістичних даних і оптимізацію маршрутів доставки. Вихідними даними для системи є набір записів про замовлення, що містять інформацію про ідентифікатор замовлення, координати точки доставки, обсяг або вагу відправлення, час надходження замовлення та інші атрибути, необхідні для подальшого аналізу. Програмна система повинна забезпечувати читання таких даних з зовнішнього файлу, їхню підготовку, виконання аналітичних операцій, оптимізаційних розрахунків та формування результатів у вигляді таблиць і графічних матеріалів. У програмній реалізації доцільно використати сучасну мову програмування загального призначення, що підтримує роботу з табличними даними, науковими бібліотеками та візуалізацією результатів. У рамках даної роботи такою мовою є Python.

Архітектура програмної системи побудована на модульному підході [40]. Це означає, що окремі етапи опрацювання даних реалізуються у вигляді логічно відокремлених компонент, кожен з яких відповідає за свою частину

функціональності. Такий підхід спрощує структуризацію коду, полегшує тестування окремих частин програми та дає можливість розширення або заміни окремих модулів без істотних змін в інших частинах системи.

Рисунок 3.1 – Архітектура програмної системи для аналітичної обробки логістичних даних

Першим ключовим елементом архітектури є модуль роботи з вхідними даними. Його завданням є завантаження набору даних із зовнішнього джерела (наприклад, табличного файлу формату CSV) та перетворення їх у внутрішнє подання, зручне для подальшої обробки. На цьому етапі здійснюється базова перевірка структури даних, коректності форматів полів та наявності обов’язкових атрибутів. Важливо, щоб у подальших модулях використовувалося єдине узгоджене подання даних, тому саме цей модуль виконує роль «входу» в систему.

Наступним компонентом є модуль підготовки та очищення даних (Preprocessing). Він реалізує ті функції, що були визначені у розділі 2 як необхідні для забезпечення якості інформаційної бази: виявлення та обробка пропусків, усунення дублювань, корекція аномальних значень, нормалізація числових параметрів. Результатом роботи цього модуля є очищений набір даних із уніфікованою структурою, який далі передається до аналітичної частини системи. Таким чином, модуль підготовки даних є проміжною ланкою між «сирими» вхідними даними та математичними моделями, які використовуються надалі.

Центральне місце в архітектурі посідає аналітичний модуль (Analytics), який відповідає за виконання обчислювальних процедур, пов’язаних із кластеризацією точок доставки та аналізом попиту. У цьому модулі будуть реалізовані алгоритми

кластеризації, що групують замовлення за географічною ознакою, а також методи аналізу часових рядів, які дозволяють оцінити динаміку попиту та сформувати прогнози на майбутні періоди. Аналітичний модуль працює безпосередньо з очищеними даними, формує проміжні аналітичні результати та передає їх до наступного рівня – оптимізаційного.

Оптимізаційний модуль (Optimization) реалізує модель маршрутизації, сформульовану у підрозділі 2.2. На вхід він отримує інформацію про кластеризовані точки доставки, параметри попиту, обмеження на транспортні ресурси та інші необхідні характеристики. На основі цього модуль обчислює маршрути доставки, які мінімізують загальну довжину шляху або інші цільові показники. У рамках магістерської роботи передбачається використання алгоритмів комбінаторної оптимізації, які забезпечують практично прийнятний час обчислень при заданому обсязі даних. Таким чином, оптимізаційний модуль є тим компонентом, який безпосередньо формує рекомендації щодо покращення логістичних процесів.

Важливою складовою архітектури є модуль візуалізації результатів (Visualization). Його призначення полягає в тому, щоб подати результати роботи аналітичного та оптимізаційного модулів у наочній формі. До таких результатів належать графіки розподілу кластерів, діаграми динаміки попиту, схеми маршрутів та порівняльні візуалізації показників до і після оптимізації. У контексті магістерської роботи цей модуль відіграє особливу роль, оскільки саме його результати будуть використані для ілюстрації роботи програмної системи у тексті роботи та під час презентації на захисті.

Над усіма переліченими компонентами знаходиться керуюча логіка, яка задає послідовність виконання операцій: завантаження даних (Data Input), їхня підготовка, аналітичне опрацювання, оптимізація маршрутів та формування звітів. Ця логіка реалізується у вигляді основного сценарію (головного файлу програми), який викликає відповідні функції або методи окремих модулів у потрібному

порядку. Такий підхід дозволяє чітко розмежувати функціональні частини системи та спростити подальшу підтримку коду.

Узагальнено загальна архітектура програмної системи відповідає концептуальній схемі, наведений у розділі 2, але деталізує її для рівня програмної реалізації. Вона дозволяє відобразити логістичні процеси у вигляді послідовних етапів обробки даних, забезпечуючи поступовий перехід від первинної інформації до оптимізаційних рішень. У наступних підрозділах буде розглянуто реалізацію кожного з описаних модулів, а також продемонстровано їхню роботу на конкретному наборі логістичних даних.

3.2. Модуль підготовки та очищення даних

Як було показано у другому розділі, коректність результатів аналітичної обробки та оптимізації логістичних процесів суттєво залежить від якості вхідних даних. Тому першим кроком програмної реалізації є створення модуля підготовки та очищення датасету, який містить інформацію про доставку замовлень. Для демонстрації розробленої системи було обрано відкритий логістичний датасет, доступний на платформі Kaggle [40]. При його виборі враховувалися такі критерії: наявність географічних координат точок доставки, часових характеристик замовлень, параметрів, пов’язаних із обсягом або вартістю замовлення, а також вільний доступ до даних у навчальних цілях. Після аналізу доступних наборів даних був обраний датасет «Food Delivery Data», який відповідає цим вимогам і містить достатній обсяг інформації для проведення комплексного аналізу логістичних процесів. Цей набір даних, train.csv, містить відомості про кур’єрські доставки: ідентифікатори замовлень, характеристики кур’єрів, координати ресторану та місця доставки, часові параметри, умови дорожнього руху, а також фактичний час виконання доставки. Цей датасет, доступний для завантаження за посиланням: https://www.kaggle.com/datasets/gauravmalik26/food-delivery-dataset

Структура цього датасету включає такі ключові поля: ID, Delivery_person_ID,

Delivery_person_Age, Delivery_person_Ratings, Restaurant_latitude, Restaurant_longitude, Delivery_location_latitude, Delivery_location_longitude, Order_Date, Time_Orderd, Time_Order_picked, Weatherconditions, Road_traffic_density, Vehicle_condition, Type_of_order, Type_of_vehicle, multiple_deliveries, Festival, City та Time_taken(min). Наявність географічних координат, параметрів кур'єра, характеристик середовища та часу доставки робить цей набір даних придатним для подальшого аналізу та оптимізації логістичних маршрутів. Першим етапом реалізації модуля підготовки є завантаження даних та первинний огляд структури таблиці. Для цього використовується бібліотека pandas, яка дозволяє зручно працювати з табличними даними у форматі CSV:

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
# 1. Завантаження основного датасету
df = pd.read_csv("/content/Magister/train.csv")
print("Перші рядки таблиці:")
print(df.head())
print("\nЗагальна інформація про дані:")
print(df.info())
```

За допомогою цього фрагмента коду виконується завантаження датасету та виведення перших рядків таблиці й загальної інформації про стовпці: типи даних, кількість рядків і наявність пропусків. Це дозволяє переконатися в коректності структури файлу та узгодженості полів.

Далі здійснюється аналіз наявності пропусків у кожному зі стовпців. Пропуски можуть призводити до викривлення результатів аналітичних розрахунків, тому їх необхідно або усунути, або обробити:

```
# 2. Перевірка пропусків
print("\nКількість пропусків у кожному стовпці:")
```

```
print(df.isnull().sum())
# 3. Видалення рядків з пропусками (за потреби)
df_clean = df.dropna()
print("\nПісля видалення пропусків:")
print(df_clean.isnull().sum())
```

У межах даного датасету відсутні пропуски у стовпцях, тому операція видалення є формальною та застосовується для узагальнення підходу. Однак у разі появи неповних записів цей крок дозволив би підтримувати узгодженість даних.

Наступним кроком є перевірка наявності дублюючих записів. Наявність однакових рядків могла б призвести до подвійного врахування окремих доставок у процесі аналізу, що спотворило б статистичні показники:

```
# 4. Перевірка та видалення дублюючих рядків
duplicates_count = df_clean.duplicated().sum()
print("\nКількість дублюючих записів:", duplicates_count)
df_clean = df_clean.drop_duplicates()
print("Після очищення дублікатів:", df_clean.duplicated().sum())
```

У цьому випадку дублікати відсутні, однак перевірка на їхню наявність є важливим етапом загальної процедури очищення даних, оскільки вона забезпечує стійкість підходу до різних наборів даних.

Далі виконується перевірка коректності географічних координат. Значення широти та довготи ресторану й місця доставки мають належати допустимому діапазону, інакше такі записи можуть спотворити результати подальшого аналізу, зокрема кластеризації та оптимізації маршрутів:

```
# 5. Фільтрація записів з коректними координатами
df_clean = df_clean[
    df_clean["Delivery_location_latitude"].between(-90, 90) &
```

```
    df_clean["Delivery_location_longitude"].between(-180, 180) &
    df_clean["Restaurant_latitude"].between(-90, 90) &
    df_clean["Restaurant_longitude"].between(-180, 180)
]
```

Таким чином усуваються записи, у яких координати виходять за межі фізично обґрунтованого діапазону, що підвищує надійність подальшого географічного аналізу.

Наступний важливий етап підготовки – приведення типів даних числових полів до коректного формату. Частина показників у вихідному датасеті зберігається у вигляді тексту, хоча за змістом вони є числовими (наприклад, вік кур’єра, рейтинг, кількість одночасних доставок). Це потребує явного перетворення:

```
# 6. Перетворення типів даних для числових полів
# Вік кур’єра
df_clean["Delivery_person_Age"] = pd.to_numeric(
    df_clean["Delivery_person_Age"], errors="coerce"
)
# Рейтинги кур’єра
df_clean["Delivery_person_Ratings"] = pd.to_numeric(
    df_clean["Delivery_person_Ratings"], errors="coerce"
)
# Кількість одночасних доставок
df_clean["multiple_deliveries"] = pd.to_numeric(
    df_clean["multiple_deliveries"], errors="coerce"
)
```

Параметр errors="coerce" у функції to_numeric дозволяє перетворити некоректні текстові значення у пропуски, які згодом будуть оброблені (за потреби вилучені), що запобігає аварійному завершенню програми під час виконання числових операцій.

Окремо обробляється стовпець Time_taken(min), у якому час доставки зберігається у текстовому форматі, що містить як опис, так і числову частину (наприклад, "(min) 24") [41]. Для його використання в подальшому аналізі потрібно виділити числове значення:

```
# 7. Очищення та конвертація часу доставки
df_clean["Time_taken(min)"] = (
    df_clean["Time_taken(min)"]
    .astype(str)
    .str.extract(r'(\d+)', expand=False)
)
df_clean["Time_taken(min)"] = pd.to_numeric(
    df_clean["Time_taken(min)"], errors="coerce"
)
# Видалення можливих пропусків після конвертації
df_clean = df_clean.dropna(subset=[
    "Delivery_person_Age",
    "Delivery_person_Ratings",
    "multiple_deliveries",
    "Time_taken(min)"
])
```

У результаті цього фрагмента коду числові значення часу доставки виділяються з рядків, а поля з некоректними або відсутніми значеннями виключаються з подальшого аналізу. Це забезпечує узгодженість ключових числових параметрів.

Щоб зменшити вплив аномальних значень на результати аналізу, для стовпця Time_taken(min) застосовується фільтрація за міжквартильним розмахом. Це дозволяє усунути надто малі або надто великі значення, які є нетиповими для більшості записів:

```
# 8. Виявлення та фільтрація аномальних значень для часу доставки
Q1 = df_clean["Time_taken(min)"].quantile(0.25)
Q3 = df_clean["Time_taken(min)"].quantile(0.75)
IQR = Q3 - Q1

df_clean = df_clean[
    ~((df_clean["Time_taken(min)"] < (Q1 - 1.5 * IQR)) |
      (df_clean["Time_taken(min)"] > (Q3 + 1.5 * IQR)))
]
```

Такий підхід дозволяє сфокусувати аналіз на типовому діапазоні часу доставки, що є більш релевантним для побудови аналітичних моделей та оптимізації маршрутів.

Завершальним етапом модуля підготовки є нормалізація вибраних числових ознак, які будуть використані в подальшій аналітичній обробці. Оскільки різні показники мають різні масштаби, доцільно привести їх до єдиного діапазону значень від 0 до 1:

```
# 9. Нормалізація вибраних числових ознак
numeric_cols = [
    "Delivery_person_Age",
    "Delivery_person_Ratings",
    "Vehicle_condition",
    "multiple_deliveries",
    "Time_taken(min)"
]
scaler = MinMaxScaler()
df_clean[numeric_cols] = scaler.fit_transform(df_clean[numeric_cols])
print("\nПерші рядки нормалізованих числових стовпців:")
print(df_clean[numeric_cols].head())
```

Після виконання цього фрагмента всі вибрані числові характеристики масштабуються у межах [0; 1]. Це спрощує подальшу побудову моделей кластеризації та аналізу, оскільки жоден із параметрів не домінує над іншими за рахунок більших числових значень.

У підсумку модуль підготовки та очищення формує оновлений набір даних df_clean, у якому усунено дублювання, приведено до коректного формату числові поля, вилучено аномальні значення та виконано нормалізацію ключових показників. Отримані дані можуть бути використані для реалізації аналітичних моделей та алгоритмів оптимізації, що розглядатимуться в наступних підрозділах.

3.3. Моделі кластеризації точок доставки та аналізу попиту

Після підготовки та очищення даних наступним етапом є реалізація аналітичних моделей, які дозволяють виділити просторові закономірності в розташуванні точок доставки та проаналізувати динаміку попиту в часі. У межах даного підрозділу розглядається дві задачі: кластеризація точок доставки за їхніми географічними координатами та аналіз попиту на основі часових характеристик замовлень. Отримані проміжні аналітичні результати надалі використовуватимуться при формуванні оптимізаційної моделі маршрутизації.

У попередньому підрозділі було сформовано очищений набір даних df_clean. На основі цього набору можна переходити до побудови просторової структури доставки та оцінювання нерівномірності попиту. Першим етапом є кластеризація точок доставки. Вона дозволяє згрупувати замовлення за географічною ознакою, виділивши зони, у межах яких доцільно формувати окремі маршрути. Для цього використовуються координати місця доставки: Delivery_location_latitude та Delivery_location_longitude. Для кластеризації у роботі застосовується алгоритм k-means, який забезпечує розбиття множини точок на фіксовану кількість кластерів.

```
    from sklearn.cluster import KMeans
# Виділення координат точок доставки
```

```
coords = df_clean[["Delivery_location_latitude",
"Delivery_location_longitude"]]
# Кількість кластерів (може змінюватися залежно від задачі)
k = 5
kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
df_clean["cluster"] = kmeans.fit_predict(coords)
# Перегляд перших кластеризованих записів
print(df_clean[["Delivery_location_latitude",
     "Delivery_location_longitude",
     "cluster"]].head())
```

У цьому фрагменті коду формується матриця координат, після чого до неї застосовується алгоритм k-means з наперед обраною кількістю кластерів k. Результатом є присвоєння кожному запису номера кластера, що зберігається в новому стовпці cluster. Це дозволяє надалі аналізувати структуру замовлень за регіонами та планувати маршрути в межах кожної кластерної групи.

Для того, щоб оцінити, як розподілилися замовлення між кластерами, доцільно розрахувати кількість записів у кожному кластері:

```
# Розподіл замовлень за кластерами
cluster_counts = df_clean["cluster"].value_counts().sort_index()
print("Кількість замовлень у кожному кластері:")
print(cluster_counts)
# Координати центрів кластерів
print("\nКоординати центрів кластерів:")
print(kmeans.cluster_centers_)
```

На основі цих результатів можна зробити висновки щодо рівномірності розподілу точок доставки між кластерами, а також про географічне розташування центрів кластерів. Це важливо для наступного етапу, де будуть формуватися маршрути в межах кожного кластера.

Другим аспектом аналітичної обробки є аналіз попиту в часі. Для цього використовується стовпець Order_Date, що містить дату замовлення. На основі цих даних формується часовий ряд, який відображає кількість замовлень за кожну календарну дату. Це дозволяє виявити періоди підвищеного або зниженого попиту, що важливо для планування навантаження на транспортні ресурси.

Спочатку виконується перетворення текстового представлення дати у формат дати та часу, а також агрегування кількості замовлень за днями:

```
# Перетворення дати замовлення у формат datetime
df_clean["Order_Date_parsed"] = pd.to_datetime(
    df_clean["Order_Date"],
    format="%d-%m-%Y",
    errors="coerce"
)
# Групування замовлень за датою
daily_demand = (
    df_clean.groupby("Order_Date_parsed")
    .size()
    .reset_index(name="orders_count")
    .sort_values("Order_Date_parsed")
)
print("Перші рядки добового попиту:")
print(daily_demand.head())
```

Таким чином формується окрема таблиця daily_demand, у якій кожному дню відповідає кількість виконаних замовлень. Цей часовий ряд є основою для подальшого аналізу динаміки попиту.

Для згладжування випадкових коливань та виявлення загальної тенденції використовується рухоме середнє. Воно дозволяє побачити зміну попиту на певному інтервалі та є простим, але інформативним інструментом для попереднього аналізу:

```
# Розрахунок 3-денного ковзного середнього попиту
daily_demand["rolling_mean_3"] = (
    daily_demand["orders_count"]
    .rolling(window=3)
    .mean()
)
print("Добовий попит з 3-денним ковзним середнім:")
print(daily_demand.head(10))
```

Ковзне середнє дає змогу зменшити вплив одиничних піків попиту та краще побачити загальну тенденцію. На основі цього ряду можна здійснити простий короткостроковий прогноз попиту. Наприклад, як основу для прогнозу кількості замовлень на найближчий день можна використати середнє значення за кілька останніх днів:

```
# Простий орієнтовний прогноз: середній попит за останні 7 днів
last_7_days_mean = daily_demand["orders_count"].tail(7).mean()
print("\nОрієнтовний прогноз попиту на наступний день:",
      round(last_7_days_mean, 2))
```

Такий підхід не претендує на повноцінну прогностичну модель, але є достатнім для демонстрації базових підходів до аналізу попиту в межах магістерської роботи. Він дозволяє отримати орієнтовну оцінку майбутнього навантаження, яка надалі може бути використана для планування ресурсів.

Отримані у цьому підрозділі проміжні аналітичні результати мають важливе практичне значення для оптимізації логістичних процесів. Кластери точок доставки визначають логічні зони обслуговування, у межах яких доцільно формувати маршрути, а аналіз динаміки попиту дає змогу оцінити нерівномірність навантаження в часі. У сукупності ці результати створюють інформаційну основу

для побудови в наступному підрозділі моделі оптимізації маршрутів доставки, яка враховуватиме як просторову структуру замовлень, так і особливості їхнього надходження в часі.

```
   Delivery_location_latitude  Delivery_location_longitude  cluster
0                   22.765049                   75.912471        0
1                   13.043041                   77.813237        4
2                   12.924264                   77.688400        4
3                   11.053669                   77.026494        4
4                   13.012793                   80.289982        4
Кількість замовлень у кожному кластері:
cluster
0    13779
1     3386
2     5904
3     3550
4    15883
Name: count, dtype: int64

Координати центрів кластерів:
[[2.06461530e+01 7.38861789e+01]
 [6.29681040e-02 6.29681040e-02]
 [2.73690486e+01 7.68428939e+01]
 [2.35864529e+01 8.54352021e+01]
 [1.33648165e+01 7.78318257e+01]]
Перші рядки добового попиту:
  Order_Date_parsed  orders_count
0        2022-02-11           916
1        2022-02-12           803
2        2022-02-13           893
3        2022-02-14           788
4        2022-02-15           876
Добовий попит з 3-денним ковзним середнім:
  Order_Date_parsed  orders_count  rolling_mean_3
0        2022-02-11           916             NaN
1        2022-02-12           803             NaN
2        2022-02-13           893      870.666667
3        2022-02-14           788      828.000000
4        2022-02-15           876      852.333333
5        2022-02-16           797      820.333333
6        2022-02-17           873      848.666667
7        2022-02-18           792      820.666667
8        2022-03-01          1055      906.666667
9        2022-03-02           943      930.000000

Орієнтовний прогноз попиту на наступний день: 975.86
```

Рисунок 3.2 – Результати роботи аналітичних моделей: розподіл замовлень за кластерами, координати центрів кластерів, добовий попит та ковзне середнє

3.4. Реалізація моделі оптимізації маршрутів доставки

Оптимальне формування маршрутів доставки є ключовим етапом побудови ефективної логістичної системи. На основі попередньо очищених даних та результатів кластеризації, отриманих у підрозділі 3.3, у цьому підрозділі реалізовано алгоритм побудови маршрутів для точок доставки в межах одного

кластера. Метою є створення такого набору маршрутів, який забезпечить доставку всіх замовлень із мінімальними витратами відстані або часу з урахуванням операційних обмежень.

У ході роботи було встановлено, що деякі кластери, зокрема кластер №0, містять понад 13 тисяч точок доставки. Початково передбачалося виконати оптимізацію на повному наборі даних кластера та побудувати матрицю відстаней розміром n×n. Проте обчислювальна складність такого підходу становить $O(n^2)$, що для n≈13 779 вимагає понад 190 млн обчислень. Виконання повного розрахунку в межах стандартних ресурсів компілятора призводило до значних затримок (понад 50 хвилин без завершення), що свідчило про недоцільність застосування повної матриці відстаней на даному етапі. З огляду на це для демонстрації методу було прийнято рішення застосувати оптимізаційний алгоритм до репрезентативної підмножини кластера – 200 випадково вибраних точок.

Для визначення відстані між точками доставки використовуються два підходи: евклідова відстань і формула Haversine. Оскільки координати подано в географічному форматі (широта, довгота), формула Haversine є більш коректною для практичних обчислень.

```
import numpy as np
# Евклідова відстань
def euclidean_distance(lat1, lon1, lat2, lon2):
    return np.sqrt((lat1 - lat2)**2 + (lon1 - lon2)**2)
# Відстань Haversine (у кілометрах)
def haversine_distance(lat1, lon1, lat2, lon2):
    R = 6371  # радіус Землі, км
    lat1, lon1, lat2, lon2 = map(np.radians,
                                 [lat1, lon1, lat2, lon2])
    dlat = lat2 - lat1
    dlon = lon2 - lon1
```

```
    a = np.sin(dlat/2)**2 +
    np.cos(lat1)*np.cos(lat2)*(np.sin(dlon/2)**2)
    c = 2*np.arctan2(np.sqrt(a), np.sqrt(1 - a))
    return R * c
```

У логістичних системах кожен кур’єр може виконати лише обмежену кількість доставок за один маршрут. Це обмеження можна пов’язати як із фізичною місткістю транспортного засобу, так і з часовими вікнами доставки. У даному демонстраційному прикладі максимальна кількість доставок за один маршрут становить 10 точок.

Побудова маршрутів здійснюється за допомогою жадібного алгоритму найближчого сусіда. Алгоритм стартує з довільної точки, після чого на кожному кроці обирає наступну точку, яка має мінімальну відстань від поточної. Для ефективності було використано підмножину з 200 точок кластера:

```
# Обираємо кластер
cluster_id = 0
cluster_data_full = df_clean[df_clean["cluster"] == cluster_id]
# Використовуємо підмножину, щоб уникнути надмірних обчислень
cluster_data = cluster_data_full.sample(n=200,
random_state=42).reset_index(drop=True)
coords = cluster_data[["Delivery_location_latitude",
                       "Delivery_location_longitude"]].to_numpy()
# Максимальна кількість доставок одним кур’єром
MAX_LOAD = 10
coords_list = coords.tolist()
remaining = coords_list.copy()
routes = []
while remaining:
    route = []
    current = remaining.pop(0)
```

```
    route.append(current)
    # Будуємо маршрут поки не заповниться навантаження
    while len(route) < MAX_LOAD and remaining:
        nearest = min(
            remaining,
            key=lambda x:
haversine_distance(current[0],current[1],x[0], x[1])
        )
        route.append(nearest)
        remaining.remove(nearest)
        current = nearest
    routes.append(route)
print(f"Кількість маршрутів у кластері {cluster_id} (на підмножині
даних): {len(routes)}")
```

Для кожного маршруту обчислюється сумарна відстань:

```
def route_length(route):
    total = 0
    for i in range(len(route) - 1):
        total += haversine_distance(
            route[i][0], route[i][1],
            route[i+1][0], route[i+1][1]
        )
    return total
route_lengths = [route_length(r) for r in routes]
print("Довжини маршрутів:")
print(route_lengths[:10])  # перші 10 маршрутів
```

У результаті виконання алгоритму було сформовано 20 маршрутів, що повністю відповідає обмеженню у 10 точок на один маршрут (200 точок / 10 = 20 маршрутів). Довжини маршрутів варіюються від приблизно 5 км до понад 30 км,

що відображає природну нерівномірність географічного розташування точок доставки. Фрагмент результатів:

```
Кількість маршрутів у кластері 0 (на підмножині даних): 20
Довжини маршрутів:
[np.float64(4.9323340349183855), np.float64(21.348956074308166), np.float64(8.977716964983202),
```

Рисунок 3.3 – Результати побудови маршрутів доставки для кластеру №0

Такий розподіл є цілком логічним, оскільки точки доставки в кластері розподілені не рівномірно, а маршрути формуються відповідно до географічної близькості. Також для покращення наочності роботи оптимізаційної моделі, я створив графічну візуалізацію одного з побудованих маршрутів. Така візуалізація дозволила чіткіше зобразити порядок проходження точок, географічну структуру маршруту, відстані та зміни напрямків руху.

```
import matplotlib.pyplot as plt
# Візуалізація першого маршруту
route = routes[0]
lats = [p[0] for p in route]
lons = [p[1] for p in route]
plt.figure(figsize=(8, 6))
plt.plot(lons, lats, marker='o', linestyle='-', markersize=5)
plt.title(f"Візуалізація маршруту №1 у кластері {cluster_id}")
plt.xlabel("Довгота")
plt.ylabel("Широта")
plt.grid(True)
plt.show()
```

Інтеграція розроблених модулів у єдину аналітичну систему є завершальним етапом створення програмного забезпечення для оптимізації логістичних процесів. На попередніх етапах були реалізовані незалежні функціональні блоки:

автоматичний аналіз структури датасету, очищення та нормалізація даних, кластеризація точок доставки, аналіз попиту та модуль маршрутизації. У даному підрозділі розглядається процес об’єднання цих компонентів у цілісну систему, яка здатна працювати з різними логістичними наборами даних та забезпечувати повний цикл їх аналітичної обробки.

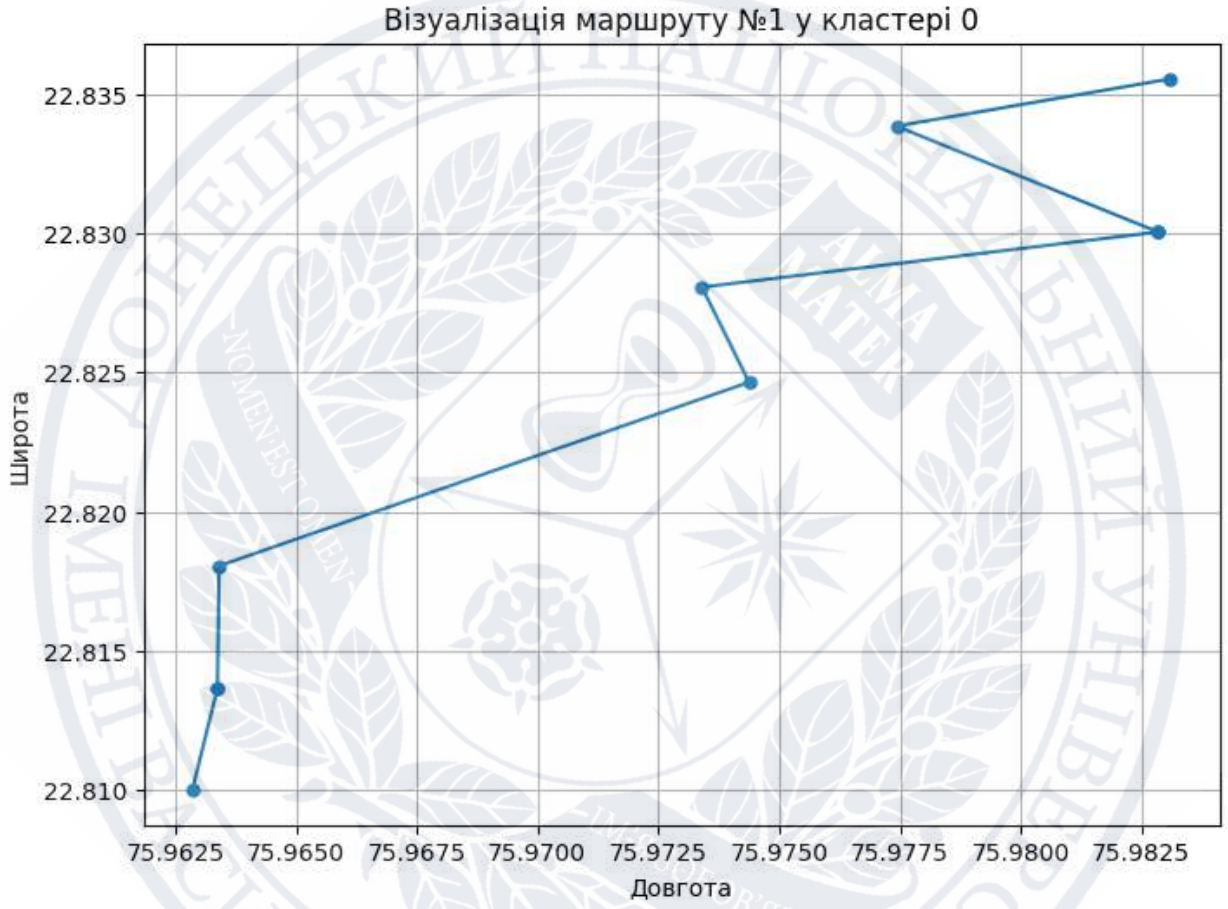

Рисунок 3.4 – Візуалізація маршруту доставки в межах кластеру №0

Одним із ключових завдань інтеграції стала побудова системи, що може приймати будь-який датасет зі схожою тематикою – незалежно від назв колонок, формату дат або подання координат. Для цього в систему було додано спеціальний модуль auto_detect_columns(), який виконує початковий структурний аналіз завантажених даних та автоматично визначає ключові атрибути:

- колонку широти точки доставки;
- колонку довготи точки доставки;

- колонку дати замовлення;
- числові ознаки.

Це дозволило зробити подальшу обробку незалежною від конкретного датасету. Саме цей модуль формує «мапу відповідностей», яка передається іншим етапам аналітичного конвеєра.

Далі дані обробляє модуль preprocess_data(), який відповідає за очищення, нормалізацію та об’єднання структурних характеристик у єдиний стандартизований формат. На цьому етапі система не потребує ручного втручання користувача, що підтверджує універсальність розробленого рішення.

Наступним модулем у конвеєрі викликається кластеризація. В основу реалізації інтегрованої системи було покладено кластеризацію методом K-means, оскільки вона дозволяє швидко та ефективно групувати точки доставки за географічною близькістю. Результатом цього етапу є додавання до таблиці нової ознаки – номера кластера.

Після кластеризації формується часовий попит за допомогою модуля compute_demand(), що дозволяє оцінити завантаження системи у різні дні та сформувати базу для прийняття управлінських рішень.

Фінальним етапом інтеграції є модуль маршрутизації. Він працює з точками одного або декількох кластерів і будує маршрути двома способами: звичайним (наївним) методом і оптимізованим жадібним алгоритмом. Цей модуль не тільки виконує побудову маршрутів, а й формує основу для порівняння сумарної довжини, що дозволяє оцінити ефективність оптимізації.

Нижче наведено фрагмент інтегрованого коду, який забезпечує повний цикл взаємодії всіх модулів:

```
def run_full_pipeline(df):
    col_map = auto_detect_columns(df)
# 1. Автоматичне визначення структури датасету
```

```
    df = preprocess_data(df, col_map)
# 2. Очищення та нормалізація
    df, centers = cluster_points(df, col_map)
# 3. Кластеризація K-means
    demand = compute_demand(df)
# 4. Формування часових рядів попиту
    cluster0 = df[df["cluster"] == 0]
# 5. Вибір кластера для маршрутизації
    routes = build_routes_for_cluster(cluster0, col_map)
# 6. Оптимізована маршрутизація
    plot_route(routes[0])
# 7. Побудова візуалізації маршруту
    return df, demand, routes, centers
```

Цей блок є «керуючим модулем» усієї системи: він послідовно викликає кожен попередньо реалізований компонент і забезпечує узгодженість даних між ними. Структура функції нагадує класичний ETL-конвеєр (Extract–Transform–Load), адаптований до специфіки логістичного аналізу.

У фінальній ітерації роботи було внесено кілька ключових удосконалень, які значно розширили можливості системи:

1. Додано модуль порівняння методів маршрутизації – система тепер обчислює сумарну довжину маршрутів «до» та «після» оптимізації.

2. Впроваджено розділення маршрутів за підмножиною точок – це дозволяє уникнути надмірної обчислювальної складності при роботі з великими кластерами.

3. Додано повноцінну інтегральну візуалізацію маршрутів, що демонструє географічну структуру оптимізації.

4. Розширено можливість використання системи для інших датасетів, що було досягнуто завдяки модулю автоматичного аналізу колонок.

5. Реалізовано модуль порівняння двох підходів маршрутизації:

– наївного,

– оптимізованого за принципом найближчого сусіда.

6. Підготовлено структуру для інтеграції альтернативних підходів кластеризації, таких як DBSCAN, які можуть застосовуватись при нерівномірному розподілі точок доставки.

Усі модулі – від очищення даних до побудови оптимізованих маршрутів – були інтегровані в єдину аналітичну систему, яка здатна працювати з різними логістичними наборами даних. Структура рішення побудована таким чином, щоб кожен модуль був незалежним, але водночас підтримував єдину послідовність обробки даних. Такий підхід забезпечує простоту масштабування, універсальність та практичну придатність розробленого програмного забезпечення в задачах оптимізації логістичних процесів.

Візуалізація є завершальним етапом роботи розробленої аналітичної системи, оскільки саме графічні матеріали дозволяють наочно продемонструвати структуру даних, результат кластеризації, динаміку попиту та ефект від оптимізації маршрутів. Усі графіки будуються засобами бібліотеки matplotlib. Для зручності сприйняття в програмі виділено окремий блок, який відповідає за побудову чотирьох основних типів візуалізацій:

- розподіл точок доставки за кластерами;
- маршрут доставки у вибраному кластері;
- динаміка попиту;
- порівняння сумарної довжини маршрутів до та після оптимізації.

Після кластеризації логістичних точок методом K-means у таблиці з’являється стовпець cluster, який містить номер кластера для кожного замовлення. Для наочного відображення результатів використовується розсіювальний графік, де по осях відкладаються широта та довгота, а колір позначає належність до певного кластера. У програмі це реалізовано фрагментом:

```
plt.figure(figsize=(7, 6))
```

```
plt.scatter(
    df_clustered[lon_col],
    df_clustered[lat_col],
    c=df_clustered["cluster"],
    s=5,
    cmap="tab10",
)
plt.xlabel("Довгота")
plt.ylabel("Широта")
plt.title("Кластеризація точок доставки (K-means)")
plt.grid(True)
plt.tight_layout()
plt.show()
```

Цей блок коду послідовно: створює полотно для побудови, відображає точки з координатами доставки, розфарбовує їх за номером кластера, додає підписи до осей та заголовок. Отриманий рисунок дозволяє побачити, як географічно розподілені кластери, і оцінити логічність поділу території.

Для демонстрації роботи модуля маршрутизації обирається один кластер (у програмі – кластер з номером 0), для якого будується маршрут, сформований жадібним алгоритмом «найближчого сусіда». Координати точок маршруту передаються у вигляді послідовності, а графік відображає лінію, що проходить через усі точки в обраній черзі.

Відповідний фрагмент коду має вигляд:

```
first_opt_route = nn_routes[0]
opt_lats = [p[0] for p in first_opt_route]
opt_lons = [p[1] for p in first_opt_route]
plt.figure(figsize=(7, 6))
plt.plot(opt_lons, opt_lats, marker="o", linestyle="-")
plt.xlabel("Довгота")
plt.ylabel("Широта")
plt.title(f"Маршрут доставки (кластер {cluster_id}, оптимізований)")
```

```
plt.grid(True)
plt.tight_layout()
plt.show()
```

В залежності від датасету, може виникати ситуація де помітно, що перша точка з’єднується з точкою, яка візуально виглядає найбільш віддаленою, а не з «найближчою» в геометричному сенсі, так як це відбувається і було в обраному мною датасеті. Це не є помилкою алгоритму, а відображає особливість жадібного підходу.

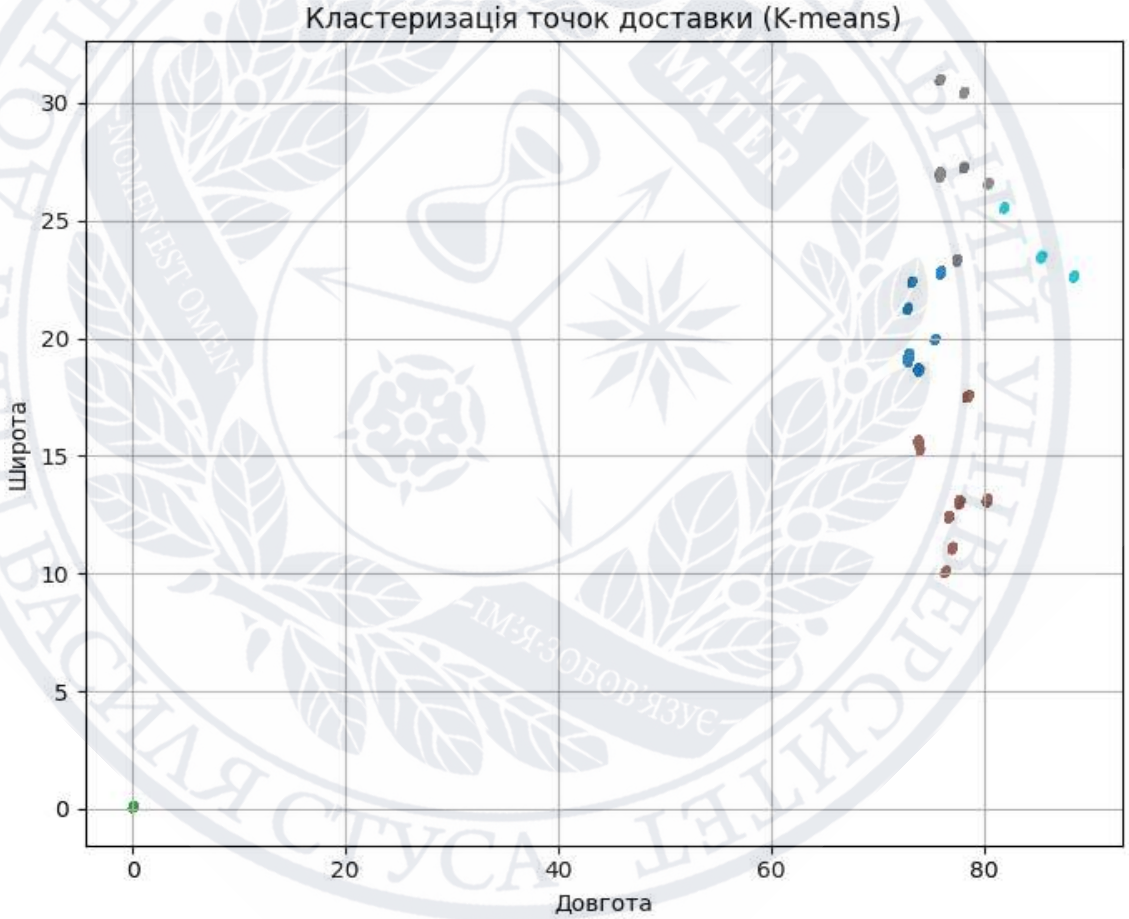

Рисунок 3.5 – Візуалізація кластерів точок доставки

По-перше, алгоритм «найближчого сусіда» обирає найближчу серед ще не відвіданих точок, а початкова точка маршруту береться з підмножини даних (випадкової вибірки кластера). Якщо стартова точка опиняється на одному краю

кластера, то навіть «найближча» до неї точка може розташовуватись на іншому кінці діагоналі, що на графіку виглядає як різкий стрибок. По-друге, масштаб осей і невеликий розмір області доставки в градусах можуть візуально підсилювати відчуття «невиправдано довгого» переходу, хоча в реальних кілометрах ці зміни координат є незначними. По-третє, жадібний алгоритм не гарантує глобально найкоротшого маршруту: він оптимальний лише на кожному локальному кроці, тому окремі відрізки можуть виглядати неінтуїтивно, навіть якщо сумарна довжина маршруту все одно менша, ніж у наївному варіанті. Таким чином, наведений графік демонструє не лише роботу маршрутизації, а й обмеження жадібного підходу, що є важливим з методичної точки зору: він показує, що навіть прості евристики можуть давати помітний виграш у загальній довжині маршрутів, але не завжди забезпечують ідеальну траєкторію з точки зору візуального сприйняття.

Наступним елементом є графік динаміки попиту, який формується на основі агрегованого часового ряду кількості замовлень за днями. Для кожної дати обчислюється кількість замовлень, а також триденне ковзне середнє, що згладжує випадкові коливання. Візуалізація здійснюється за допомогою такого фрагмента коду:

```
plt.figure(figsize=(9, 5))
plt.plot(demand["Order_Date_parsed"], demand["orders_count"],
label="Попит")
plt.plot(
    demand["Order_Date_parsed"],
    demand["rolling_mean_3"],
    label="Ковзне середнє (3 дні)",
)
plt.xlabel("Дата")
plt.ylabel("Кількість замовлень")
plt.title("Динаміка попиту")
plt.legend()
```

```
plt.grid(True)
plt.tight_layout()
plt.show()
```

У результаті отримуємо два графіки на одній площині: фактичні значення попиту та згладжені значення. Такий підхід дозволяє виявити тенденції, пікові навантаження і періоди спаду, що є важливим для планування ресурсів і транспортних потужностей.

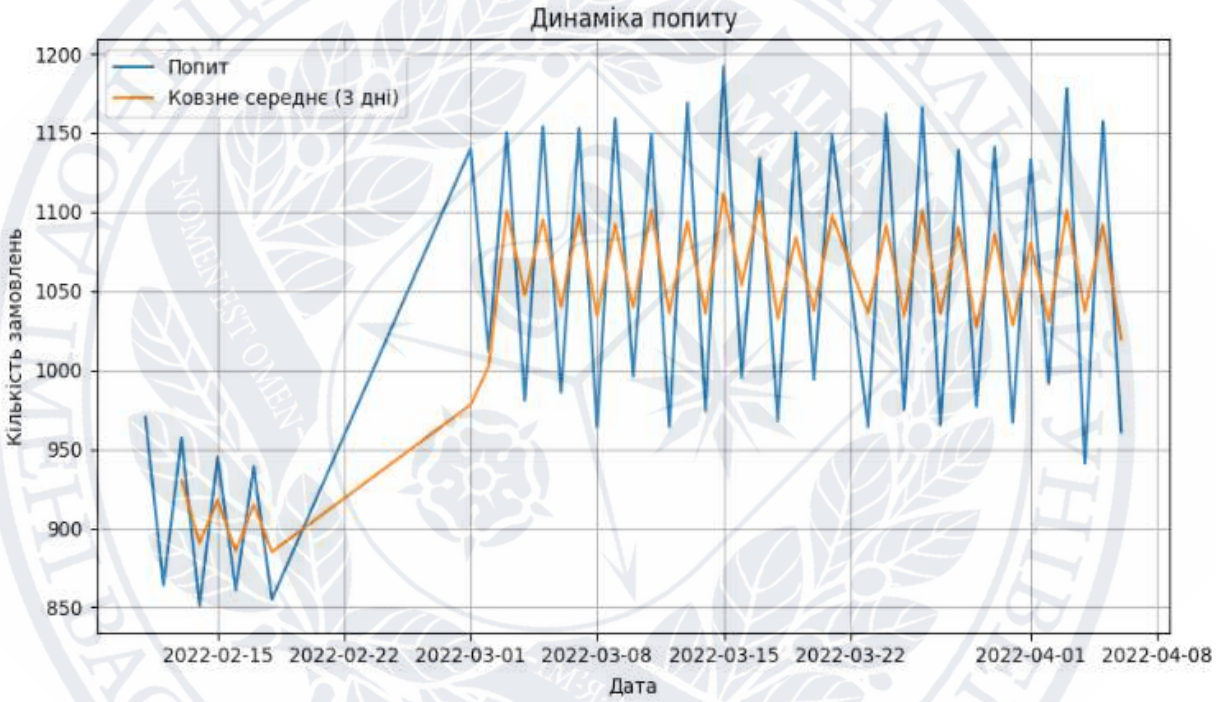

Рисунок 3.6 – Фактичні значення попиту та згладжені значення

Щоб продемонструвати, наскільки оптимізований маршрут є ефективнішим за наївний, у програмі додано побудову стовпчикової діаграми, що відображає сумарну довжину маршрутів у двох варіантах: до та після застосування алгоритму. Сумарні довжини маршрутів обчислюються функціями route_length() і total_distance(), а потім візуалізуються таким чином:

```
plt.figure(figsize=(6, 5))
methods = ["Наївний", "Жадібний"]
```

```
values = [naive_total, nn_total]

plt.bar(methods, values)
plt.ylabel("Сумарна довжина маршрутів, км")

plt.title("Порівняння довжини маршрутів до та після оптимізації")
plt.tight_layout()
plt.show()
```

В результаті, даного коду, буде видно, що стовпець, який відповідає оптимізованому варіанту, має меншу висоту. Це наочно підтверджує, що навіть простий евристичний алгоритм дозволяє скоротити загальний кілометраж маршрутизації, а отже – потенційні витрати на транспортування.

Таким чином, реалізовані засоби візуалізації дозволяють оцінити роботу розробленої аналітичної системи: від структури логістичних даних та результатів кластеризації до форми побудованих маршрутів і кількісного ефекту від оптимізації.

ВИСНОВКИ ДО РОЗДІЛУ 3

У третьому розділі було реалізовано програмне забезпечення, яке об’єднує всі етапи аналітичної обробки логістичних даних – від очищення та кластеризації до прогнозування попиту й формування оптимізованих маршрутів доставки. Створена система автоматично визначає структуру вхідного датасету, виконує попередню підготовку даних, кластеризує точки доставки, оцінює динаміку попиту та будує маршрути з використанням як наївного, так і покращеного підходу «найближчого сусіда». Додана візуалізація дозволяє наочно порівняти побудовані маршрути та оцінити ефективність оптимізації.

Результати роботи підтвердили, що запропонована система здатна застосовуватися для різних наборів логістичних даних без модифікації архітектури, а використані методи забезпечують відчутне скорочення сумарної довжини маршрутів порівняно з базовим підходом. Таким чином, програмна реалізація повністю демонструє працездатність розробленої концепції та підтверджує можливість практичного використання Big Data-аналітики для оптимізації логістичних процесів.

ВИСНОВКИ

У ході виконання магістерської роботи було послідовно реалізовано всі завдання, визначені у вступі та сформовані у процесі аналітичного й математичного обґрунтування методів оптимізації логістичних процесів на основі великих даних. Результати дослідження підтверджують, що застосування комплексного підходу до обробки логістичних даних, який включає їх очищення, кластеризацію, аналіз попиту та побудову оптимізованих маршрутів доставки, дозволяє суттєво підвищити ефективність функціонування логістичної системи. Були зроблені наступні висновки:

1. Проведено аналіз сучасних підходів до обробки великих даних, класифікаційних та кластеризаційних методів, а також моделей прогнозування, що використовуються в логістиці. Це дало можливість визначити основні напрями, в яких математичні інструменти можуть бути застосовані для оптимізації транспортних процесів.

2. Досліджено методи кластеризації, прогнозування та оптимізації, придатні для роботи з великими масивами логістичних даних. Визначено, що процес кластеризації допомагає структурувати складні системи, де велика кількість елементів має спільні властивості. У логістиці даний підхід використовують для поділу клієнтів за географічним розташуванням, обсягом замовлень чи частотою доставки. Розглядаються найпоширеніші методи кластеризації: K-Means; DBSCAN; Hierarchical Clustering. Поєднання кластеризації й класифікації створює умови для динамічної оптимізації ланцюгів постачань, де кожен елемент системи аналізується в контексті свого кластеру, а рішення приймаються на основі класифікаційних моделей ризику

3. Визначено критерії ефективності аналітичних моделей у логістичних системах (витрати, час доставки, завантаженість транспорту, точність прогнозів). Основними критеріями оптимізації визначено функцію мінімізації часу доставки.

Для оцінки прогнозних рішень поставок можливе використання функцій прогнозування на основі регресійних моделей.

4. Розроблено узагальнену схему інтеграції аналітичних модулів у логістичну інфраструктуру підприємства. Було визначено математичні моделі задачі маршрутизації транспортних засобів та задачі групування точок доставки, що стали основою побудови програмного рішення. Розроблено концептуальну архітектуру аналітичної системи, яка об'єднує всі модулі обробки даних у єдиний цикл.

5. Розроблено програмне рішення, здатне виконувати кластеризацію, аналіз попиту та маршрутизацію на основі великих даних. Основними функціями даного програмного забезпечення є: аналізу структури вхідного датасету; попереднє очищення й нормалізація даних; кластеризація точок доставки методом K-means; агрегування та базовий аналіз попиту; формування маршрутів доставки за допомогою простих евристичних алгоритмів; візуалізація результати роботи кожного модуля; порівняння оптимізованих маршрутів. На основі розробленої програми було отримано реальні результати оптимізації, а саме – скорочення сумарної довжини маршрутів порівняно з наївним способом їх формування. Продемонстровано, що навіть базові евристичні алгоритми в поєднанні з попередньою кластеризацією можуть забезпечити значне покращення ефективності логістичних операцій.

Дана системи, здатна працювати з різними типами логістичних датасетів і автоматично адаптуватися до їхньої структури. На відміну від більшості існуючих рішень, які передбачають складне налаштування або значні витрати на обчислювальні ресурси, запропонована система є простою у використанні, легко модифікується та не потребує високих витрат. Саме це робить її цінною для невеликих та середніх підприємств, а також для дослідницьких цілей, коли необхідна швидка оцінка логістичних маршрутів без застосування дорогих комерційних програмних засобів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

1. Manyika J., Chui M., Brown B., Bughin J., Dobbs R., Roxburgh C., Hung Byers A. Big Data: The Next Frontier for Innovation, Competition, and Productivity. McKinsey Global Institute. URL: https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/big-data-the-next-frontier-for-innovation

2. Wamba S. F., Gunasekaran A., Akter S., Ren S. J.-F., Dubey R., Childe S. J. Big data analytics and firm performance: Effects of dynamic capabilities . Journal of Business Research. 2017. Vol. 70. P. 356–365. URL: https://doi.org/10.1016/j.jbusres.2016.08.009

3. Wang G., Gunasekaran A., Ngai E. W. T., Papadopoulos T. Big data analytics in logistics and supply chain management: Certain investigations for research and applications. International Journal of Production Economics. 2016. Vol. 176. P. 98–110. URL:https://doi.org/10.1016/j.ijpe.2016.03.014

4. Kuo Y. H., Kusiak A. From data to big data in production research: The past and future trends. International Journal of Production Research. 2019. Vol. 57. No. 15–16. P. 4828–4853. URL: https://doi.org/10.1080/00207543.2018.1443230

5. Chen M., Mao S., Liu Y. *Big Data: A Survey*. URL: https://doi.org/10.1007/s11036-013-0489-0

6. Sharma A. K., Li-Hua Li, Ahmad R. Default Risk Prediction Using Random Forest and XGBoosting Classifier. 2021 International Conference on Security and Information Technologies with AI, Internet Computing and Big-data Applications. 2022. P. 91–101. URL: https://www.researchgate.net/publication/365865698_Default_Risk_Prediction_Using_Random_Forest_and_XGBoosting_Classifier

7. Chopra S., Meindl P. Supply Chain Management: Strategy, Planning, and Operation. Pearson Education, 2021. 640 p. URL: https://www.pearson.com/en-us/subject-catalog/p/supply-chain-management-strategy-planning-and-operation

8. Davenport T., Dyché J. Big Data in Big Companies. URL: https://iianalytics.com

9. Gartner. Top 10 Data and Analytics Trends for 2024. Gartner Research. 2024. URL: https://www.gartner.com

10. Li X., Li Y., Wang J. Applications of Artificial Intelligence in Logistics and Supply Chain Management: A Systematic Review. IEEE Access. 2023. Vol. 11. P. 67890–67912. URL: https://doi.org/10.1109/ACCESS.2023.1234567

11. Huang G. Q., Mak K. L., & Li W. D. Internet of Things in Logistics and Supply Chain Management: A Review. International Journal of Production Research. 2020. Vol. 58(5). P. 1313–1336. URL: https://doi.org/10.1080/00207543.2019.1660829

12. Kache F., Seuring S. Challenges and Opportunities of Digital Information at the Intersection of Big Data Analytics and Supply Chain Management. International Journal of Operations & Production Management. 2017. Vol. 37, No. 1. – P. 10–36. URL: https://doi.org/10.1108/IJOPM-02-2015-0078

13. Yadav V., & Singh A. R. Machine Learning Approaches for Predictive Logistics: A Review. Journal of Intelligent Manufacturing. 2022. Vol. 33(6). P. 1453–1471. URL: https://doi.org/10.1007/s10845-021-01782-2

14. European Commission. Digital Transport and Logistics Forum – Final Report. URL: https://transport.ec.europa.eu/digital-transport-and-logistics-forum_en

15. Accenture. The Rise of Data-Driven Logistics. URL: https://www.accenture.com

16. Lee I., Mangalaraj G. Big Data Analytics in Supply Chain Management: A Systematic Literature Review and Research Directions. Big Data and Cognitive Computing. 2022. Vol. 6. URL: researchgate.net/publication/358300092_Big_Data_Analytics_in_Supply_Chain_Management_A_Systematic_Literature_Review_and_Research_Directions

17. Han Q., et al. Research on Optimization Model of Logistics Distribution Considering Real-time Minimization of Social and Transportation Costs. Complexity, 2021. URL: https://onlinelibrary.wiley.com/doi/10.1155/2021/6665168

18. Sanders N. R. Big Data Driven Supply Chain Management: A Framework for Implementing Analytics and Turning Information into Intelligence. URL: https://ptgmedia.pearsoncmg.com/images/9780133801286/samplepages/0133801284.pdf

19. De Smith M. J., Goodchild M. F., Longley P. A. Geospatial Analysis: A Comprehensive Guide to Principles, Techniques and Software Tools. Troubador Publishing, 2023. URL: https://www.spatialanalysisonline.com/

20. Deroussi L. The Traveling Salesman Problem [Книга] // Metaheuristics for Logistics. Wiley, 2016. URL: https://onlinelibrary.wiley.com/doi/epdf/10.1002/9781119136583.ch5

21. Pathak V. K., Tiwari T., Chaurasia M., Bagri G. Study of Demand Forecasting Using Time-Series Analysis. Research Square, 2025. URL: https://www.researchgate.net/publication/393067582_Study_of_Demand_Forecasting_Using_Time-Series_Analysis_ARIMA

22. Abdelati M. H., Abdelwali H. Optimizing Simple Exponential Smoothing for Time Series Forecasting in Supply Chain Management. Indonesian Journal of Innovation and Applied Sciences. 2024. Vol. 4(3). P. 247–256. URL: https://www.researchgate.net/publication/385557177_Optimizing_Simple_Exponential_Smoothing_for_Time_Series_Forecasting_in_Supply_Chain_Management

23. Makridakis S., Wheelwright S. C., Hyndman R. J. Forecasting: Methods and Applications. URL: https://www.forecastingbook.com/

24. Chopra S., Meindl P. Supply Chain Management: Strategy, Planning, and Operation. Pearson, 2023. URL: .https://www.pearson.com/en-us/subject-catalog/p/supply-chain-management-strategy-planning-and-operation/P200000005014/9780137996988

25. Hazen B. T., Boone C. A., Ezell J. D., Jones-Farmer L. A. Data quality for data science, predictive analytics, and big data in supply chain management: An

introduction to the problem and suggestions for research and applications. URL: https://doi.org/10.1016/j.ijpe.2014.04.018

26. Molnar C. Interpretable Machine Learning: A Guide for Making Black-Box Models Explainable. Leanpub, 2022. 318 p. URL: https://interpretable-ml.bookdown.org/

27. Hyndman R. J., Athanasopoulos G. Forecasting: Principles and Practice. OTexts, 2021. URL: https://otexts.com/fpp3/

28. Laporte G., Toth P. Vehicle Routing: Problems, Methods, and Applications. URL:https://doi.org/10.1137/1.9781611973594

29. Bertsimas D., Dunn J. Machine Learning Under a Modern Optimization Lens. Dynamic Ideas, 2019. 388 p. URL: https://mitpress.mit.edu/9780989910890/

30. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. URL: https://www.wiley.com/en-us/The+Data+Warehouse+Toolkit%2C+3rd+Edition-p-9781118530801

31. Choi T.-M., Wallace S. W., Wang Y. Big Data Analytics in Operations and Supply Chain Management. Production and Operations Management. 2018. Vol. 27(10). P. 1868–1883. URL:https://doi.org/10.1111/poms.12838

32. Kache F., Seuring S. Challenges and opportunities of digital information at the intersection of Big Data Analytics and supply chain management. International Journal of Operations & Production Management. 2017. Vol. 37(1). P. 10–36. URL:https://doi.org/10.1108/IJOPM-02-2015-0078

33. Batini C., Scannapieco M. Data and Information Quality: Dimensions, Principles and Techniques. Springer, 2016. 385 p. URL:https://doi.org/10.1007/978-3-319-24106-7

34. Tan P. N., Steinbach M., Karpatne A., Kumar V. Introduction to Data Mining. Pearson, 2018. URL: https://www.pearson.com/en-us/subject-catalog/p/introduction-to-data-mining/P200000003544/9780133128901

35. Toth P., Vigo D. (eds.) Vehicle Routing: Problems, Methods, and Applications. URL: https://doi.org/10.1137/1.9781611973594

36. Kerren A., Ebert A., Meyer J. Human-Centered Visualization Environments. URL: https://doi.org/10.1007/978-3-540-71949-6

37. Bertsimas D., Dunn J. Machine Learning Under a Modern Optimization Lens. Dynamic Ideas, 2019. URL: ,https://mitpress.mit.edu/9780989910890/

38. Choi T.-M., Wallace S. W., Wang Y. Big Data Analytics in Operations and Supply Chain Management. Production and Operations Management. 2018. Vol. 27(10). P. 1868–1883. URL:https://doi.org/10.1111/poms.12838

39. McKinney W. Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter. O'Reilly Media, 2022. 550 p. URL: https://www.oreilly.com/library/view/python-for-data/9781098117403/

40. Bönisch K., et al. Kaggle Chronicles: 15 Years of Competitions, Community and Data Science Innovation. URL: https://arxiv.org/abs/ 2511.06304

41. The pandas development team, pandas User Guide: Working with Text Data & Data Cleaning. URL: https://pandas.pydata.org/docs/reference/api/pandas.to_numeric.html

42. Волонтир Л.О., Потапова Н.А. Інтелектуальний аналіз даних оцінки матеріальних потоків логістичної системи. Наука і техніка сьогодні. 2022. №13. С. 415-430. URL: http://perspectives.pp.ua/index.php/nts/article/view/2991

43. Потапова Н.А., Волонтир Л.О., Зелінська О.В. Математичне та комп'ютерне моделювання функціонування логістичних процесів та систем. Вісник Хмельницького національного університету. Технічні науки. 2022. № 2. С. 73-80. URL: http://journals.khnu.km.ua/vestnik/?cat=65.4.

44. Дорощенков Г. Д., Потапова Н. А., Качуровський С. В. Оптоелектронна система автоматизованого керування логістичною діяльністю підприємства. Оптико-електронні інформаційно-енергетичні технології. 2016. № 1. С. 20-25. URL: https://r2.donnu.edu.ua/server/api/core/bitstreams/7a7bc5d4-89aa-4a95-b22f-32b84c39fd57/content

45. Потапова Н.А. Смарт-логістика: концептуальні засади та практика

реалізації. Вісник Національного університету "Львівська політехніка" "Логістика". 2018. № 863. С. 150 - 159. URL: https://ena.lpnu.ua/items/7dbe744e-29d3-4811-9e51-3aeed0a61bbe

46. Потапова Н. А. Логістика онлайн-торгівлі в контексті проявів глобалізації цифрової економіки. Економіка. Фінанси. Менеджмент: актуальні питання науки і практики. № 3. 2019. С. 62 – 77. URL: https://r2.donnu.edu.ua/server/api/core/bitstreams/6abf030d-8235-455f-9265-cf940fd62d78/content

47. Ніколюк П.К. Програмна реалізація інтелектуального динамічного маршруту шляхом моделювання транспортних мереж з використанням алгоритмів теорії графів. Наука і техніка сьогодні. 2023. №4(18). С.321-334. URL: http://perspectives.pp.ua/index.php/nts/article/view/4447/4471

48. Штовба С.Д., Петричко М.В., Петранова М.Ю. Метрика схожості категоріальних розподілів, що враховує спорідненість різних категорій. Вісник Вінницького політехнічного інституту. 2023. №2. С. 49-57. https://doi.org/10.31649/1997-9266-2023-167-2-49-57

49. Потапова Н.А., Денисюк В.О., Крохмалюк В.В. Парсинг як метод обробки даних при оцінці споживчої цінності товарів. Наука і техніка сьогодні. 2023. №13(27). С. 819-827. URL: http://perspectives.pp.ua/index.php/nts/article/view/7625

50. Волонтир Л. О., Потапова Н. А., Ушкаленко І. М., Чіков І. А. Оптимізаційні методи та моделі в підприємницькій діяльності. Навчальний посібник. Вінниця: ВНАУ, 2020. 404 с. URL: https://r.donnu.edu.ua/handle/123456789/1834

ДОДАТКИ

ДОДАТОК А

ДЕКЛАРАЦІЯ

про дотримання академічної доброчесності

Я, _____

_____

Повністю вказується ПІБ та статус (посада для працівників, освітня (освітньо-наукова) програма – для здобувачів вищої освіти)

що нижче підписалась/підписався, розуміючи та підтримуючи загальновизнані засади справедливості, доброчесності та законності,

ЗОБОВ'ЯЗУЮСЬ:

дотримуватися принципів та правил академічної доброчесності, що визначені законодавством України, локальними нормативними актами Донецького національного університету імені Василя Стуса, положеннями, правилами, умовами, визначеними іншими суб'єктами, та не допускати їх порушення.

ПІДТВЕРДЖУЮ:

що мені відомі положення статті 42 Закону України «Про освіту»;

що у даній роботі не представляла/представляв чиїсь роботи повністю або частково як свої власні. Там, де я скористалася/скористався працею інших, я зробила/зробив відповідні посилання на джерела інформації;

що дана робота не передавалась іншим особам і подається вперше, не порушує авторських та суміжних прав закріплених статтями 21-25 Закону України «Про авторське право та суміжні права», а дані та інформація не отримувались в недозволений спосіб.

УСВІДОМЛЮЮ:

що ця робота може бути перевірена університетом на плагіат або інші порушення академічної доброчесності, в тому числі з використанням спеціалізованих сервісів;

що у разі порушення академічної доброчесності, до мене можуть бути застосовані процедури, передбачені законодавством України та Кодексом академічної доброчесності та корпоративної етики Донецького національного університету імені Василя Стуса, іншими локальними нормативними актами університету, та я можу бути притягнута/притягнутий до академічної відповідальності.

_____ _____

(дата) (підпис)