

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ

Факультет автоматизації інформаційних
технологій

Кафедра інформаційних технологій
проекування та прикладної математики

(назва випускової кафедри)

ПОЯСНЮВАЛЬНА ЗАПИСКА

ДО КВАЛІФІКАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТР

на тему:

СИСТЕМА ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРИЙНЯТТЯ РІШЕНЬ У
ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ

Дружкін Єгор Сергійович

Київ 2024 р.

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ

Факультет автоматизації інформаційних технологій

Кафедра інформаційних технологій проектування та
прикладної математики

ЗАТВЕРДЖУЮ

Завідувач кафедри ІТППМ

Олександр ТЕРЕНТЬЄВ

„___” _____ 2024 року

ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО КВАЛІФІКАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ НА ЗДОБУТТЯ
ОСВІТНЬОГО СТУПЕНЯ МАГІСТР
**СИСТЕМА ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРИЙНЯТТЯ
РІШЕНЬ У ЛОГІСТИЦІ ТА
РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ**

Виконав	
Дружкін Єгор Сергійович	
126 Інформаційні системи та технології (спеціальність)	
Інформаційні системи та технології (освітня програма)	
Групи	ІСТ
Керівник	PhD, Рябчун Ю.В.
Ідентичність підтверджую	

Київ 2024 р.

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І
АРХІТЕКТУРИ**

Факультет:	Автоматизації інформаційних технологій
Випускова кафедра:	Інформаційних технологій проектування та прикладної математики
Освітній ступінь:	Магістр
Спеціальність:	Інформаційні системи та технології
Освітня програма:	Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедри ІТППМ

Олександр ТЕРЕНТЬЄВ

„___” _____ 2024 року

З А В Д А Н Н Я

ДО ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ НА
ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТР

	Дружкіну Єгору Сергійовичу
1. Тема роботи <u>СИСТЕМА ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРИЙНЯТТЯ РІШЕНЬ У ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ</u>	
затверджена наказом ректора КНУБА _____	
2. Керівник роботи	Рябчун Юлія Володимирівна, PhD

3. Строк подання Здобувачем роботи до захисту 25.11.2024

4. Зміст пояснювальної записки за розділами:

- P.1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ
- P.2 РОЗРОБКА МОДЕЛІ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АВТОМАТИЗАЦІЇ ПРОЦЕСУ ПРИЙНЯТТЯ РІШЕНЬ У ЛОГІСТИЦІ
- P.3 РЕАЛІЗАЦІЯ ПРОТОТИПУ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ
- P.4 ОЦІНКА ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОЇ СИСТЕМИ ДЛЯ ПОКРАЩЕННЯ ЛОГІСТИЧНИХ ПРОЦЕСІВ
- P.5 ЕРГОНОМІКА ІТ ТА ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ РОЗРОБКИ. ФОРМУВАННЯ РЕКОМЕНДАЦІЙ ЩОДО ВПРОВАДЖЕННЯ ТА МАСШТАБУВАННЯ РОЗРОБЛЕНОЇ СИСТЕМИ

5. Графічний матеріал за розділами:

Р.1. 14 рисунків

Р.2. 2 рисунки, 1 таблиця

Р.3. 17 рисунків

Р.4. -

Р.5. 14 таблиця

6. Календарний план виконання роботи:

Види робіт та їх зміст	Дата виконання
Розділ 1	
Розділ 2	
Розділ 3	
Розділ 4	
Розділ 5	
Направлення роботи для перевірки на плагіат	
Попередній захист роботи на випусковій кафедрі	
Направлення роботи на рецензування	

7. Консультанти розділів атестаційної випускної роботи

Розділ	Прізвище, ініціали та посада консультанта	Перевірив	
		дата	підпис
Розділ 1			
Розділ 2			
Розділ 3			
Розділ 4			
Розділ 5			

8. Дата видачі завдання

Зав. кафедри			Терентьєв О.О.
	(підпис)		(прізвище та ініціали)
Керівники			Рябчун Ю.В.
	(підпис)		(прізвище та ініціали)
Здобувач			Дружкін Є.С.
	(підпис)		(прізвище та ініціали)

АНОТАЦІЯ

Дружкін Є.С. СИСТЕМА ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРИЙНЯТТЯ РІШЕНЬ У ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ.

Атестаційна випускна робота магістра за спеціальністю 126 «Інформаційні системи та технології», освітня програма «Інформаційні системи та технології». – Київський національний університет будівництва та архітектури. – Київ, 2024.

Дана дипломна робота присвячена розробці системи автоматизації прийняття рішень у логістиці, що використовує штучний інтелект для оптимізації процесів управління ресурсами та прийняття рішень у реальному часі. Робота включає практичну реалізацію основних функціональних можливостей системи, таких як оптимізація транспортних маршрутів, автоматизація управління запасами та аналіз ефективності ресурсів. Експериментальна частина оцінює точність прогнозів, швидкість прийняття рішень та загальну продуктивність системи через тестування на реальних даних. На основі отриманих результатів формуються висновки та рекомендації щодо подальшого вдосконалення системи, підвищення її адаптивності до змін у середовищі та покращення ефективності роботи в логістичних процесах.

Робота викладена на 124 аркушах, містить 3 додатки, 2 таблиці, 34 рисунки, список використаної літератури із 44 найменувань.

Ключові слова: штучний інтелект, логістика, розподіл ресурсів, оптимізація, машинне навчання, автоматизація, інформаційні системи.

SUMMARY

Druzhkin Y. S. Artificial intelligence system for decision-making in logistics and distribution of production resources. Master's thesis for the specialty 126 "Information Systems and Technologies", educational program "Information Systems and Technologies". – Kyiv National University of Construction and Architecture. – Kyiv, 2024.

This thesis is dedicated to the development of an automated decision-making system in logistics that uses artificial intelligence to optimize resource management

processes and real-time decision-making. The work includes the practical implementation of the system's core functionalities, such as demand forecasting, optimization of transportation routes, inventory management automation, and resource efficiency analysis. The experimental part evaluates the accuracy of forecasts, decision-making speed, and overall system performance through testing on real data. Based on the obtained results, conclusions and recommendations are formed regarding the further improvement of the system, enhancing its adaptability to environmental changes, and improving operational efficiency in logistics processes.

The paper is set out on 12 pages, contains 3 appendices, 2 tables, 34 figures, and a list of 44 references.

Keywords: artificial intelligence, logistics, resource allocation, optimization, machine learning, automation, information systems.

ЗМІСТ

ВСТУП	10
Розділ 1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ	14
1.1. Актуальність застосування штучного інтелекту (ШІ) в логістиці та управлінні ресурсами	14
1.2 Огляд технологій штучного інтелекту, що використовуються в логістиці	16
1.2.1 Машинне навчання (ML) та його роль у прогнозуванні та оптимізації процесів.	16
1.2.2 Глибоке навчання (DL) для обробки великих даних та складних системних взаємодій.	23
1.2.3 Інтелектуальні системи для підтримки прийняття рішень (Expert Systems).	26
1.2.4 Застосування роботизованих систем та автоматизації.	29
1.3. Сфери застосування ШІ у логістиці	30
1.4. Приклади успішного впровадження ШІ в логістиці та управлінні ресурсами	34
Розділ 2. РОЗРОБКА МОДЕЛІ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АВТОМАТИЗАЦІЇ ПРОЦЕСУ ПРИЙНЯТТЯ РІШЕНЬ У ЛОГІСТИЦІ	42
2.2 Розробка архітектури моделі штучного інтелекту для автоматизації управління ресурсами	45
2.3 Аналіз алгоритмів оптимізації для управління розподілом виробничих ресурсів	48
2.4 Порівняння та вибір найбільш ефективних алгоритмів оптимізації	51
2.5 Адаптація та інтеграція алгоритмів до моделі системи	53
Розділ 3. РЕАЛІЗАЦІЯ ПРОТОТИПУ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ	57

	8
3.1 Розробка прототипу системи на основі вибраних алгоритмів	57
3.2 Тестування прототипу на реальних або симуляційних даних	72
3.3. Навчання прототипу системи ШІ	75
Розділ 4. ОЦІНКА ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОЇ СИСТЕМИ ДЛЯ ПОКРАЩЕННЯ ЛОГІСТИЧНИХ ПРОЦЕСІВ	78
4.1 Оцінка впливу системи на швидкість прийняття рішень.	78
4.2 Оцінка економічного ефекту та потенційних вигод для підприємства	80
Розділ 5. ЕРГОНОМІКА ІТ ТА ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ РОЗРОБКИ. ФОРМУВАННЯ РЕКОМЕНДАЦІЙ ЩОДО ВПРОВАДЖЕННЯ ТА МАСШТАБУВАННЯ РОЗРОБЛЕНОЇ СИСТЕМИ	83
5.1 Вимоги до програмного забезпечення та основні підходи до його проекткування з погляду користувача	83
5.2 Ергономічні цілі і показники якості програмного продукту	85
5.3 Основні характеристики, що враховуються при розробці інтерфейсу користувача	86
5.4 Техніко-економічне обґрунтування розробки	87
5.5. Опис проєктованого продукту	91
ВИСНОВКИ	113
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	115
ДОДАТОК А. Опис програми	117
ДОДАТОК Б. Керівництво користувача	118
ДОДАТОК В. Код розробки	119

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

ІІІ - штучний інтелект

ML (Machine Learning) - машинне навчання;

DL (Deep Learning) - глибинне навчання;

RFID (Radio Frequency IDentification) - радіочастотна ідентифікація, спосіб автоматичної ідентифікації об'єктів, в якому за допомогою радіосигналів зчитуються або записуються дані, що зберігаються в про транспондерах, або RFID-мітках;

QA (Quality Assurance) - контроль якості;

WMS (warehouse management system) - система керування складом;

ERP (enterprise resource planning) - планування ресурсів підприємства;

TMS (transportation management system) - інформаційна система з вантажоперевезень, також інформаційна система з вантажів та транспорту;

GDPR (General Data Protection Regulation) - загальний регламент із захисту даних;

ISO 27001 - Міжнародний стандарт з інформаційної безпеки;

CRM (Customer Relationship Management) - система управління взаємовідносинами з клієнтами.

ВСТУП

У сучасних умовах глобалізації та динамічного розвитку цифрових та інформаційних технологій, ефективне планування логістичних систем стає невід'ємною складовою успішної діяльності суб'єктів господарювання. Водночас, зростання обсягів та складності логістичних завдань, а також швидкі зміни ринкових умов ставлять перед сучасним бізнесом ряд серйозних викликів. Одним із ключових аспектів даної проблематики в даний час є ефективне використання штучного інтелекту в процесах планування логістичних систем. Це обумовлюється тим, що суттєве зростання обсягів даних та складність сучасних логістичних мереж ставлять під сумнів традиційні методи планування, які не завжди здатні ефективно адаптуватися до змін у реальному часі та враховувати велику кількість варіантів оптимізації ланцюгів постачання. При цьому, недостатній рівень автоматизації та використання сучасних технологій у сфері логістики може призводити до втрат часу, ресурсів та загального погіршення якості обслуговування клієнтів.

Логістика як концепція управління потребує інноваційних інструментів щодо управління потоковою взаємодією в режимі реального часу, з найменшою кількістю помилок, здатністю до адаптації та гнучкої реакції на вимоги ринку. Шляхи та процеси досягнення такої взаємодії досліджують науковці теоретики і практики, спеціалісти з логістики, економіки, інформаційних технологій. Саме штучний інтелект як інноваційна технологія повністю здатний задовольнити потреби логістики, але виникає коло питань, вирішити які можливо лише комплексно, а саме: розроблення адаптованих систем штучного інтелекту, правове забезпечення функціонування цих систем та наявність спеціалістів, які можуть здійснювати управління ланцюгами поставок за допомогою штучного інтелекту.

Мета роботи полягає у розробці ефективної моделі системи штучного інтелекту (ШІ), яка здатна автоматизувати процес прийняття рішень у сфері логістики та управління виробничими ресурсами, з метою підвищення ефективності, зниження витрат та оптимізації управлінських процесів.

Для досягнення цієї мети необхідно:

1. Провести дослідження сучасних методів і технологій ШІ, що застосовуються в логістиці та управлінні ресурсами.
2. Розробити архітектуру системи ШІ для автоматизації логістичних операцій та розподілу ресурсів.
3. Реалізувати та протестувати систему на основі симуляцій або реальних даних.
4. Оцінити ефективність системи та її вплив на процеси управління логістикою і виробничими ресурсами.

Об'єкт дослідження – процеси управління логістикою та розподілом виробничих ресурсів на підприємствах із застосуванням інноваційних технологій, зокрема систем штучного інтелекту.

Предметом дослідження – є методи та алгоритми штучного інтелекту, що застосовуються для автоматизації процесу прийняття рішень у логістиці та управлінні ресурсами, а також розробка системи для їх ефективної інтеграції в ці процеси.

Ці формулювання дозволять точно окреслити область дослідження, зосереджуючись на конкретному аспекті впровадження штучного інтелекту в логістику та виробництво.

Методи дослідження, що будуть використанні в даній роботі:

1. Аналіз літературних джерел та документів
 - Огляд наукової та технічної літератури, статей, досліджень, пов'язаних зі штучним інтелектом, логістикою та управлінням виробничими ресурсами.
 - Аналіз існуючих систем штучного інтелекту в логістиці та їх впливу на бізнес-процеси.
2. Системний аналіз
 - Вивчення логістичних процесів та структури управління виробничими ресурсами на підприємстві для розробки системи прийняття рішень.
 - Аналіз взаємодії між компонентами логістичної системи та оптимізація ресурсних потоків.

3. Математичне моделювання

- Використання математичних моделей для опису логістичних процесів і методів розподілу ресурсів.
- Моделювання сценаріїв для автоматизації прийняття рішень за допомогою алгоритмів штучного інтелекту.

4. Алгоритмічний підхід

- Розробка та тестування алгоритмів штучного інтелекту (машинне навчання, нейронні мережі, генетичні алгоритми тощо) для прийняття рішень у логістиці.
- Створення та оптимізація алгоритмів для ефективного розподілу виробничих ресурсів.

5. Експериментальний метод

- Проведення експериментів із застосуванням розробленої системи штучного інтелекту на реальних або симульованих даних логістичних процесів.
- Тестування ефективності системи, її продуктивності, часу реакції на зміни у виробничих потребах.

6. Методи комп'ютерного моделювання та симуляції

- Використання комп'ютерних програм для симуляції логістичних операцій та розподілу ресурсів у різних умовах.
- Моделювання впливу запропонованої системи на ефективність виробничих і логістичних процесів.

7. Статистичний аналіз

- Збір та обробка даних про функціонування логістичних процесів до і після впровадження системи штучного інтелекту.
- Оцінка ефективності системи за допомогою статистичних методів, наприклад, аналізу продуктивності, витрат часу та ресурсів.

8. Метод порівняльного аналізу

- Порівняння результатів впровадження системи штучного інтелекту з традиційними методами управління логістикою та розподілом ресурсів.
- Оцінка переваг та недоліків різних підходів.

Розділ 1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В ЛОГІСТИЦІ ТА РОЗПОДІЛІ ВИРОБНИЧИХ РЕСУРСІВ

Метою першого розділу буде дослідження та аналіз сучасних методів використання штучного інтелекту (ШІ) в галузі логістики та управління виробничими ресурсами.

Особливу увагу приділено вивченню методів, які сприяють підвищенню ефективності процесів оптимізації витрат і зниженню часу на виконання завдань.

Основним завданням цього розділу буде огляд існуючої наукової літератури та практичних досліджень, які описують впровадження ШІ в логістиці та управлінні ресурсами. Буде систематизовано ці дослідження і створено класифікацію методів ШІ, що застосовуються в різних аспектах логістики, включаючи прогнозування попиту, оптимізацію ланцюгів постачання, управління запасами та маршрутизацію транспортних засобів.

1.1. Актуальність застосування штучного інтелекту (ШІ) в логістиці та управлінні ресурсами

Складування є життєво важливою частиною управління ланцюжком поставок, що є основою, що дозволяє підприємствам забезпечувати безпечне зберігання, організацію та готовність своєї продукції для відправки клієнтам [1]. Однак складське зберігання пов'язане з логістичними проблемами, такими як прогнозування необхідної кількості запасів, визначення найкращого способу організації товарів для швидкого доступу і навіть вирішення фізичних завдань щодо переміщення товарів за складом.

Алгоритми ШІ можуть аналізувати історичні дані про рух продуктів та моделей попиту, щоб розробляти чи пропонувати оптимальні планування складів [2]. Це включає визначення кращих місць для зберігання товарів з урахуванням їх розміру, ваги і частоти доступу, забезпечуючи тим самим логічне розміщення, що дозволяє максимально використовувати простір і мінімізувати час комплектації.

ШІ, безперечно, революційна технологія. Проте думати, що з ШІ можуть отримати вигоду лише корпорації «блакитних фішок» і великі міжнародні технологічні компанії, було б неправильно. Зрештою, штучний інтелект змінить кожен галузь та бізнес.

- Якісна комунікація. Сховища даних залежить від постійного потоку даних. Проте людська помилка може призвести до неефективності та дорогих помилок. Ці помилки можна усунути за допомогою ШІ, який також може надавати точні дані на запит.
- Більш ефективне планування у разі непередбачених обставин.

Проблеми зі складом, які неможливо було передбачити, часто виникають без складних систем. Прогнозуючи закономірності та дозволяючи підприємствам керувати своїми запасами та операціями в мінливих обставинах, ШІ допомагає компаніям підвищити свою готовність до різних непередбачуваних обставин.

ШІ – це передова технологія, яка, безперечно, змінить складські операції. Точність процесів та поточний рівень продуктивності мають бути підвищені за допомогою технологій. ШІ підвищує стандарти безпеки у будівлях та знижує експлуатаційні витрати.

Існує безліч потенційних застосувань ШІ в автоматизації складу (рис.1.1). Компанії повинні досліджувати способи включення та використання їх у своїй повсякденній діяльності.

Основними перевагами використання машинного навчання в прогнозуванні та оптимізації є [3]:

- **Швидкість і точність обробки даних:** ML дозволяє аналізувати величезні обсяги інформації набагато швидше і точніше, ніж це можливо за допомогою традиційних методів.
- **Адаптивність:** ML-моделі здатні навчатися на нових даних і вдосконалювати свої прогнози та рекомендації в реальному часі.
- **Автоматизація:** Багато рутинних процесів у логістиці та управлінні ресурсами можуть бути автоматизовані за допомогою ML, що дозволяє зосередитись на стратегічних завданнях.



Рисунок 1.1 Напрями застосування штучного інтелекту в логістичних системах

1.2 Огляд технологій штучного інтелекту, що використовуються в логістиці

ШІ на складі створює цінність за допомогою різних допоміжних технологій, включаючи машинне навчання, обробку природної мови, робототехніку та комп'ютерний зір.

1.2.1 Машинне навчання (ML) та його роль у прогнозуванні та оптимізації процесів.

Машинне навчання (ML) є однією з ключових технологій ШІ, яка широко використовується в логістиці та управлінні виробничими ресурсами [4]. Його роль у прогнозуванні та оптимізації процесів має вирішальне значення для підвищення ефективності операцій і зменшення витрат.

Машинне навчання використовує алгоритми для навчання на досвіді і прийняття практичних складських рішень. Система виявляє шаблони даних датчиків і рекомендує більш швидке поповнення товарів, яких майже немає, більш короткі маршрути і краще розміщення запасів.

Сьогодні машинне навчання в тренді. Усі знають про штучний інтелект ChatGPT [5], який вміє писати тексти, кодити та будувати логічну аргументацію в бесіді. Усіх вражають ілюстрації, які малює алгоритм MidJourney.

Аналогічно є моделі машинного навчання і для логістичної галузі. Вони можуть, наприклад, аналізувати маршрути транспорту, шукати організаційні та логістичні помилки, прогнозувати ситуацію на дорогах та ринковий попит [6, 7].

Машинне навчання дозволяє будувати високоточні моделі прогнозування, які базуються на аналізі великих обсягів даних [8-11]. У логістиці ці алгоритми можуть використовуватись для передбачення попиту на товари, що дозволяє ефективніше планувати виробничі ресурси та запобігати дефіциту або надлишкам. Наприклад, за допомогою історичних даних про продажі, сезонність і змінні ринкові фактори, моделі машинного навчання можуть передбачати, коли та де попит на певний продукт буде зростати і спадати. Це допомагає компаніям оптимізувати запаси, зменшити витрати на зберігання та уникнути несвоєчасної доставки.

Прогнозування є важливим завданням у різних областях, таких як економіка, фінанси, маркетинг та інші. Методи прогнозування дозволяють передбачати майбутні значення змінних на основі наявних даних. Одним з найбільш ефективними методами прогнозування є метод машинного навчання.

Метод машинного навчання – це підхід до аналізу даних, за якого комп'ютерні системи навчаються на основі досвіду і самостійно покращують свою продуктивність без програмування [12, 13]. У контексті прогнозування моделі машинного навчання використовують наявні дані для побудови математичної моделі, здатної передбачати майбутні значення.

Існує кілька основних типів моделей машинного навчання прогнозування. Один із них - *лінійна регресія* [14]. Лінійна регресія передбачає лінійну залежність між залежною змінною та однією або декількома незалежними змінними. Модель будує лінію або поверхню, яка найкращою чином відповідає наявним даним. Використовуючи цю лінію чи поверхню, можна передбачити майбутні значення залежної змінної на основі значень незалежних змінних.

Крім того, існують інші методи машинного навчання для прогнозування, такі як нейронні мережі та алгоритми кластеризації [15]. Нейронні мережі імітують роботу людського мозку і складаються з безлічі зв'язаних нейронів, які передають інформацію один одному. Алгоритми кластеризації дозволяють групувати дані на основі їх схожості чи відмінності.

Прогнозування попиту також є важливим для управління ланцюгом постачань. ML допомагає передбачити, які матеріали будуть потрібні і в яких обсягах, що дозволяє уникнути затримок у виробництві та оптимізувати роботу з постачальниками. Крім того, прогностичні моделі ML можуть враховувати зовнішні фактори, такі як економічні коливання, зміни в поведінці споживачів, погодні умови та інші змінні, що можуть вплинути на логістику.

Розглянемо основні принципи машинного навчання, які використовуються для прогнозування [16].

1. Навчання із викладачем. Однією з основних технік машинного навчання прогнозування навчання з викладачем. Цей підхід передбачає наявність набору даних, де кожному об'єкту відповідає правильна відповідь або цільова змінна. Модель машинного навчання аналізує ці дані та будує математичну модель, яка може передбачати значення цільової змінної нових спостережень.
2. Вибір алгоритму. Для досягнення добрих результатів у задачі прогнозування необхідно вибрати підходящий алгоритм машинного навчання. Існує широкий спектр алгоритмів, таких як лінійна регресія, дерева рішень, випадковий ліс та нейронні мережі. Кожен алгоритм має свої особливості та припущення, тому вибір повинен ґрунтуватися на характеристиках даних та необхідних цілях прогнозування.
3. Попередня обробка даних. Перед навчанням моделі машинного навчання необхідно провести попередню обробку даних. Це включає в себе видалення викидів, заповнення пропущених значень, масштабування ознак та кодування категоріальних змінних. Чисті та добре підготовлені дані є основою точного прогнозування.

4. Поділ на навчальну та тестову вибірки. З метою оцінки продуктивності моделі після її навчання необхідно поділити дані на дві частини: навчальну вибірку та тестову вибірку. Навчальна вибірка використовується для тренування моделі, а тестова вибірка – для оцінки її точності на нових даних. Це дозволяє перевірити працездатність моделі до її застосування на реальних завданнях прогнозування.
5. Оцінка та оптимізація моделі. Після навчання моделі необхідно оцінити її продуктивність. Для цього використовуються різні метрики, такі як середня абсолютна помилка (MAE), середня квадратична помилка (MSE) коефіцієнт детермінації (R^2). Якщо результати є незадовільними, можна спробувати змінити гіпер параметри моделі або вибрати інший алгоритм.
6. Процес прогнозування. Після успішного навчання моделі можна використовувати прогнозування нових даних. Модель отримує на вхід значення ознак нових об'єктів та видає передбачені значення цільової змінної. Це дозволяє робити прогнози на основі наявних даних та використовувати їх для прийняття рішень.

Інший тип моделі машинного навчання для прогнозування – *дерева рішень* (рис.1.2). Дерева рішень будуються як ієрархічна структура, де кожен вузол представляє собою питання про значення певної змінної. Відповіді на ці питання визначають, який шлях йти по дереву для отримання прогнозу [17 – 21]. Цей тип моделі особливо корисний під час роботи з категоріальними чи бінарними змінними.

Застосування алгоритмів машинного навчання прогнозування має безліч переваг. По-перше, вони здатні вловлювати складні залежності і шаблони даних, які завжди очевидні для людини. Крім того, вони можуть працювати з більшими обсягами даних та обробляти інформацію набагато швидше, ніж традиційні методи прогнозування.

Проте слід зазначити деякі обмеження методу. Наприклад, алгоритми машинного навчання вимагають великих обсягів даних для навчання та тестування

моделей. Крім того, ці алгоритми не завжди здатні давати інтерпретовані результати, що може бути неприйнятним у деяких сферах діяльності.

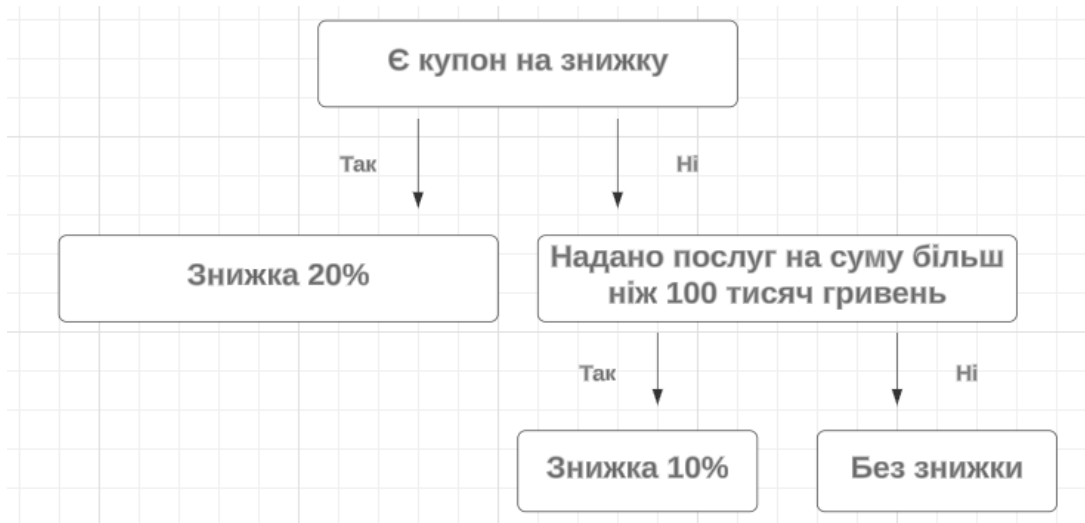


Рисунок 1.2. Приклад дерева рішень

Предиктивна аналітика - ця технологія дозволяє компаніям моніторити наявні товари, аби задовольняти попит покупців навіть за умови скорочення запасів на складах [22, 23]. Це схоже на передбачення майбутнього, але замість гадання на картах чи кавовій гущі, штучний інтелект вибудовує математичну модель вашої нагальної проблеми. Великі масиви даних та численні математичні моделі перевіряються доти, доки серед них не буде виявлено найнадійнішу. Завдяки такому аналізу компанії можуть прогнозувати обмінні курси та попит, швидко коригувати ціни та організувати оперативну доставку товарів.

Згідно з річним звітом МНІ за 2021 рік, 49% провідних операторів ланцюжків постачання прискорити інтеграцію цифрових технологій, зокрема предиктивної аналітики.

Управління складом [24 – 28]. Інтегроване до складських систем, машинне навчання автоматизує ручну роботу, а також прогнозує можливі проблеми у процесах виробництва, зберігання та пакування, що скорочує час обробки товарів. Це дуже важлива функція, адже на склади щодня надходять нові товари, які потрібно систематизувати та зберігати. Машинне навчання аналізує поточне

розташування товарів на складах, і коли привозять нові партії, їх просто додають до попередніх запасів.

Виробництво, пакування, сортування. Машинне навчання здатне розпізнавати проблеми з роботою конвеєра ще на ранніх етапах. За допомогою комп'ютерного зору здійснюється перевірка продукції на відповідність стандартам без потреби застосовувати ручний контроль якості. Завдяки автоматизованому розпізнаванню та аналізу пошкоджень чи дефектів ймовірність надіслати клієнту браковане відправлення знижується до мінімуму.

Алгоритми машинного навчання також використовуються для оптимізації логістичних процесів. Одним із найважливіших напрямків є оптимізація маршрутів транспорту. За допомогою ML можна побудувати моделі, які з урахуванням багатьох факторів — дорожні умови, час доставки, витрати пального, час очікування в пунктах навантаження — пропонують найефективніші маршрути. Це знижує експлуатаційні витрати, скорочує час доставки та підвищує загальну продуктивність.

Задачі ML(машинного навчання). Глобальне завдання машинного навчання — створити штучний інтелект, який за своїми аналітичними здібностями дорівнюватиме або навіть перевершуватиме людський. Це дуже складне завдання, яке наука цілком може вирішити в найближчі 5–10 років.

ШІ, який здатний змагатися з людиною і вирішувати завдання різного типу, називають сильним або загальним (artificial general intelligence). Що стосується більш вузьких завдань машинного навчання, їх можна розділити чотирма групи: класифікація, регресія, кластеризація і зменшення розмірності [29 – 32].

1. *Класифікація.* Використовується для вирішення завдань, де на підставі ознак об'єктів потрібно розподілити їх за заданими категоріями. Наприклад, на виробництві можуть відокремлювати деталі з браком від хороших за допомогою комп'ютерного зору.
2. *Регресія.* Регресія в теорії ймовірностей та математичної статистики - це залежність середнього значення будь-якої величини від деякої іншої величини або від декількох величин. Комп'ютери можуть аналізувати

величезні масиви даних і робити прогнози на їх основі. Наприклад, можна завантажити в комп'ютер дані про рух курсу акцій за останні 10 років і попросити передбачити їх ціну цього року.

3. *Кластеризація* - це розподіл об'єктів за категоріями, коли невідомо, скільки категорій вийде в результаті. Розподіл відбувається за заданим критерієм. Наприклад, компанія може використовувати кластеризацію для визначення типів клієнтів за патернами їх покупок і робити на підставі цього персоналізовані пропозиції товарів.
4. *Зменшення розмірності* - допомагає скоротити кількість ознак даних без втрати інформації. Це спрощує їх обробку і прискорює алгоритми машинного навчання, оскільки кількість даних, з якими вони мають працювати, зменшується. При розпізнаванні зображень зниження розмірності дозволяє не аналізувати кожен піксель, а використовувати лише важливі ознаки. Наприклад, щоб розпізнати зебру серед тварин, головне – навчитися бачити білі та чорні смуги.

Наскрізні процеси машинного навчання впроваджуються у бізнес на всіх рівнях, від аналітики та виробництва до комплектування та транспортування. Зокрема, йдеться про такі технології:

Ще одним прикладом оптимізації є *управління запасами* [33]. ML-моделі можуть автоматично визначати, коли і скільки товарів потрібно поповнювати на складах, щоб уникнути нестачі або перевищення запасів. Це дозволяє більш точно координувати процеси закупівлі та зберігання, мінімізуючи затрати та втрати.

Крім того, машинне навчання активно застосовується в галузі управління виробничими потужностями. Алгоритми здатні аналізувати великі обсяги даних з виробництва та визначати оптимальні режими роботи обладнання, графіки змін або планувати обслуговування з мінімальними зупинками виробничого процесу. Це знижує ризик простоїв та збоїв у виробництві.

1.2.2 Глибоке навчання (DL) для обробки великих даних та складних системних взаємодій.

Глибоке навчання (Deep Learning, DL) [34] є підгалуззю машинного навчання, яка використовує штучні нейронні мережі з багатьма шарами для аналізу великих і складних наборів даних. Завдяки своїй здатності виявляти приховані шаблони та робити прогнози на основі даних, глибоке навчання стало одним із найбільш перспективних напрямків штучного інтелекту.

Якщо поглянути на глибоке навчання під іншим кутом, то глибоке навчання позначає «комп'ютерне моделювання» або «автоматизацію» процесів навчання людиною від джерела (наприклад, зображення собак) до об'єкта навчання (собак). Тому мають сенс поняття, сформульовані як «глибше» (англ. "deeper") та «найглибше» (англ. "deepest") навчання [35]. Найглибше навчання позначає повністю автоматичне навчання від джерела до кінцевого об'єкта навчання. Глибоке навчання відтак позначає змішаний процес навчання: процес навчання людиною від джерела до навченого напів-об'єкта, за яким слідує комп'ютерний процес навчання від навченого людиною напів-об'єкта до кінцевого об'єкта навчання.

Концепція глибокого навчання (Deep Learning - DL) вперше з'явилася у 2006 році як нова галузь досліджень у машинному навчанні. Спочатку воно було відоме як ієрархічне навчання [36], і як правило воно включало в себе безліч областей досліджень, пов'язаних з розпізнаванням образів. Глибоке навчання в основному бере до уваги два ключові фактори: нелінійна обробка в декількох шарах або стадіях і навчання під наглядом або без нього [37 – 39]. Нелінійна обробка в кількох шарах відноситься до алгоритму, в якому поточний шар приймає як вхідні дані вихідні дані попереднього шару. Ієрархія встановлюється між шарами, щоб упорядкувати важливість даних, корисність яких слід встановити. З іншого боку, контрольоване та неконтрольоване навчання пов'язане з міткою класів цілей: її присутність має на увазі контрольовану систему, а відсутність – неконтрольовану.

Глибоке навчання використовує багатошарові нейронні мережі для моделювання складних залежностей у даних. Кожен шар нейронної мережі

трансформує вхідні дані таким чином, щоб наступний шар міг виявити більш складні шаблони. Глибоке навчання застосовується для таких завдань, як:

- класифікація зображень, звуку, тексту;
- розпізнавання образів і об'єктів;
- прогнозування і автоматизація процесів на основі великих даних.

Ці технології вимагають великих обсягів даних та потужних обчислювальних ресурсів, однак їх застосування призводить до високоточних результатів навіть у складних і багатофакторних завданнях.

Глибоке навчання знаходить застосування в логістиці через свою здатність обробляти та аналізувати величезні масиви даних, що виникають у сучасних ланцюгах постачання, транспортних системах і управлінні запасами. Ось кілька прикладів, як DL може бути корисним у цій сфері:

1. Оптимізація ланцюга постачань та управління запасами

Глибокі нейронні мережі можуть аналізувати величезні масиви даних про постачання, продажі, ринкові умови, кліматичні зміни та інші фактори, які впливають на ланцюги постачань. Системи, засновані на DL, можуть передбачати коливання попиту з більшою точністю, ніж традиційні алгоритми, враховуючи приховані залежності між різними факторами.

Також глибоке навчання може аналізувати тенденції у використанні виробничих ресурсів і надавати рекомендації щодо оптимізації запасів, наприклад, зменшуючи витрати на зберігання або забезпечуючи своєчасне поповнення запасів.

2. Оптимізація транспортних маршрутів і управління флотом

Глибоке навчання може використовуватися для аналізу великих обсягів транспортних даних, включаючи поточні дорожні умови, поведінку водіїв, витрати пального, час доставки та інші змінні. Нейронні мережі можуть побудувати прогнозні моделі для оптимізації маршрутів в реальному часі, вибираючи найбільш ефективні шляхи. Це не лише економить час і ресурси, але й знижує викиди шкідливих речовин та підвищує екологічність логістичних операцій.

Наприклад, використання глибокого навчання для розпізнавання реальних дорожніх умов на основі даних з датчиків або камер спостереження дозволяє передбачати затори або інші проблеми на дорозі і негайно коригувати маршрути.

3. Прогнозування збоїв у постачаннях і управління ризиками

Глибоке навчання дозволяє будувати моделі, які передбачають можливі збої у постачаннях, базуючись на аналізі історичних даних та зовнішніх факторів, таких як природні катаклізми, політична нестабільність або зміни у ринковій ситуації. Нейронні мережі можуть виявляти приховані зв'язки між цими факторами та прогнозувати, коли і де можуть виникнути проблеми.

Також такі моделі можуть допомогти компаніям краще управляти ризиками, розробляючи стратегії щодо мінімізації впливу можливих проблем у ланцюзі постачань.

4. Автоматизація процесів на складах

Глибоке навчання активно використовується в роботизованих складах для автоматизації різних процесів, таких як сортування, пакування та перевезення товарів. Камери та сенсори на складах, підключені до систем DL, можуть автоматично розпізнавати товари, відстежувати їхній рух і забезпечувати ефективне управління запасами.

Наприклад, роботизовані системи на базі DL можуть самостійно визначати найкращі шляхи переміщення товарів по складу, враховуючи дані про доступні шляхи, стан запасів та поточні замовлення.

5. Аналіз великих даних (Big Data) для підвищення ефективності

Глибоке навчання чудово працює з великими масивами даних. У контексті логістики та управління ресурсами, де генерується багато різних типів даних (транзакційні, операційні, геолокаційні), DL може допомогти виявити нові інсайти, які неможливо отримати за допомогою традиційних методів аналізу.

Нейронні мережі DL можуть виявляти приховані закономірності в даних [40], що допомагає компаніям краще розуміти свої операції, виявляти нові можливості для оптимізації та приймати більш обґрунтовані рішення.

Переваги глибокого навчання у логістиці та розподілі ресурсів:

- *Автоматизація складних процесів:* глибоке навчання може виконувати складні завдання, які важко автоматизувати за допомогою традиційних алгоритмів.
- *Прогнозування з високою точністю:* DL здатне враховувати величезну кількість змінних, що дає можливість точніше передбачати попит, ризики і збої в процесах.
- *Адаптивність:* системи глибокого навчання можуть навчатися на нових даних і вдосконалювати свої прогнози й оптимізаційні рішення в режимі реального часу.
- *Оптимізація ресурсів:* за допомогою DL можна суттєво підвищити ефективність використання людських, транспортних та виробничих ресурсів.

1.2.3 Інтелектуальні системи для підтримки прийняття рішень (Expert Systems).

Система підтримки прийняття рішень - комп'ютерна автоматизована система, метою якої є допомога людям, які приймають рішення в складних умовах для повного та об'єктивного аналізу предметної діяльності [41]. СППР виникли внаслідок злиття управлінських інформаційних систем та систем управління БД.

Як правило, експертні системи створюються для вирішення практичних завдань у деяких вузькоспеціалізованих галузях, де велику роль відіграють знання «бувалих» фахівців, а також для вирішення низки різних типів проблем у кількох категоріях: інтерпретація, прогнозування, діагностика, проектування, планування, моніторинг, відстеження, відновлення, управління, управління. Прикладними областями для експертних систем є агрономія, хімія, комп'ютерні системи, електроніка, інженерія, геологія, юриспруденція, промисловість, математика, медицина, метеорологія, військова наука, фізика, управління процесами, космічні технології. Експертні системи можуть також використовуватися для навчання у відповідній галузі, використовуючи свій досвід у вирішенні проблеми та їх організованих знань. Експертні системи були першими розробками, які змогли привернути велику увагу до результатів досліджень у галузі штучного інтелекту.

Експертні системи мають одну велику відмінність від інших систем штучного інтелекту: вони не призначені для вирішення якихось універсальних завдань, як нейронні мережі або генетичні алгоритми. Експертні системи призначені для якісного вирішення завдань у певній розробниками області, у поодиноких випадках – областях.

Вони створені для вирішення складних проблем, що вимагають знань спеціалістів, і можуть бути використані для [42, 43]:

- Діагностики проблем.
- Прогнозування результатів.
- Оптимізації процесів.
- Прийняття рішень у складних сценаріях.

Експертна система зазвичай складається з трьох основних компонентів: бази знань, засобів збору та/або обробки вхідних даних та механізму формування висновків (рис.1.3).



Рисунок 1.3. Структура експертної системи

База знань містить безліч знань, специфічних для предметної області, представлених у формі правил, фактів і відносин. Він є основою для реалізації процесу прийняття рішень експертною системою. Засоби збору та обробки вхідних даних дозволяють отримати та перетворити призначені для аналізу дані у формат, який буде придатний для обробки експертною системою. А механізм формування

висновків відповідає за надання висновків експертної системи, заснованих на обробці вхідних даних за допомогою закладених у систему знань.

Експертні системи можуть допомагати автоматизувати та покращити процес прийняття рішень на кожному етапі логістичних і виробничих операцій. Це стосується як стратегічного рівня (довгострокове планування), так і операційного рівня (щоденні операції).

1. Підтримка прийняття рішень щодо оптимізації ланцюга постачань.

Експертні системи можуть аналізувати численні змінні у ланцюгу постачань і допомагати приймати рішення щодо оптимізації процесів постачання товарів і матеріалів. Наприклад, вони можуть рекомендувати, коли і в якій кількості потрібно замовляти товари для запобігання дефіциту або надлишку запасів, враховуючи такі фактори, як час доставки, попит, умови ринку та можливі ризики. Можливості експертної системи в цій галузі можуть включати:

- Автоматичний вибір постачальників на основі критеріїв, таких як вартість, надійність, час доставки.
- Прогнозування попиту на товари та матеріали, щоб оптимізувати запаси.
- Оптимізація управління запасами шляхом визначення оптимальних рівнів запасів і часу поповнення.

2. Розробка інтелектуальних транспортних систем.

Експертні системи можуть грати важливу роль в управлінні транспортними ресурсами. Вони здатні допомагати диспетчерам і логістам у прийнятті рішень щодо оптимальних маршрутів, часу доставки, управління автопарком і витратами на транспорт. Завдяки аналізу різних факторів, таких як погодні умови, дорожні затори, витрати на паливо, експертна система може:

- Рекомендувати найкращі маршрути для доставки вантажів.
- Оптимізувати розклад транспорту для мінімізації простоїв та зменшення витрат.
- Прогнозувати й вирішувати проблеми в дорозі, такі як можливі затори чи поломки транспорту.

3. Управління виробничими ресурсами.

Експертні системи можуть бути використані для прийняття рішень, пов'язаних із розподілом виробничих ресурсів, управлінням обладнанням, плануванням виробництва та управлінням запасами матеріалів. Вони можуть враховувати змінні, як-от доступність матеріалів, графік виробництва, стан обладнання, і надавати рекомендації щодо оптимізації цих процесів. Завдання, які може вирішувати експертна система:

- Планування виробничих процесів на основі аналізу доступних ресурсів та попиту.
- Прогнозування потреб у технічному обслуговуванні обладнання для мінімізації простоїв.
- Оптимізація використання людських ресурсів через автоматичний розподіл робочих змін.

4. Аналіз і управління ризиками.

Оскільки логістичні операції та виробництво пов'язані з численними ризиками (затримки постачання, збої в роботі обладнання, коливання попиту тощо), експертні системи можуть допомагати в їх управлінні. Вони можуть прогнозувати можливі проблеми на основі історичних даних і надавати стратегії мінімізації цих ризиків:

- Оцінка ризиків затримок у постачанні через аналіз минулих даних про збої в ланцюгу постачань.
- Виявлення й управління критичними точками у виробничих процесах, що можуть призвести до збоїв.
- Рекомендації щодо оптимізації ланцюга постачання для зменшення впливу зовнішніх ризиків (наприклад, коливань ринкових цін або політичних змін).

1.2.4 Застосування роботизованих систем та автоматизації.

Використання роботизованих систем у сучасному виробництві є ключовим фактором для забезпечення ефективності та конкурентоспроможності підприємств [43]. Ці системи використовуються для автоматизації рутинних і повторюваних

завдань, що дозволяє знизити витрати на оплату праці, підвищити продуктивність і покращити якість продукції.

Роботизовані системи можуть працювати без перерви та втоми, забезпечуючи стабільний і надійний режим виробництва. Крім того, вони дозволяють виконувати завдання з високою точністю та швидкістю, що призводить до зменшення кількості дефектів та відходів. Крім підвищення продуктивності, роботизовані системи також сприяють підвищенню безпеки праці, зменшуючи ризики для людей у виробничому середовищі. Вони можуть виконувати небезпечні або важкі завдання, такі як підйом важких вантажів чи робота з токсичними речовинами, забезпечуючи безпеку та комфорт працівників. У цілому, роботизовані системи є необхідною складовою для підвищення ефективності виробництва та забезпечення стабільного розвитку підприємств у сучасному світі.

Робототехніка дає ШІ фізичну присутність, просторову обізнаність та здатність рухатися у реальному приміщенні. Можливості роботів зі штучним інтелектом можуть змінюватись від завантаження або розвантаження піддона до переміщення вантажу по складу та виконання операцій зі збору.

1.3. Сфери застосування ШІ у логістиці

Штучний інтелект (ШІ) значно трансформує логістику, автоматизуючи процеси, підвищуючи ефективність та точність рішень. Застосування ШІ в логістиці охоплює різні аспекти, від управління ланцюгами постачань до аналізу даних і прогнозування попиту.

Застосування штучного інтелекту різними сферами на користь бізнесу – питання актуальне, але неоднозначне. З одного боку, на практиці ми бачимо кейси із залучення ШІ в електронну торгівлю – в комунікацію продавця та цільової аудиторії, в кастомізацію досвіду користувача. Наприклад, за допомогою мобільного застосування та технологій штучного інтелекту покупець може приміряти взуття, одяг, ввівши параметри фігури, або інтегрувати предмети для дому (категорії DIY) у дизайн інтер'єру.

З іншого боку, наприклад, у сфері складської логістики коректніше говорити про автоматизацію процесів, ніж використання ШІ в операційних завданнях. Діяльність фулфілмент-операторів будується за суворими алгоритмами, більшість із яких автоматизовані. Однак існує низка операцій, де потрібні профільні компетенції співробітників, їхнє залучення до ситуації "тут і зараз". Подібним етапом, наприклад, є приймання, де співробітник часом повинен зіставити кожен окремий товар із базою, перевірити його на справжність та наявність шлюбу.

Загалом, на сьогоднішній день ШІ знаходить у логістиці різноманітне застосування. Розглянемо основні напрямки.

- *Прогнозування попиту.* Алгоритми машинного навчання та штучного інтелекту можуть аналізувати великі обсяги даних, включаючи історичні продажі, ринкові умови, сезонні коливання та навіть погодні фактори. Це дозволяє більш точно прогнозувати попит на продукцію, що є критичним для ефективного управління запасами та зниження витрат на зберігання.

- *Оптимізація ланцюга постачань.* ШІ допомагає покращити управління ланцюгами постачань (рис.1.4) через автоматизацію процесів планування, моніторингу та управління. Використовуючи прогностичні алгоритми, ШІ дозволяє компаніям:

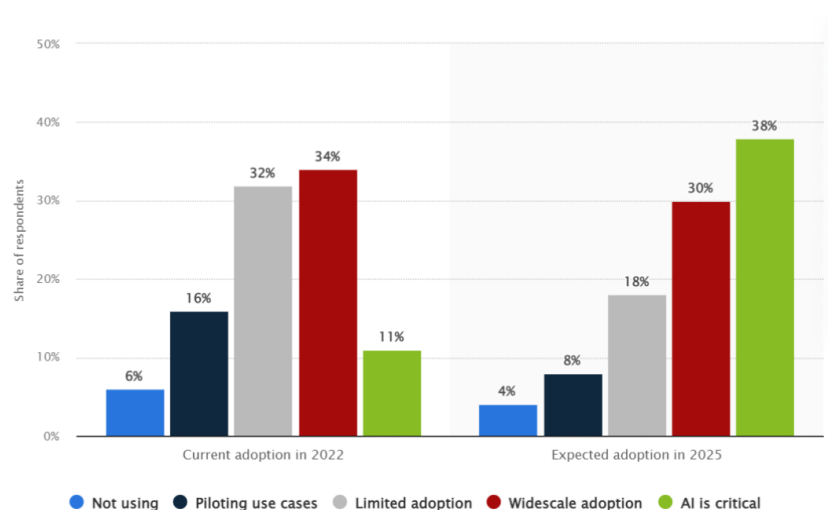


Рисунок 1.4. Світовий рівень впровадження ШІ в механізми регулювання ланцюжків постачання

- Прогнозувати попит на основі аналізу історичних даних і ринкових тенденцій.
- Визначати оптимальні маршрути для транспортування вантажів, враховуючи дорожні умови, погоду та вартість.
- Автоматично вибирати постачальників і оптимізувати графіки доставки.

Технології штучного інтелекту допомагають оптимізувати дистанцію та траєкторію шляхів доставки, а також краще координувати час доставки, зменшуючи його разом із витратами на паливо.

- *Автоматизація складів та центрів обробки* [41]. Використання ШІ для автоматизації складів та центрів обробки замовлень дозволяє підвищити ефективність та зменшити час обробки.

Для реалізації цих та інших процесів використовуються різні технології та джерела даних, такі як:

- збір даних із датчиків, встановлених на транспортних засобах, складах та інших об'єктах логістичної інфраструктури;
- використання міток RFID та інших технологій, що дозволяють відстежувати місцезнаходження вантажів та контролювати їх переміщення;
- використання даних, отриманих від комунікаційних мереж, таких як Інтернет речей (IoT), щоб моніторити обладнання та процеси на відстані;
- аналіз зовнішньої інформації, у тому числі погоди, пробок на дорогах та інших факторів, що впливають на час доставки та роботу ланцюжка поставок;
- використання софту та програм для управління процесами логістики, які інтегрують усі зібрані дані та автоматизують ряд процесів.

- *Розумні дороги* [44]. Об'єднання ШІ з датчиками, RFID мітками, системами супутникового моніторингу та мережами 5G може дати потужний поштовх розвитку розумних доріг. Розумні дороги є компонентом концепції розумних міст, де застосовуються передові інформаційні технології (інтернет речей, хмарні обчислення, великі дані, штучний інтелект) для полегшення

планування, будівництва, управління та обслуговування. Технологія розумних доріг дозволить підвищити безпеку дорожнього руху та скоротити затримки в ланцюжку поставок, спричинені несприятливими погодними умовами та дорожніми інцидентами.

- *Прогнозування та управління ризиками.* Штучний інтелект допомагає прогнозувати можливі збої в ланцюзі постачань через аналіз різних факторів, таких як політичні зміни, природні катастрофи, коливання валют та інші ризики. Це дозволяє компаніям краще управляти ризиками та розробляти стратегії для мінімізації негативних наслідків.

- *Персоналізація обслуговування клієнтів.* ШІ також активно використовується в покращенні взаємодії з клієнтами. Наприклад, чат-боти та голосові асистенти можуть автоматично відповідати на запитання клієнтів, відстежувати їхні замовлення та допомагати вирішувати проблеми. Це знижує навантаження на контактні центри та забезпечує швидший і ефективніший сервіс.

- *Аналіз даних і бізнес-аналітика.* ШІ дозволяє збирати та аналізувати величезні масиви даних, які генеруються в процесі логістичних операцій. Алгоритми машинного навчання можуть виявляти приховані закономірності й робити прогнози, допомагаючи керівникам приймати більш обґрунтовані рішення. Завдяки цьому компанії можуть краще розуміти свої операції та оптимізувати процеси на основі даних.

- *Безпілотний транспорт.* У логістиці дедалі активніше застосовуються автономні транспортні засоби, керовані штучним інтелектом. Вантажівки-безпілотники або дрони можуть здійснювати доставку товарів без участі людини, що знижує витрати на транспорт і підвищує швидкість та точність доставки.

ШІ змінює правила гри в логістиці, надаючи компаніям можливість автоматизувати процеси, підвищувати ефективність та забезпечувати точніше прогнозування. Завдяки цим технологіям компанії можуть краще відповідати на вимоги клієнтів, знижувати витрати та покращувати свої операції на всіх етапах ланцюга постачань.

1.4. Приклади успішного впровадження ШІ в логістиці та управлінні ресурсами

1.4.1 Огляд реальних кейсів застосування ШІ у логістиці

Для виробників автоматизація складів стає відповідним рішенням для мінімізації ручної праці та зниження експлуатаційних витрат. Автоматизоване зберігання даних дозволяє компаніям швидше обробляти замовлення та точніше планувати графік.

Якщо в ШІ є щось потужне, то він, безумовно, може обробляти та аналізувати великі дані — величезні обсяги інформації, які наш людський розум не може засвоїти, — в режимі реального часу. По-перше, ШІ вивчає ці великі набори даних, які можуть включати обсяг минулих покупок, що переважають погодні умови, поточні тенденції в соціальних мережах та будь-які важливі майбутні події, які можуть змінити переваги споживачів.

Amazon (рис.1.5) - яскравий приклад компанії, що використовує алгоритми штучного інтелекту для революції в прогнозуванні попиту. Використовуючи Amazon Forecast, сервіс, заснований на машинному навчанні, Amazon може прогнозувати майбутній попит на мільйони продуктів у всьому світі за лічені секунди.



Рисунок 1.5. Компанія Amazon

Віртуальні полиці Amazon заповнені достатньою кількістю товарів, оскільки ШІ передбачає, скільки товарів буде продано. А оскільки вони краще розуміють, що і де знадобиться, Amazon може переміщати товари ближче до місця проживання клієнтів ще до того, як вони натиснуть «купити».

Презентація «технологій майбутнього» Toyota (рис.1.6) на виставці CeMAT привернула серйозну увагу відвідувачів, включаючи концепції «дронів для

піддонів» і «рій» – обидва натякали на форму майбутнього горизонтального транспорту. Цього року Toyota виводить це мислення на вищий рівень, передбачаючи спосіб, за допомогою якого горизонтальні транспортери можуть безпосередньо спілкуватися з машинами високого рівня за допомогою штучного інтелекту. Ідея полягає в тому, що всі переміщення мають бути оптимізовані, за допомогою правильного типу вантажно-розвантажувальних машин, які розгортаються для кожного завдання, передаючи вантажі між машинами за потреби.



Рисунок 1.6. Toyota

В 2018 році автомобілебудівна компанія Toyota представила на виставці CeMAT концепцію AI Team Logistics. Концепція пропонує модернізувати спосіб взаємодії у реальному часі горизонтальних перевізників із високорівневими машинами під управлінням штучного інтелекту. Основна ідея полягає в оптимізації переміщення всіх вантажівок на складі та у забезпеченні їх взаємозв'язку. «Штучний інтелект, заснований на мільйонах обчислень щосекунди, дозволить вантажівкам визначати оптимальний маршрут і використовувати ресурси», — пояснює Борис Анберг, ключовий член команди досліджень і розробок Toyota, який керував проектом. «Завдяки використанню такого типу прямого зв'язку це означає, що традиційне управління центральною системою управління складом більше не буде потрібно – це буде лише питання командної роботи машини».

Концепція безпосередньо відображає ощадливий підхід Toyota до логістичних операцій, заснований на шановній виробничій системі Toyota. Ключовим принципом ощадливого мислення є усунення відходів, і екстремальна оптимізація ресурсів сприяє досягненню цієї мети.

Кожну секунду програмна система на основі ШІ обчислює оптимальне використання ресурсів та маршрути перевізників. Така пряма взаємодія між транспортом замінює традиційну централізовану концепцію складу командної роботи машин. Підтримуючи повністю автоматизований безперервний потік, компанія прагне оптимізувати процеси, що виконуються на складах.

Поряд з історією ATTeamLogistics, Toyota також представляє наступний етап розробки концепції Swarm, який працює на стенді CeMAT. Використовуючи традиційне керування WMS, блоки Swarm уже розробляються для співпраці з системами обробки багажу Vanderlande.

Ocado (рис.1.7) - роботизована технологія поєднує новаторське комп'ютерне бачення, глибоке навчання з підкріпленням і розширені датчики для автоматизації процесу збирання та пакування!



Рисунок 1.7. Компанія Ocado

Роботизовані руки приймають розумні рішення на льоту, щільно пакуючи продукти в пакети, не знаючи, що це за продукти чи що буде далі.

Завдяки плавній інтеграції з технологією The Hive замовлення клієнтів можуть виконуватися без допомоги людини.

Вулик. У центрі компанії центрів обслуговування клієнтів (CFC) працюють групи ботів, які працюють у 3D-системі зберігання, що містить тисячі продуктових товарів. Ми називаємо це найсучасніше виконання «Вулик»!

Система управління «повітряним рухом» штучного інтелекту керує рухом флоту, спілкуючись з кожним ботом 10 разів на секунду, щоб забезпечити максимальну продуктивність. Рухаючись зі швидкістю до 4 м/с, боти

співпрацюють, щоб транспортувати контейнери з продуктовими продуктами, які будуть упаковані в сумки клієнтів роботами або людьми.

Роботизована технологія Nive дозволяє нашим партнерам отримувати замовлення з 50 товарів лише за п'ять хвилин.

Компанія експрес-доставки **STO Express** використовує роботів для сортування своїх посилок (рис.1.8). Робот, оснащений камерою, що пересувається самостійно, завдяки оптичному розпізнаванню здатний для швидкої ідентифікації сканувати етикетки товарів та їх характеристики (вага, розміри, географія та адреса доставки). Завдяки заздалегідь визначеній системі сортування, роботи автоматично направляють посилки до місць призначення.



Рисунок 1.8. Компанія STO Express

Завдяки роботам-сортувальникам, які працюють спільно зі штучним інтелектом, STO Express обробляють 18 000 посилок на годину. Також у компанії підвищилась ефективність, точність та безпека процесу сортування, що додатково знизило трудовитрати на 70%.

Щоб вирішити зазначені проблеми та мінімізувати ризики, компанія **L'Oréal** (рис.1.9), наприклад, запровадила безпілотну систему інвентаризації

L'ORÉAL[®]
PARIS

Рисунок 1.9. Компанія L'Oreal

Дрон, оснащений бортовою камерою, пролітає повз стелажів по кожній позиції та ярусу для проведення інвентаризації. Завдяки обробці відео за допомогою штучного інтелекту дрон може зчитувати штрих-коди, розпізнавати порожні місця, враховувати висоту шарів і визначати, де закінчився один осередок і почався інший.

Маркетплейс **Cdiscount** (рис.1.10) впровадив у себе роботів із ШІ. Завдяки роботизованій системі відбувається оптимізація комплектації замовлень. На складі є парк із сотні роботів, які можуть рухатися в трьох напрямках. Роботи можуть пересуватися не тільки землею, але й підніматися на висоту. Завдяки цій технології Cdiscount збільшили ємність складу у п'ять разів. Продуктивність ж збільшилася втричі-вчетверо, ніж при ручному управлінні.



Рисунок 1.10. Cdiscount

Компанія **DB Schenker** (рис.1.11) у рамках пілотного проекту у своєму розподільчому центрі в Лейпцигу проводить випробування автономних роботів, які поєднують штучний інтелект зі стереоскопічними камерами. У компанії розповіли, що автономні логістичні роботи Gideon Brothers орієнтуються у просторі за допомогою стереоскопічних камер.



Рисунок 1.11. DB SCHENKER

Використовуючи ШІ, вони створюють та зберігають карту навколишнього простору та безпечно обходять співробітників, стаціонарні перешкоди та інше вантажно-розвантажувальне обладнання. При цьому дані машини здатні

переміщати вантажі масою до 800 кг і мають систему швидкої заміни акумулятора, яка суттєво скорочує час простою.

ШІ гарантує онлайн-покупцям швидше виконувати замовлення. Тонким місцем тут є складання відправлень, коли товари можуть бути далеко один від одного, і доводиться багато пересуватися. Тепер за допомогою програми IDEA можна обчислити оптимальний маршрут для формування замовлення.

Враховуючи величезну кількість факторів програма, що самонавчається, робить це за лічені секунди. Більше того, комп'ютер переглядає майбутні замовлення, і при повторях того самого продукту повідомляє співробітника про необхідну його кількість. Програма IDEA вже використовується в логістичних центрах для формування замовлень для великих інтернет-магазинів.

За словами доктора Клауса Дорманна, ШІ здатний відстежувати цілі ланцюжки поставок. Така програма вже використовується у філіях DHL (рис.1.12). Вона стежить за чітким виконанням запланованих маршрутів і дає сигнал, якщо в ньому намічаються збої. При цьому ШІ нічого не варто перенаправити вантаж.



Рисунок 1.12. Логістична компанія DHL

Walmart(рис.1.13) запровадила ШІ, щоб революціонізувати свої процеси управління запасами. Гігант роздрібної торгівлі використовує ШІ для аналізу даних про продажі, сезонних тенденцій та інших факторів для точного прогнозування попиту на товари. Ця функція прогнозування допомагає Walmart підтримувати оптимальний рівень запасів, знижуючи витрати, пов'язані із затоварюванням та нестачею товару.



Рисунок 1.13 Walmart

Крім того, Walmart використовує роботів на базі ШІ у своїх магазинах для сканування полиць та моніторингу рівня запасів у режимі реального часу. Ці роботи можуть виявляти товари, яких немає на складі, і повідомляти персонал про поповнення, гарантуючи, що полиці завжди заповнені продуктами, які потрібні покупцям. Інтегруючи ШІ в управління запасами, Walmart підвищує операційну ефективність та покращує якість покупок для своїх клієнтів.

IBM пропонує низку рішень на основі ШІ, адаптованих для логістичної галузі. Однією з примітних пропозицій компанії є IBM Watson Supply Chain, в якому ШІ використовується для підвищення наочності ланцюжка поставок та прийняття рішень.

Watson (рис.1.14) використовує обробку природної мови та машинне навчання для аналізу величезних обсягів даних з різних джерел, таких як зведення погоди, соціальні мережі та ринкові тенденції.



Рисунок 1.14. IBM Watson

Цей аналіз дає цінну інформацію, яка допомагає компаніям передбачати збої, оптимізувати маршрути та підвищити загальну стійкість ланцюжка постачання. Крім того, рішення IBM у галузі штучного інтелекту дозволяють відстежувати постачання в режимі реального часу, підвищуючи прозорість і дозволяючи більш точно оцінювати постачання. Ці можливості дозволяють логістичним компаніям приймати рішення на основі даних, підвищувати операційну ефективність та покращувати обслуговування своїх клієнтів.

1.4.2 Результати та вплив на виробничі процеси і логістичні ланцюги

ШІ може бути використаний для автоматизації базових логістичних процесів, таких як обробка замовлень, відстеження постачань та планування маршрутів. Це сприяє зниженню витрат та підвищенню продуктивності підприємства, що особливо важливо для соціальних підприємств, котрі функціонують в рамках обмеження ресурсів.

Крім того ШІ забезпечує ефективний аналіз великих обсягів даних та може допомогти в точному прогнозуванні попиту на товари та послуги, що є важливим для запобігання формування надмірного або недостатнього товарного запасу. Таким чином реалізується оптимізація запасів та зниження витрат соціального підприємства. При цьому штучний інтелект дозволяє розробляти найбільш оптимальні маршрути для доставки товарів, що сприяє зменшенню транспортних витрат та скороченню вуглецевих викидів, що є важливим для соціальних підприємств, котрі орієнтуються на екологічну відповідальність.

Загалом ШІ забезпечує зростання ефективності логістичних процесів завдяки аналізу великих обсягів даних та автоматизації. Він допомагає вдосконалити прогнозування попиту на товари, оптимізувати маршрути доставки, автоматизувати обробку замовлень і виявляти аномалії в ланцюгах постачання. Також ШІ може впроваджувати машинне навчання для постійного вдосконалення логістичних процесів, що призводить до зменшення витрат, підвищення точності та швидкості реакції на зміни, що, в свою чергу, сприяє зростанню ефективності логістичних операцій підприємства.

Отже, ШІ знаходить застосування практично на усіх етапах логістичних процесів формування ланцюгів постачання. При цьому базою його впровадження є засоби автоматизації, котрі давно застосовуються в управлінні логістикою підприємств. Відповідно, можна стверджувати, що цифрові та інформаційні технології, на основі яких реалізується ШІ, забезпечують підвищення ефективності управління логістичними системами за допомогою використання спеціалізованих програмних додатків, що функціонують за допомогою штучного інтелекту.

Розділ 2. РОЗРОБКА МОДЕЛІ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АВТОМАТИЗАЦІЇ ПРОЦЕСУ ПРИЙНЯТТЯ РІШЕНЬ У ЛОГІСТИЦІ

У сучасному бізнес-середовищі логістика є однією з найважливіших сфер для підвищення ефективності та зниження витрат. З огляду на зростаючі обсяги даних, кількість взаємодій між постачальниками, транспортними компаніями та клієнтами, а також складність логістичних процесів, впровадження технологій ШІ стає необхідністю. ШІ дозволяє автоматизувати процеси прийняття рішень, оптимізувати маршрути, покращувати управління запасами та передбачати ризики.

2.1 Визначення критеріїв та вимог до системи

Основні критерії для системи ШІ у логістиці:

- Точність прогнозування
 - Система повинна забезпечувати високу точність прогнозів щодо попиту, запасів, термінів доставки та ризиків. Моделі машинного навчання, використовувані в системі, мають навчатися на великій кількості даних для досягнення максимальної точності.
 - Наприклад, прогнозування попиту з похибкою менше 5% вважається високою точністю.
- Швидкість обробки та прийняття рішень
 - Швидкість обробки даних і прийняття рішень у режимі реального часу є критично важливою для оптимізації маршрутів, управління запасами та швидкого реагування на зміни. Система має забезпечувати миттєву обробку інформації, отриманої через IoT, GPS або інші канали.
 - Оптимальний час реагування на зміни в умовах транспортування чи складу — від кількох секунд до хвилин.
- Масштабованість
 - Система ШІ повинна підтримувати масштабування для роботи з великим обсягом даних та кількістю учасників логістичних процесів (склади, постачальники, транспортні компанії). Це включає можливість обробляти тисячі

транзакцій одночасно і забезпечувати безперебійну роботу у великих ланцюгах постачання.

- Адаптивність

- Модель ШІ має бути адаптивною до змінних факторів, таких як коливання попиту, дорожні умови або зовнішні ризики (наприклад, погодні умови, геополітичні ризики). Система повинна коригувати свої алгоритми на основі нових даних без суттєвої затримки.

- Інтегрованість

- Система повинна мати здатність інтегруватися з іншими інформаційними системами підприємств (ERP, WMS, TMS) для обміну даними та забезпечення узгодженості між різними відділами. Важливо, щоб система ШІ могла працювати з різними форматами даних і протоколами передачі даних.

- Прогнозування ризиків

- Система повинна ідентифікувати та передбачати потенційні ризики на основі аналізу історичних даних та поточних умов (затримки в ланцюгу постачання, погодні умови, збої транспорту тощо). Це може включати сценарії моделювання для визначення можливих впливів і прийняття запобіжних рішень.

- Користувацький інтерфейс та легкість використання

- Система повинна мати зручний та інтуїтивно зрозумілий інтерфейс, який дозволить користувачам без спеціальної технічної підготовки ефективно працювати з нею. Важливо забезпечити швидкий доступ до ключових функцій, аналітики та управління даними.

- Безпека та конфіденційність

- Забезпечення безпеки даних є критичною вимогою до системи. Система повинна підтримувати шифрування даних, контроль доступу та інші механізми захисту, зокрема відповідність стандартам безпеки (GDPR, ISO 27001).

Основні функціональні вимоги до системи ШІ у логістиці

1. Оптимізація транспортних маршрутів

- Система має будувати та оптимізувати маршрути для транспортування з урахуванням актуальних даних про стан доріг, вантажопотоків та геолокації

транспортних засобів. Повинні враховуватися витрати на паливо, час доставки, обмеження за вагою та типом вантажу.

2. Автоматизація управління запасами

- ШІ повинен допомагати прогнозувати рівень запасів і пропонувати оптимальні стратегії для їх поповнення або оптимізації. Це має включати автоматичні замовлення матеріалів на основі прогнозів попиту та рівня наявних ресурсів.

3. Моніторинг в реальному часі

- Система повинна мати можливість безперервного моніторингу всіх логістичних операцій в реальному часі: від стану транспортних засобів до умов зберігання на складах. Дані мають надходити з різних сенсорів, IoT-пристроїв та систем управління.

4. Прогнозування попиту та управління ланцюгами постачання

- ШІ повинен аналізувати дані про продажі, сезонні коливання та інші фактори для передбачення змін попиту та пропонувати відповідні дії. Це допоможе покращити керування ланцюгами постачання і уникнути ситуацій надлишку або дефіциту товарів.

5. Автоматизація процесів прийняття рішень

- ШІ повинен забезпечувати автоматизацію прийняття ключових рішень на основі аналітики великих даних, допомагаючи менеджерам приймати оперативні рішення без залучення значних людських ресурсів.

6. Аналіз ефективності та пропозиції щодо оптимізації

- Система повинна надавати регулярні звіти та аналітичні рекомендації щодо покращення процесів: зменшення витрат, прискорення доставки, підвищення ефективності роботи транспорту та складів.

Нефункціональні вимоги

1. Висока доступність і надійність

- Система повинна працювати з високою доступністю (99,9%) і бути стійкою до збоїв або аварій. Забезпечення безперебійної роботи є критичним для великих логістичних операцій.

2. Гнучкість налаштувань

- Система повинна мати можливість гнучкої конфігурації під потреби конкретних компаній або ринків, забезпечуючи адаптацію до специфічних логістичних сценаріїв.

3. Можливість інтеграції

- ШІ має підтримувати стандартизовані інтерфейси для легкої інтеграції з іншими системами управління, базами даних та платформами прогнозування.

2.2 Розробка архітектури моделі штучного інтелекту для автоматизації управління ресурсами

Архітектура моделі ШІ для автоматизації управління ресурсами має бути гнучкою, масштабованою та інтегрованою із бізнес-системами підприємства. Основна мета цієї моделі — забезпечити ефективний розподіл виробничих, транспортних та складських ресурсів, зменшення витрат і підвищення продуктивності. Модель має підтримувати роботу в реальному часі та враховувати складність логістичних процесів на підприємстві (рис.2.1).

Основні принципи побудови архітектури

1. Модульність: система складається з окремих модулів, які виконують специфічні функції (прогнозування, оптимізація, управління).
2. Інтеграція: можливість підключення до ERP-систем, WMS-систем, IoT-пристроїв та інших бізнес-рішень.
3. Адаптивність: система повинна навчатися на нових даних та адаптуватися до змін у логістичних процесах.
4. Реал-тайм режим: прийняття рішень на основі актуальних даних у режимі реального часу.

Основні компоненти архітектури моделі ШІ (рис.2.2)

1. Модуль збору та обробки даних
 - Отримання даних із внутрішніх джерел (ERP, CRM, WMS) та зовнішніх (GPS-трекери, сенсори IoT).

- Попередня обробка даних: очищення, нормалізація та агрегація для подальшого аналізу.

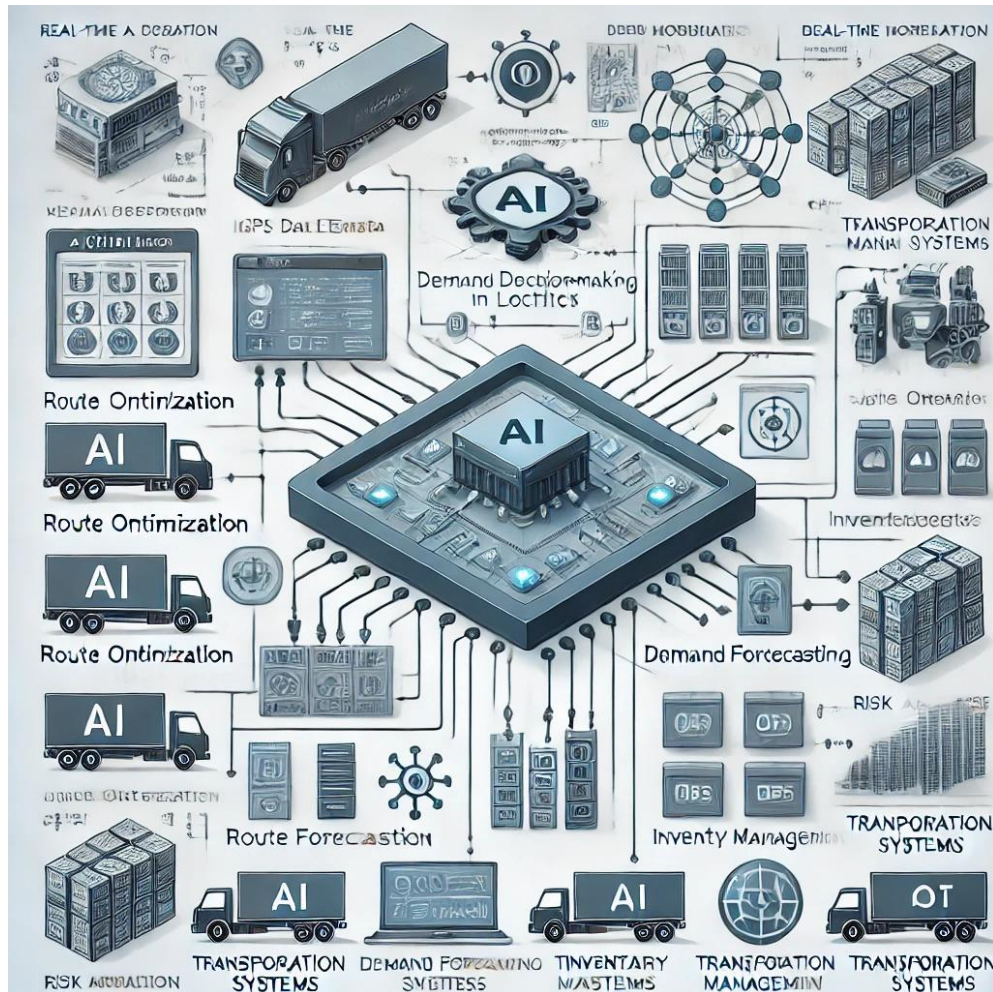


Рисунок 2.1. Схема архітектури моделі системи штучного інтелекту для автоматизації процесу прийняття рішень у логістиці

2. Аналітичний модуль та модуль прогнозування
 - Застосування алгоритмів машинного навчання для аналізу історичних даних.
 - Прогнозування попиту на ресурси, обсягів виробництва та оптимальних графіків доставки.
3. Модуль оптимізації
 - Використання алгоритмів оптимізації для пошуку найкращих рішень (лінійна, комбінаторна та багатокритеріальна оптимізація).
 - Планування маршрутів перевезень, графіків постачання та розподілу ресурсів.
4. Модуль прийняття рішень

- Використання нейронних мереж або генетичних алгоритмів для генерації оптимальних рішень у реальному часі.
 - Можливість автоматичного або напівавтоматичного прийняття рішень з урахуванням ризиків.
5. Інтерфейс користувача (UI)
- Інтерактивний дашборд для відображення ключових показників ефективності та прогнозів.
 - Можливість ручного втручання оператора для корекції запропонованих рішень.
6. Інтеграційний модуль
- Забезпечення з'єднання з іншими системами через REST API або GraphQL.
 - Інтеграція із хмарними платформами (AWS, Azure) для забезпечення масштабованості.

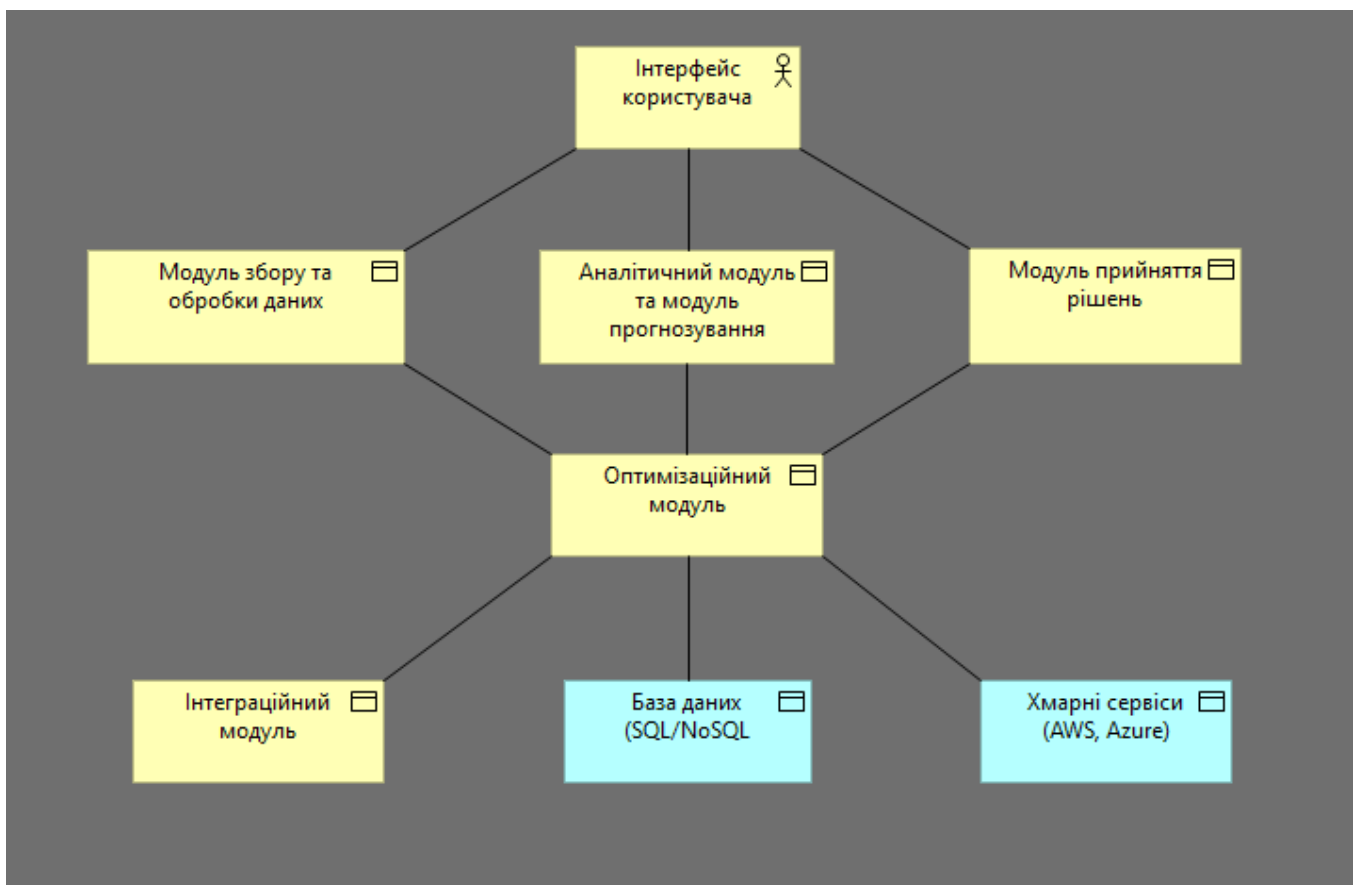


Рисунок 2.2. Схема архітектури моделі

2.3 Аналіз алгоритмів оптимізації для управління розподілом виробничих ресурсів

Для системи ШІ з автоматизації процесу прийняття рішень у логістиці управління розподілом виробничих ресурсів є однією з основних задач. Для цього використовують низку оптимізаційних алгоритмів, які дозволяють ефективно розподіляти ресурси, такі як транспорт, складські потужності, робоча сила та час. Зупинимося на ключових алгоритмах, які можуть бути використані для оптимізації управління розподілом виробничих ресурсів у логістиці:

1. Генетичні алгоритми

- Принцип роботи: Генетичні алгоритми (GA) базуються на природному відборі та еволюції. Вони використовують популяцію можливих рішень (індивідів), які еволюціонують через процеси схрещування, мутацій та відбору, що дозволяє знайти оптимальне або наближене до оптимального рішення для складних задач.
- Застосування: GA можуть використовуватися для вирішення задач оптимального розподілу транспортних засобів, робочої сили або планування виробництва, особливо коли необхідно врахувати множину обмежень і критеріїв (витрати, час, продуктивність).

2. Алгоритм мурашиної колонії (*Ant Colony Optimization, ACO*)

- Принцип роботи: ACO імітує поведінку мурах у пошуку найкоротших шляхів між джерелами їжі та мурашником. Мурахи залишають феромони на шляху, якими прямують інші мурахи, що дозволяє знайти оптимальний шлях з часом.
- Застосування: Цей алгоритм підходить для задач оптимізації маршрутів та управління транспортом, де необхідно знайти найкращі шляхи доставки з урахуванням реальних умов (дорожні затори, відстані, час доставки).

3. Алгоритм табу-пошуку (*Tabu Search*)

- Принцип роботи: Алгоритм використовує ітеративний пошук, де на кожному кроці обчислюється рішення, близьке до поточного, з певними обмеженнями (запобігання поверненню до попередніх рішень, що називається «табування»).

- Застосування: Табу-пошук є ефективним для планування ресурсів та розподілу робочої сили, де потрібно швидко знайти рішення в складних умовах з великою кількістю змінних.

4. Метод рою частинок (Particle Swarm Optimization, PSO)

- Принцип роботи: PSO моделює поведінку рою птахів або риби, які пересуваються як група до певної цілі. Кожна частинка (можливе рішення) рухається в напрямку кращих рішень, вивчаючи досвід інших частинок.

- Застосування: Цей алгоритм підходить для оптимізації процесів управління ресурсами на рівні підприємства або складу, наприклад, для оптимального розподілу товарів у сховищах або розподілу навантаження на робочі потужності.

5. Лінійне програмування (LP)

- Принцип роботи: Лінійне програмування використовується для знаходження оптимальних рішень задач, де цільова функція та обмеження є лінійними. Мета — мінімізувати або максимізувати певну ціль (наприклад, витрати, час) за наявності набору обмежень.

- Застосування: LP часто використовується для задач планування виробничих ресурсів, таких як розподіл робочої сили, управління запасами або транспортування, особливо коли є явні лінійні залежності між змінними.

6. Динамічне програмування

- Принцип роботи: Динамічне програмування розбиває складну задачу на підзадачі, які можуть бути вирішені окремо, і потім об'єднує рішення для досягнення загальної мети. Воно особливо ефективне при вирішенні багатокрокових задач оптимізації.

- Застосування: Використовується для управління запасами, коли необхідно приймати рішення поетапно, наприклад, для визначення оптимальних моментів поповнення запасів або планування виробничих потужностей з урахуванням зміни попиту.

7. Метод гілок і меж (Branch and Bound)

- Принцип роботи: Цей метод будує дерево рішень, де кожна гілка представляє можливий варіант рішення. Він досліджує гілки для пошуку оптимального рішення, відсікаючи ті, які не можуть привести до кращого результату.

- Застосування: Метод гілок і меж ефективний для вирішення задач розподілу ресурсів у логістиці, особливо коли потрібно працювати з дискретними змінними, наприклад, плануванням транспорту або виробничих завдань.

8. Алгоритм імітаційного відпалу (Simulated Annealing)

- Принцип роботи: Алгоритм імітує процес охолодження металів, коли під час відпалу кристалічна решітка прагне до найбільш енергетично вигідного стану. Алгоритм поступово зменшує «температуру» пошуку оптимальних рішень, дозволяючи уникнути локальних мінімумів.

- Застосування: Імітаційний відпал корисний для складних логістичних задач, таких як планування маршрутів, коли є багато локальних мінімумів, і необхідно уникнути застрягання в неперспективних рішеннях.

9. Комбінаторна оптимізація

- Принцип роботи: Цей підхід вирішує задачі, де потрібно вибрати найкраще рішення з кінцевого числа варіантів. Метою є знаходження комбінації рішень, яка максимально задовольняє всі вимоги.

- Застосування: Використовується для задач, де потрібно обрати найкращу комбінацію транспортних засобів або оптимальне планування роботи на підприємстві.

2.4 Порівняння та вибір найбільш ефективних алгоритмів оптимізації

Для того щоб визначити найбільш ефективні алгоритми оптимізації, потрібно враховувати кілька ключових аспектів: складність задачі, розмірність проблеми, наявність обмежень та швидкість отримання результату. Давайте порівняємо найпопулярніші алгоритми за цими критеріями в таблиці 2.1.

Таблиця 2.1

Алгоритм	Точність	Швидкість	Складність реалізації	Здатність до масштабування	Підходить	Недоліки
Генетичні алгоритми (GA)	Висока (підбір параметрів)	Середня	Середня	Висока	Оптимізація складних багатофакторних задач	Потребує великої кількості ітерацій, можливо застрягання в локальних мінімумах
Алгоритм мурашиної колонії (ACO)	Середня	Середня	Середня	Висока	Оптимізація маршрутів, пошук найкоротших шляхів	Повільний на великих даних, залежність від кількості агента
Алгоритм табу-пошуку	Висока	Швидка	Середня	Висока	Планування, розподіл ресурсів	Потребує тонкої настройки для уникнення великих витрат часу
Метод рою частинок (PSO)	Середня	Швидка	Середня	Висока	Нелінійні та багатокритеріальні задачі	Може застрягти у локальних мінімумах
Лінійне програмування (LP)	Висока	Дуже швидка	Низька	Низька	Задачі з лінійними обмеженнями	Непридатний для нелінійних або складних систем з великими даними
Динамічне програмування	Висока	Повільна	Висока	Середня	Задачі з ієрархічною структурою, багатокрокові задачі	Вимоги до великих обсягів пам'яті та обчислень

Метод гілок і меж	Висока	Повільна	Висока	Низька	Рішення задач розподілу ресурсів	Повільний процес вирішення через розгалуження дерева рішень
Алгоритм імітаційного відпалу (SA)	Висока	Середня	Середня	Висока	Оптимізація маршрутів, розподіл навантаження	Може бути повільним при великій кількості обмежень

Вибір найкращого підходу:

1. Для транспортної логістики (оптимізація маршрутів)

- Алгоритми: Алгоритм мурашиної колонії (ACO) або імітаційний відпал (SA).
- Обґрунтування: ACO підходить для пошуку найкоротших маршрутів, враховуючи реальні умови (затори, зміна дорожніх умов), тоді як SA може знижувати ризики застрягання у локальних мінімумах та забезпечувати більш глобальне вирішення задач.

2. Для планування запасів та управління ресурсами

- Алгоритми: Генетичні алгоритми (GA) або табу-пошук.
- Обґрунтування: Генетичні алгоритми можуть ефективно працювати з великою кількістю змінних і обмежень, тоді як табу-пошук забезпечує стабільні рішення при задачах з великою кількістю варіантів планування.

3. Для швидкого прийняття рішень у реальному часі

- Алгоритми: Метод рою частинок (PSO) або лінійне програмування (LP).
- Обґрунтування: PSO забезпечує швидке знаходження рішень для багатокритеріальних задач у реальному часі, а LP дозволяє швидко вирішувати прості задачі з чіткими обмеженнями.

Для складних і багатофакторних задач оптимізації ресурсів у логістиці найбільш ефективними будуть **генетичні алгоритми та алгоритм табу-пошуку**. Для транспортної логістики оптимізація маршрутів за допомогою **ACO** та

імітаційного відпалу також буде доречною, враховуючи реальні умови дорожнього руху.

2.5 Адаптація та інтеграція алгоритмів до моделі системи

Адаптація та інтеграція оптимізаційних алгоритмів у систему штучного інтелекту для логістики вимагає ретельного планування, врахування специфіки задачі, а також розуміння, як конкретні алгоритми можуть бути інтегровані в єдину архітектуру системи. Розглянемо загальні принципи адаптації алгоритмів, моделі їх взаємодії та кроки для інтеграції у логістичну систему.

Кроки адаптації алгоритмів до логістичної системи

1. Аналіз потреб системи та задач

Перед початком адаптації важливо чітко визначити ключові задачі логістичної системи:

- Оптимізація маршрутів: Пошук найкоротших шляхів для доставки товарів.
- Розподіл ресурсів: Управління складськими запасами, робочою силою та транспортом.
- Планування попиту: Прогнозування кількості товарів, які потрібні на складі.
- Прийняття рішень у реальному часі: Адаптація до змін умов (дорожні затори, затримки у виробництві тощо).

На цьому етапі вирішується, які конкретні алгоритми будуть використані для кожної з задач:

- Алгоритм мурашиної колонії (ACO) для оптимізації маршрутів.
- Генетичні алгоритми (GA) для розподілу ресурсів.
- Метод рою частинок (PSO) або імітаційний відпал (SA) для швидкого прийняття рішень у реальному часі.

2. Адаптація алгоритмів під специфіку даних

Алгоритми повинні бути адаптовані до типів даних, що використовуються у системі. Це може включати:

- Формат даних: Транспортні мережі, інформація про складські запаси, замовлення та потоки транспорту повинні бути подані у вигляді графів, масивів або таблиць.
- Структура задач: Якщо задача нелінійна, алгоритми повинні враховувати нелінійні обмеження та залежності. Наприклад, GA можна адаптувати для роботи з нелінійними обмеженнями щодо транспорту.
- Вхідні параметри: Оптимізаційні алгоритми можуть вимагати тонкого налаштування параметрів, таких як кількість ітерацій для GA або кількість мурашок у АСО.

3. Розподіл задач між алгоритмами

Логістична система може використовувати комбінований підхід, де різні алгоритми вирішують різні підзадачі. Наприклад:

- АСО для оптимізації маршрутів.
- Генетичний алгоритм для розподілу транспортних ресурсів на різні маршрути.
- PSO для прийняття рішень у реальному часі щодо змін у планах.

Можлива інтеграція декількох алгоритмів у багаторівневу систему, де АСО використовується на першому рівні для планування маршрутів, а GA — на другому рівні для управління ресурсами на цих маршрутах.

4. Вибір платформи та інфраструктури

Для інтеграції алгоритмів у реальну систему логістики необхідно врахувати:

- Вибір платформи: Використання мов програмування (Python, C++), які мають підтримку бібліотек для оптимізації, таких як TensorFlow, PyTorch, SciPy, DEAP (для генетичних алгоритмів).
- Інфраструктура для обчислень: Використання кластерів або хмарних сервісів для обробки великих обсягів даних та паралельного запуску алгоритмів. Особливо важливо для складних ітераційних алгоритмів, таких як GA та АСО.

5. Інтерфейс для взаємодії з алгоритмами

Користувачі логістичної системи повинні мати доступ до зрозумілих інтерфейсів для налаштування параметрів алгоритмів, аналізу результатів оптимізації та прийняття остаточних рішень. Це може включати:

- Графічні інтерфейси користувача (GUI): Дашборди для відображення результатів, маршрутів, даних про ресурси та витрати.
- API для інтеграції з іншими системами: API для підключення до ERP-систем, баз даних, транспортних систем тощо.

Інтеграція алгоритмів у модель ШІ

1. Взаємодія з компонентами ШІ-системи

Алгоритми повинні бути частиною загальної архітектури ШІ, яка включає:

- Модуль збору та підготовки даних: Важливий компонент, що відповідає за збір, очищення та структурування даних, необхідних для роботи алгоритмів оптимізації.
- Модуль навчання: Використання алгоритмів машинного навчання, що допоможуть системі покращувати рішення на основі історичних даних.
- Модуль прийняття рішень: Адаптація обчислених алгоритмами рішень до реальних умов.
- Модуль моніторингу: Автоматизований моніторинг ефективності рішень, з можливістю адаптації на основі поточної інформації.

2. Оптимізація роботи системи у реальному часі

Алгоритми повинні бути здатні приймати рішення у реальному часі:

- Інкрементальне оновлення: Оновлення маршрутів або планів без повної перерахунку, що може бути корисним для алгоритмів, таких як PSO або динамічне програмування.
- Обробка поточних даних: Використання систем для обробки поточних даних у реальному часі (наприклад, Apache Kafka або Spark Streaming), що дозволить адаптувати алгоритми під поточні зміни.

3. Моделі взаємодії алгоритмів

Для ефективної роботи в одній системі різні алгоритми можуть взаємодіяти через:

- Каскадну архітектуру: Де результат одного алгоритму є вхідними даними для іншого. Наприклад, АСО генерує маршрути, які потім оптимізуються GA для розподілу транспортних засобів.
- Паралельну архітектуру: Коли алгоритми працюють одночасно для вирішення різних підзадач (оптимізація маршрутів та планування ресурсів), а результати потім комбінуються.

Приклад інтеграції в логістичну систему

1. Збір та підготовка даних: Дані про замовлення, транспортні засоби, стан доріг та складів збираються з різних джерел.
2. Алгоритм АСО: Використовується для планування маршрутів доставки з урахуванням обмежень (відстань, час доставки, умови дороги).
3. Генетичний алгоритм: Після визначення маршрутів, GA оптимізує розподіл транспортних ресурсів по різних маршрутах, враховуючи їх доступність та витрати.
4. Алгоритм імітаційного відпалу (SA): Використовується для адаптації рішень у реальному часі, якщо змінились умови (наприклад, виникли затримки на дорогах або змінився попит).
5. Моніторинг: Рішення контролюються та коригуються у процесі їх виконання на основі поточних даних.

Розділ 3. РЕАЛІЗАЦІЯ ПРОТОТИПУ СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ

3.1 Розробка прототипу системи на основі вибраних алгоритмів

Для розробки прототипу системи на основі вибраних алгоритмів необхідно дотримуватися чіткої послідовності етапів.

1. Формулювання задачі (цілі) та визначення вимог до системи.

Основна задача для створюваної системи автоматизації прийняття рішень у логістиці полягає в розробці інтелектуальної платформи, здатної оптимізувати та автоматизувати процеси розподілу ресурсів і транспортних маршрутів для підвищення ефективності ланцюга постачань.

Система повинна забезпечити можливість:

- Швидкого та точного прогнозування потреб у ресурсах на основі історичних даних і поточних умов.
- Розрахунку оптимальних транспортних маршрутів з урахуванням параметрів, таких як трафік, погодні умови, день тижня, та інші логістичні фактори.
- Автоматизації процесу прийняття рішень для зниження операційних витрат і підвищення швидкості реакції на зміни в попиті.
- Забезпечення моніторингу в реальному часі з можливістю адаптації планів залежно від поточних умов і непередбачуваних факторів.

2. Вибір алгоритму - дослідження підходів, оцінка відповідності.

На етапі вибору алгоритму, варто ретельно оцінити різні підходи, які можуть вирішувати завдання оптимізації та автоматизації у логістиці. Давайте розглянемо кілька основних алгоритмів та спробуємо розібратися, чому саме алгоритм мурашиної колонії є найбільш відповідним для системи автоматизації прийняття рішень у логістиці.

1. Огляд різних підходів до оптимізації

Для оптимізації маршрутів і розподілу ресурсів у логістиці можна використовувати різні методи, зокрема:

- **Генетичні алгоритми:** Імітують еволюційний процес, створюючи "популяції" рішень, які через процеси кросоверу і мутації еволюціонують до кращих рішень. Генетичні алгоритми добре працюють у складних задачах оптимізації, але можуть бути повільними при дуже великій кількості маршрутів та ресурсів. Крім того, для логістики, де зміна вхідних умов часто вимагає негайної реакції, генетичний алгоритм може не забезпечити достатньої швидкості реакції.
- **Алгоритм рою частинок:** Заснований на імітації поведінки зграї птахів або рою комах, цей алгоритм використовує набір рішень ("частинок"), які пересуваються в пошуках оптимального рішення. Хоча він добре підходить для пошуку глобального оптимуму у задачах з великою кількістю параметрів, йому часто бракує точності при обробці географічних задач з множинними обмеженнями, характерними для логістики.
- **Метод гілок та меж:** Цей підхід зводиться до ітеративного розбиття задачі на підзадачі з обмеженням, поки не буде знайдено оптимальний маршрут. Метод забезпечує точне рішення, але вимагає значних обчислювальних ресурсів при великій кількості пунктів доставки.
- **Лінійне програмування та комбінаторна оптимізація:** Ефективні при розв'язанні завдань, що піддаються лінійній формалізації. Проте, такі алгоритми часто не враховують фактори реального світу, як-от трафік або погодні умови, що ускладнює їх застосування у логістичних системах, де потрібна адаптація до динамічних змін.
- **Алгоритм мурашиної колонії:** Імітує поведінку мурах у пошуку найкоротших шляхів між джерелами їжі та мурашником. Мурахи залишають феромони на шляху, якими прямують інші мурахи, що дозволяє знайти оптимальний шлях з часом.

2. Переваги алгоритму мурашиної колонії

Алгоритм мурашиної колонії, натхненний поведінкою мурах у природі, коли вони шукають найкоротші шляхи до їжі, підходить для задач логістики завдяки декільком ключовим особливостям:

- **Адаптація до динамічних змін:** Мурашиний алгоритм імітує процес пошуку оптимального маршруту на основі феромонів, які підсумовують інформацію про пройдені маршрути. Завдяки цьому алгоритм адаптується до змінних умов, як-от затори або раптові потреби у зміні маршруту, що є критичним у логістиці.
- **Паралельне вивчення рішень:** Кілька "мурах" одночасно досліджують маршрути, що дозволяє швидше знаходити можливі шляхи і оптимальні маршрути. Це значно прискорює обчислення у порівнянні з методами, які досліджують варіанти послідовно.
- **Урахування множинних факторів:** Мурашиний алгоритм добре підходить для врахування множинних факторів, таких як відстань, трафік і погодні умови. Це відбувається завдяки змінним параметрам алгоритму — альфа (α) та бета (β), які визначають вплив феромонів і відстаней відповідно.
- **Гнучкість та масштабованість:** Алгоритм мурашиної колонії може працювати як на малій кількості точок (маршрутів), так і на значних мережах доставки. Це робить його ідеальним для логістичних систем, що можуть масштабуватися та пристосовуватися до великих обсягів даних.

3. Відповідність алгоритму задачі системи

Алгоритм мурашиної колонії є оптимальним вибором для системи, оскільки він відповідає основним вимогам до системи логістики:

- **Оптимізація маршрутів:** Алгоритм ефективно знаходить коротші маршрути, що відповідає задачі системи — автоматизації оптимізації маршрутів доставки.
- **Адаптивність до реальних умов:** У логістиці часто відбуваються зміни, наприклад, зростання попиту на деяких пунктах доставки, зміни погодних умов або раптові затори. Завдяки адаптивності до феромонних слідів алгоритм дозволяє оптимально підлаштовуватися під такі зміни.
- **Реалістичність моделі обробки даних:** Алгоритм мурашиної колонії дозволяє проводити багаторазові ітерації з використанням даних про

попередні маршрути, що покращує якість прогнозування оптимальних шляхів.

Таким чином, вибір алгоритму мурашиної колонії є виправданим, оскільки він найкраще відповідає вимогам до створюваної системи, надаючи оптимальні, швидкі й адаптивні рішення для автоматизації логістичних процесів.

4. Архітектурний дизайн прототипу - модулі системи, комунікація між модулями.

У створюваній системі автоматизації прийняття рішень для логістики модулі мають тісно взаємодіють один з одним, щоб забезпечити ефективну роботу алгоритмів і своєчасне надання оптимізованих рішень. Ось детальний опис модулів, їх функцій та взаємодії між ними:

1. Модуль збору та обробки даних

- **Вхідні дані:** В цей модуль завантажуються різного роду дані такі як, погода, день тижня, запаси на складах, трафік і так далі.
- **Обробка та передача даних:** Після збору даних цей модуль очищає і перетворює їх у придатний для аналізу формат. Після обробки дані передаються до аналітичного та прогнозуючого модуля для подальшої роботи.

2. Аналітичний та прогнозуючий модуль

- **Отримання даних:** Отримує оброблені дані від модуля збору та обробки.
- **Прогнозування попиту та аналіз тенденцій:** Використовуючи модель машинного навчання, цей модуль прогнозує майбутній маршрут логістики зважуючи усі фактори, що дозволяє завчасно планувати оптимальні маршрути, запаси і витрати ресурсів.
- **Передача результатів:** Прогнозні результати надсилаються до оптимізаційного модуля, який використовує їх для прийняття конкретних рішень.

3. Оптимізаційний модуль

- **Отримання прогнозів та додаткових даних:** Отримує дані з прогнозуючого модуля, включаючи заплановані обсяги попиту, дані про наявні ресурси та

маршрути. Також може взаємодіяти з даними про поточну погоду, а також із запаси та умовами на дорогах.

- **Оптимізація маршрутів:** На основі алгоритму мурашиної колонії цей модуль знаходить найефективніший маршрут з урахуванням всіх вхідних факторів. Він обирає оптимальні маршрути, які забезпечують мінімальні витрати часу, палива або інших ресурсів.
- **Відправка оптимальних рішень:** Після обчислення найкращого маршруту результати надсилаються до модуля прийняття рішень.

4. Модуль прийняття рішень

- **Аналіз оптимізації:** Використовуючи оптимізовані дані з оптимізаційного модуля, цей модуль формує кінцевий набір рішень для кожного конкретного сценарію (маршрут, час дороги).
- **Автоматизація процесів:** Модуль може автоматично виконувати рішення на основі отриманих даних або ж передавати їх оператору для затвердження.
- **Зворотний зв'язок:** Надсилає інформацію про виконання до системи для обробки, щоб результати були враховані для подальшого аналізу та прогнозування.

5. Загальна схема взаємодії

Усі модулі взаємодіють циклічно та постійно оновлюють один одного даними:

- ❖ Дані збираються та обробляються, передаються в аналітичний модуль.
- ❖ Аналітичний модуль прогнозує потреби та передає інформацію до оптимізаційного модуля.
- ❖ Оптимізаційний модуль розраховує найкращий шлях і передає інформацію до модуля прийняття рішень.
- ❖ Прийняте рішення надсилається на виконання та зворотний зв'язок передається в систему для подальшого аналізу і корекції.

6. Підготовка даних - збір даних, обробка даних.

Збір даних

Підготовка даних є ключовим етапом в будь-якій системі, яка включає алгоритми машинного навчання та оптимізації. Цей етап включає кілька важливих кроків, які мають забезпечити належну якість і структуру даних для подальшої обробки та аналізу.

На цьому етапі збираються всі необхідні дані, які будуть використовуватися для навчання моделі машинного навчання, оптимізації маршрутів, а також для прогнозування часу доставки.

Основні джерела даних:

1. Логістичні дані:

- **Місцезнаходження точок доставки:** координати складів, магазинів, клієнтів тощо.
- **Дані про відстані між точками:** Це можуть бути як географічні дані, так і відстані, що вимірюються на основі карт або GPS. Також це можуть просто бути дані, які мають сенс всередині системи(тестові дані).

2. Дані про трафік:

- **Рівень трафіку** в певний час доби, день тижня. Це важливі фактори для точного прогнозування часу на шляху.

3. Погодні умови:

- **Погодні дані** (наприклад, температура, опади, вітер тощо), що можуть впливати на час доставки або на вибір маршруту. Або можуть просто бути сприятливі та несприятливі погодні дані.

4. Дані про зупинки:

- Інформація про кількість зупинок, які будуть зроблені на шляху (наприклад, зупинки для завантаження чи розвантаження товарів).

Обробка даних

Після того як дані зібрані, вони повинні бути попередньо оброблені, щоб бути у зручному для аналізу та моделювання вигляді.

Кроки обробки

1)Перевірка на відсутні значення:

- Для даних, які можуть бути неповними, потрібно провести перевірку на **відсутність значень** (наприклад, якщо для деяких точок не вистачає погодних умов або рівня трафіку).
- Відсутні дані можуть бути заповнені середнім значенням або видалені, якщо їх занадто багато.

2) Масштабування та нормалізація даних:

- В залежності від виду даних, їх необхідно **масштабувати** або **нормалізувати**, щоб привести всі значення до одного масштабу. Наприклад:
 - Рівень трафіку (від 1 до 4) можна масштабувати для зручності.
 - Відстані між точками, що вимірюються в кілометрах, можуть бути масштабовані для кращої роботи алгоритмів оптимізації.

3) Формування ознак для моделі машинного навчання:

- Зібрані дані потрібно перетворити в **структуровані ознаки**, які будуть подаватися до моделі машинного навчання. Наприклад:

Відстань, день тижня, рівень трафіку, погодні умови, кількість зупинок — це все перетворюється в **матрицю ознак** для моделі.

Для алгоритму мурашиної колонії дані, такі як координати точок і відстані між ними, використовуються для побудови графу.

4) Очищення даних:

- Важливо очистити дані від **неправильних або неадекватних значень**. Це може включати видалення аномальних значень або фільтрацію даних, які не відповідають необхідним умовам.

Структуровані дані для алгоритмів

★ Алгоритм мурашиної колонії:

- Для цього алгоритму необхідні дані про відстані між точками, які будуть використовуватися для побудови графу. Виходячи з координат точок, система обчислює евклідову відстань між точками, що стає вагами на ребрах графу.

★ Модель машинного навчання:

- Для лінійної регресії використовуються дані про трафік, погоду, відстань, день тижня та кількість зупинок, щоб побудувати навчальну вибірку та здійснити прогнозування.

Цей етап є критичним для забезпечення ефективності та надійності всієї системи, адже без якісних даних модель не зможе працювати коректно.

5. Реалізація - програмування алгоритмів, налаштування параметрів.

Прототип системи ШІ буде реалізований у середовищі PyCharm на мові програмування Python.

Для початку завантажимо усі необхідні бібліотеки для створення системи ШІ, яка буде автоматично обирати найкращий маршрут між складом і магазинами куди потрібно буде доставити товар, враховуючи різні побічні фактори (день тижня, погода, затори, тощо), а також розраховувати кілометраж та час за який буде подолано цю відстань.

Завантажимо усі необхідні бібліотеки для роботи у середовищі PyCharm (рис. 3.1).

```

1 import numpy as np
2 import networkx as nx
3 import random
4 from sklearn.linear_model import LinearRegression
5 from sklearn.model_selection import train_test_split
6 from sklearn.metrics import mean_squared_error
7

```

Рисунок 3.1. Бібліотеки

import numpy as np:

- **numpy** — це поширена бібліотека для роботи з масивами та матрицями в Python, а також для виконання наукових обчислень. Вона дозволяє ефективно працювати з великими наборами даних і надає численні функції для математичних операцій, таких як обчислення відстаней, виконання лінійної алгебри, генерація випадкових чисел тощо.

- `np` — це загальноприйнята аббревіатура, яку використовують для зручності при роботі з цією бібліотекою.

import networkx as nx:

- `networkx` — це бібліотека для створення, маніпулювання та аналізу графів та мереж. Вона дозволяє працювати з різними типами графів (орієнтованими, неорієнтованими) і виконувати різноманітні операції над ними, такі як пошук найкоротших шляхів, оцінка центральності, візуалізація та інші.
- `nx` — це псевдонім, який зазвичай використовують для зручності, щоб не писати щоразу `networkx`.

import random:

- `random` — стандартна бібліотека Python для роботи з випадковими числами. Вона дозволяє генерувати випадкові числа, вибирати випадкові елементи з послідовностей, перемішувати елементи в списках і т. д. У нашому коді ми використовуємо її для вибору випадкових маршрутів при реалізації алгоритму мурашиної колонії.

from sklearn.linear_model import LinearRegression:

- `sklearn` (або `scikit-learn`) — це популярна бібліотека для машинного навчання, яка надає інструменти для створення та використання моделей машинного навчання, обробки даних і оцінки моделей.
- `LinearRegression` — це клас у `sklearn`, який реалізує модель лінійної регресії. Лінійна регресія використовується для прогнозування числових значень (наприклад, часу доставки на основі різних ознак).

from sklearn.model_selection import train_test_split:

- `train_test_split` — це функція з бібліотеки `sklearn`, яка дозволяє розділити набір даних на дві частини: одну для тренування моделі (`train`) та іншу для її тестування (`test`). Це дозволяє перевірити, наскільки добре модель працює на нових даних, яких вона не бачила під час тренування.

from sklearn.metrics import mean_squared_error:

`mean_squared_error` — це функція з бібліотеки `sklearn`, яка обчислює середню квадратичну помилку (MSE). Це одна з метрик для оцінки якості регресійних

моделей, яка показує, наскільки точними є прогнози моделі. Чим менше значення MSE, тим точніше модель.

Цей алгоритм використовується для пошуку оптимального маршруту доставки (наприклад, від складу до різних магазинів). Ідея алгоритму полягає в тому, що мурахи, шукаючи їжу, залишають феромони на пройдених шляхах. Інші мурахи, слідуючи за феромонами, вибирають більш короткі маршрути, що допомагає алгоритму знаходити найкращий шлях.

```

evaporation_rate = 0.5)

# Створюємо граф для моделювання точок доставки
G = nx.complete_graph(len(locations))
for (u, v) in G.edges():
    # Відстань між точками
    G.edges[u, v]['distance'] = np.linalg.norm(
        np.array(coordinates[u]) - np.array(coordinates[v])) # Евклидова відстань

# Ініціалізація феромонів
pheromones = {edge: 1 for edge in G.edges}

best_route = None
best_distance = float('inf')

for _ in range(iterations):
    all_routes = []
    for vehicle in range(num_vehicles):
        route = [0] # Початкова точка
        unvisited = set(range(1, len(locations)))

        while unvisited:
            current = route[-1]
            probabilities = []
            for neighbor in unvisited:
                pheromone = pheromones[(min(current, neighbor), max(current, neighbor))]
                distance = G.edges[current, neighbor]['distance']
                probability = (pheromone ** alpha) * ((1 / distance) ** beta)
                probabilities.append((neighbor, probability))

```

Рисунок 3.2. Алгоритм мурашиної колонії (Ant Colony Optimization, ACO)

Робота алгоритму в кодї:

- **Граф:** Створюється повний граф (за допомогою networkx), де кожна точка доставки є вершиною, а з'єднання між ними — ребра.

- **Евклідова відстань:** Для кожної пари точок визначається відстань за допомогою Евклідової метрики.
- **Феромони:** Спочатку всі ребра мають однаковий рівень феромонів. Під час кожної ітерації алгоритм оновлює ці рівні в залежності від пройденого шляху.
- **Ймовірність вибору шляху:** Мураха вибирає наступну точку на основі ймовірності, яка враховує як феромони, так і відстань між точками. Чим більше феромонів на шляху і менша відстань, тим більша ймовірність вибору цього шляху.
- **Оновлення феромонів:** Після кожного циклу феромони на шляху, який привів до найкращого маршруту, збільшуються, а на інших шляхах зменшуються.

```

40     # Нормалізація ймовірностей
41     total = sum(p for _, p in probabilities)
42     probabilities = [(node, p / total) for node, p in probabilities]
43
44     # Вибір наступної точки за ймовірностями
45     next_node = random.choices([node for node, _ in probabilities],
46                               weights=[p for _, p in probabilities])[0]
47
48     route.append(next_node)
49     unvisited.remove(next_node)
50
51     # Оновлення феромонів на пройденому маршруті
52     pheromones[(min(current, next_node), max(current, next_node))] *= (1 - evaporation_rate)
53
54     # Закнути маршрут, повернувшись до початкової точки
55     route.append(route[0])
56     all_routes.append(route)
57
58     # Оцінка маршрутів та оновлення феромонів
59     for route in all_routes:
60         route_distance = sum(G.edges[route[i], route[i + 1]]['distance'] for i in range(len(route) - 1))
61         if route_distance < best_distance:
62             best_distance = route_distance
63             best_route = route
64
65     # Посилення феромонів на найкращих маршрутах
66     for i in range(len(route) - 1):
67         edge = (min(route[i], route[i + 1]), max(route[i], route[i + 1]))
68         pheromones[edge] += 1 / route_distance
69
70     return best_route, best_distance

```

Рисунок 3.3. Мурашиний алгоритм

Друга частина коду працює за принципом алгоритму мурашиної колонії де і реалізовано вибір наступної точки, потім йде оцінка маршрутів та оновлення феромонів. І останньою частиною є посилення феромонів на найкращий маршрут.

Наступним кроком буде створення моделі навчання ШІ для системи.

```

73 # 2. Модель машинного навчання (лінійна регресія) для передбачення часу
74 def train_ml_model():
75     # Генерація даних з більш реалістичним розподілом часу в дорозі
76     distances = np.random.uniform(low=5, high=50, size=100) # Випадкові відстані (км) між 5 і 50
77     traffic_levels = np.random.randint(low=1, high=5, size=100) # Рівень трафіку від 1 до 4
78     weather_conditions = np.random.randint(low=1, high=3, size=100) # Погодні умови (1 - сприятливі, 2 - несприятливі)
79     days_of_week = np.random.randint(low=0, high=6, size=100) # Дні тижня від 0 до 6
80     num_stops = np.random.randint(low=3, high=10, size=100) # Кількість зупинок
81
82     # Цільова змінна – реалістичний час у дорозі з урахуванням різних факторів
83     y = distances * 2 + traffic_levels * 10 + weather_conditions * 15 + np.random.normal(loc=0, scale=10, size=100)
84
85     # Об'єднання всіх ознак у матрицю
86     X = np.column_stack([distances, days_of_week, traffic_levels, weather_conditions, num_stops])
87
88     # Поділ даних на тренувальну та тестову вибірку
89     X_train, X_test, y_train, y_test = train_test_split(*arrays: X, y, test_size=0.2, random_state=42)
90
91     # Створення та навчання моделі лінійної регресії
92     model = LinearRegression()
93     model.fit(X_train, y_train)
94
95     # Оцінка точності моделі
96     y_pred = model.predict(X_test)
97     mse = mean_squared_error(y_test, y_pred)
98     print(f'Mean Squared Error: {mse}')
99
100     return model
101

```

Рисунок 3.4 Модель навчання ШІ

Тепер можна використати нашу створену модель навчання ШІ для передбачення часу маршруту від складу до магазинів і назад.

```

103 # 3. Використання моделі машинного навчання для передбачення часу маршруту
104 def predict_route_time(model, route, coordinates, day_of_week, traffic_level, weather):
105     # Перетворимо маршрут на ознаки
106     distances = [np.linalg.norm(np.array(coordinates[route[i]]) - np.array(coordinates[route[i + 1]])) for i in
107                 range(len(route) - 1)]
108     total_distance = sum(distances)
109
110     # Ознаки: відстань, день тижня, трафік, погода, кількість зупинок
111     features = np.array([total_distance, day_of_week, traffic_level, weather, len(route)])
112     features = features.reshape(1, -1)
113
114     # Передбачення часу маршруту з використанням моделі
115     predicted_time = model.predict(features)
116     predicted_time = max(predicted_time[0], 0) # Забезпечення невід'ємного значення
117     return predicted_time

```

Рисунок 3.5. Використання моделі машинного навчання

Наступна частина коду буде включати дані про знаходження нашого складу і координати магазинів.

```

120 # Основний блок
121 if __name__ == "__main__":
122     # Приклад даних
123     locations = ['Склад', 'Магазин №1', 'Магазин №2', 'Магазин №3', 'Магазин №4']
124     coordinates = {
125         0: [0, 0], # Склад
126         1: [10, 10], # Магазин №1
127         2: [20, 20], # Магазин №2
128         3: [30, 30], # Магазин №3
129         4: [40, 40] # Магазин №4
130     }

```

Рисунок 3.6. Дані про магазини

locations = ['Склад', 'Магазин №1', 'Магазин №2', 'Магазин №3', 'Магазин №4']

- Змінна `locations` містить список рядків, в яких перераховані назви різних локацій, наприклад, склад і магазини. Ці дані будуть використовуватись для позначення точок на графі, які потрібно відвідати (в даному випадку це склад і магазини).

coordinates = { ... }

- `coordinates` — це словник, де ключами є індекси (цифри від 0 до 4), а значеннями — списки з двома елементами, що представляють координати відповідних локацій в двовимірному просторі (наприклад, x і y координати).
- Наприклад:
 - Склад (0): координати $[0, 0]$.
 - Магазин №1 (1): координати $[10, 10]$.
 - Магазин №2 (2): координати $[20, 20]$.
 - Магазин №3 (3): координати $[30, 30]$.
 - Магазин №4 (4): координати $[40, 40]$.
 - Цей словник буде використовуватись для розрахунку відстаней між локаціями, що є важливим для обчислення оптимальних маршрутів та часу в алгоритмі оптимізації маршруту.

```

138 # Прогнозуємо час на основі знайденого маршруту
139 predicted_time = predict_route_time(ml_model, best_route, coordinates, day_of_week=3, traffic_level=2, weather=1)
140 print(f'Predicted time for the best route: {predicted_time} minutes')
141 print("Best route found:", best_route)
142 print("Distance of best route (km):", best_distance)
143

```

Рисунок 3.7. Вивід результату

За допомогою цих рядків виводимо:

- Прогнозований час для найкращого маршруту.
- Найкращий маршрут, знайдений за допомогою алгоритму мурашиної колонії.
- Відстань для цього маршруту.

3.2 Тестування прототипу на реальних або симуляційних даних

Запускаємо наш прототип і аналізуємо отриманий результат.

```

Mean Squared Error: 116.02908753685145
Predicted time for the best route: 250.3136507983628 minutes
Best route found: [0, 1, 4, 3, 2, 0]
Distance of best route (km): 113.13708498984761

```

Рисунок 3.8 Результат тестування

1. Mean Squared Error (MSE)

```
Mean Squared Error: 116.02908753685145
```

Середньоквадратична помилка (MSE) — це метрика, яка вимірює середнє квадратичне відхилення між реальними значеннями та передбаченими. Чим менше значення MSE, тим краща модель. У моєму випадку, $MSE = 116.03$ — це величина помилки прогнозу моделі машинного навчання (лінійної регресії). Це означає, що в середньому передбачений час маршруту має відхилення приблизно на 116.03 хвилин від реальних значень.

Варто зазначити що **середньоквадратична помилка (MSE)** повинна зменшуватися з кожним новим запуском, якщо постійно покращувати модель і дозволяти їй навчатися з новими даними.

2. Predicted time for the best route

```
Predicted time for the best route: 250.3136507983628 minutes
```

Це прогнозований час для оптимального маршруту, знайденого за допомогою алгоритму мурашиної колонії та прогнозування з використанням навченої моделі машинного навчання (лінійної регресії). Модель передбачила, що час виконання цього маршруту складе приблизно 250.31 хвилин. Це залежить від кількох факторів, таких як відстань, трафік, погодні умови, день тижня та кількість зупинок.

3. Best route found

```
Best route found: [0, 1, 4, 3, 2, 0]
```

Це **оптимальний маршрут**, знайдений за допомогою **алгоритму мурашиної колонії** (Ant Colony Optimization, ACO). У цьому випадку маршрут починається з точки 0 (склад), потім йде в точки 1 (Магазин №1), 4 (Магазин №4), 3 (Магазин №3), 2 (Магазин №2) і повертається в точку 0 (склад). Тобто, алгоритм обрав маршрут, який мінімізує загальну пройдену відстань з урахуванням відстаней між точками.

Також алгоритм враховував такі фактори, як погода, день тижня тощо, для визначення найкращого маршруту.

4. Distance of best route

```
Distance of best route (km): 113.13708498984761
```

Це відстань оптимального маршруту, знайденого алгоритмом мурашиної колонії. Загальна довжина маршруту складає 113.14 км. Ця відстань обчислюється за допомогою евклідової відстані між усіма точками маршруту.

Можна запустити прототип ще пару разів, щоб удостоверитися в тому, що він показує різні дані.

```

Mean Squared Error: 92.70093735077498
Predicted time for the best route: 271.660623676305 minutes
Best route found: [0, 1, 3, 4, 2, 0]
Distance of best route (km): 113.13708498984761

```

Рисунок 3.9. Другий запуск програми

Можемо бачити, що **середньоквадратична помилка** стала меншою ніж після першого запуску, також збільшився час маршруту через певні критерії про які згадувалося вище, змінився шлях.

```

Mean Squared Error: 81.70744240720349
Predicted time for the best route: 259.1258876144317 minutes
Best route found: [0, 1, 2, 3, 4, 0]
Distance of best route (km): 113.13708498984761

```

Рисунок 3.10. Третій запуск програми

При третьому запуску прототип системи також вказує інші дані, а також помилка зменшилася ще більше, що означає що модель навчається з кожним новим запуском та видає більш точні дані.

Робота в контексті прототипу системи ШІ:

1. **Алгоритм мурашиної колонії (ACO):** Знаходить оптимальний маршрут для перевезення товарів між різними точками (наприклад, склад та магазини). Цей алгоритм використовує феромони та ймовірнісні переходи для вибору шляху з найменшими витратами (в даному випадку, мінімізація відстані).

2. **Модель машинного навчання (лінійна регресія):** Використовується для прогнозування часу виконання маршруту з урахуванням різних факторів, таких як:

- відстань,
- день тижня,
- рівень трафіку,
- погодні умови,

- кількість зупинок.

3. **Навчання моделі** відбувається на випадкових даних, і вона намагається знайти залежність між цими ознаками та часом на маршруті. Після навчання модель використовується для передбачення часу маршруту для знайденого оптимального шляху.

3.3. Навчання прототипу системи ШІ

Розглянемо модель прототипу навчання ШІ у системі автоматизованого прийняття рішень.

```

88 # 2. Модель машинного навчання (лінійна регресія) для передбачення часу
89 def train_ml_model():
90     # Генерація даних з більш реалістичним розподілом часу в дорозі
91     distances = np.random.uniform(low=5, high=50, size=100) # Випадкові відстані (км) між 5 і 50
92     traffic_levels = np.random.randint(low=1, high=5, size=100) # Рівень трафіку від 1 до 4
93     weather_conditions = np.random.randint(low=1, high=3, size=100) # Погодні умови (1 - сприятливі, 2 - несприятливі)
94     days_of_week = np.random.randint(low=0, high=6, size=100) # Дні тижня від 0 до 6
95     num_stops = np.random.randint(low=3, high=10, size=100) # Кількість зупинок
96
97     # Цільова змінна – реалістичний час у дорозі з урахуванням різних факторів
98     y = distances * 2 + traffic_levels * 10 + weather_conditions * 15 + np.random.normal(loc=0, scale=10, size=100)
99
100     # Об'єднання всіх ознак у матрицю
101     X = np.column_stack([distances, days_of_week, traffic_levels, weather_conditions, num_stops])
102
103     # Поділ даних на тренувальну та тестову вибірку
104     X_train, X_test, y_train, y_test = train_test_split(*arrays=X, y, test_size=0.2, random_state=42)
105
106     # Створення та навчання моделі лінійної регресії
107     model = LinearRegression()
108     model.fit(X_train, y_train)
109
110     # Оцінка точності моделі
111     y_pred = model.predict(X_test)
112     mse = mean_squared_error(y_test, y_pred)
113     print(f'Mean Squared Error: {mse}')
114
115     return model
116

```

Рисунок 3.11. Загальний вигляд системи

У цій системі використовується **модель лінійної регресії (Linear Regression)**, що є одним із методів машинного навчання. Лінійна регресія — це статистичний метод, який використовують для моделювання залежності між однією цільовою змінною та однією або кількома незалежними змінними (ознаками). Вона намагається знайти лінійний зв'язок між вхідними ознаками та вихідною змінною.

Лінійна регресія: метод навчання з учителем, у якому цільова змінна (у моєму випадку — час в дорозі) представлена лінійною функцією від вхідних ознак (у даному випадку — такі фактори, як відстань, трафік, погодні умови і т. д.).

Математично лінійна регресія намагається знайти параметри моделі (коефіцієнти при ознаках), які мінімізують помилку між передбаченими значеннями та реальними значеннями (цільова змінна).

```

91 distances = np.random.uniform( low: 5, high: 50, size=100) # Випадкові відстані (км) між 5 і 50
92 traffic_levels = np.random.randint( low: 1, high: 5, size=100) # Рівень трафіку від 1 до 4
93 weather_conditions = np.random.randint( low: 1, high: 3, size=100) # Погодні умови (1 - сприятливі, 2 - несприятливі)
94 days_of_week = np.random.randint( low: 0, high: 6, size=100) # Дні тижня від 0 до 6
95 num_stops = np.random.randint( low: 3, high: 10, size=100) # Кількість зупинок
96

```

Рисунок 3.12. Генерація даних

Тут генеруються випадкові дані для симуляції різних умов маршруту:

- **distances:** масив з випадковими відстанями між точками маршруту в межах 5–50 км.
- **traffic_levels:** рівень трафіку, від 1 до 4.
- **weather_conditions:** погодні умови, де 1 — сприятливі, а 2 — несприятливі.
- **days_of_week:** день тижня (від 0 до 6), де 0 — понеділок, а 6 — неділя.
- **num_stops:** кількість зупинок на маршруті, від 3 до 10.

```

97 # Цільова змінна – реалістичний час у дорозі з урахуванням різних факторів
98 y = distances * 2 + traffic_levels * 10 + weather_conditions * 15 + np.random.normal( loc: 0, scale: 10, size=100)
99

```

Рисунок 3.13. Цільова змінна

Це формула для генерації цільової змінної y , що є часом у дорозі. Час у дорозі залежить від кількох факторів:

- Відстань: множиться на коефіцієнт 2, оскільки чим більша відстань, тим більше часу потрібно для подолання маршруту.
- Рівень трафіку: множиться на 10, що показує, як трафік впливає на час у дорозі.
- Погодні умови: множиться на 15, адже несприятливі погодні умови можуть значно збільшити час подорожі.

- Додається випадкова похибка з нормальним розподілом, щоб імітувати реальні умови (невизначеність у часі).

```

100 # Об'єднання всіх ознак у матрицю
101 X = np.column_stack([distances, days_of_week, traffic_levels, weather_conditions, num_stops])
102

```

Рисунок 3.14. Формування матриці ознак

Матриця ознак **X** складається з кількох змінних:

- Відстань
- День тижня
- Рівень трафіку
- Погодні умови
- Кількість зупинок

Ці ознаки використовуються для навчання моделі, щоб передбачити цільову змінну.

```

103 # Поділ даних на тренувальну та тестову вибірку
104 X_train, X_test, y_train, y_test = train_test_split(*arrays: X, y, test_size=0.2, random_state=42)
105

```

Рисунок 3.15. Поділ даних на тренувальну та тестову вибірки

Дані розбиваються на **тренувальну (80%)** та **тестову (20%)** вибірки. Це робиться для того, щоб модель навчалась на одних даних, а потім перевіряла свою точність на інших.

train_test_split — це функція з бібліотеки `sklearn`, яка випадковим чином розбиває дані.

```

106 # Створення та навчання моделі лінійної регресії
107 model = LinearRegression()
108 model.fit(X_train, y_train)

```

Рисунок 3.16. Створення та навчання моделі

Створюється модель лінійної регресії, яка буде вивчати зв'язок між ознаками (**X**) та цільовою змінною (**y**).

fit — це метод, який навчає модель на тренувальних даних, щоб вона змогла знаходити залежність між вхідними та вихідними змінними.

```
110     # Оцінка точності моделі
111     y_pred = model.predict(X_test)
112     mse = mean_squared_error(y_test, y_pred)
113     print(f'Mean Squared Error: {mse}')
114
```

Рисунок 3.17. Оцінка точності моделі

Після навчання моделі, її точність перевіряється на тестових даних.

predict — метод для передбачення значень на основі тестових даних.

mean_squared_error — це метрика для оцінки точності моделі. Вона обчислює середньоквадратичну помилку (MSE), яка вимірює середнє квадратичне відхилення між реальними та передбаченими значеннями.

В кінці прописується `return model` для повернення моделі з функції, щоб її можна було використовувати для передбачення часу маршруту у майбутньому.

Лінійна регресія добре підходить для задач, де існує лінійна залежність між вхідними ознаками та цільовою змінною. У нашому випадку час у дорозі залежить від кількох факторів (відстань, трафік, погода і т. д.), і лінійна регресія намагається знайти, як кожен із цих факторів впливає на кінцевий час.

Розділ 4. ОЦІНКА ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОЇ СИСТЕМИ ДЛЯ ПОКРАЩЕННЯ ЛОГІСТИЧНИХ ПРОЦЕСІВ

4.1 Оцінка впливу системи на швидкість прийняття рішень.

Оцінку впливу ІІІ-системи для автоматизації прийняття рішень у логістиці на швидкість прийняття рішень можемо провести з урахуванням її ключових компонентів і функціональностей, про які ми раніше згадували. Ось як ми можемо оцінити в контексті системи:

1. Вимірювання часу обробки даних у реальному часі

Модуль збору та обробки даних збиратиме інформацію з різних джерел, таких як транспортні засоби, склади, а також дані про погодні умови і трафік. Вимірювання швидкості цього модуля покаже, як швидко система оновлює інформацію, що важливо для ухвалення рішень. Реальний вплив на швидкість рішень можна оцінити шляхом порівняння часу, витраченого на обробку, з цільовими показниками.

2. Оцінка оптимізації в модулі ухвалення рішень

Оскільки система включає автоматизований модуль прийняття рішень, ми можете оцінити, наскільки швидко він обирає оптимальні маршрути чи алгоритми розподілу ресурсів у порівнянні з традиційними методами. Наприклад, у сценаріях перерозподілу запасів або зміни транспортних маршрутів система повинна адаптуватися та видавати рекомендації за кілька хвилин.

3. Аналіз результатів роботи оптимізаційного модуля

Модуль оптимізації використовує алгоритми, як-от мурашиний алгоритм чи генетичні алгоритми, для швидкої оптимізації маршрутів. Вплив цього модуля на швидкість рішень можна оцінити через зменшення часу на планування та побудову маршрутів. Порівняння продуктивності системи до і після її впровадження, а також вимірювання середнього часу виконання оптимізаційного алгоритму для реальних даних, дадуть чітке уявлення про її ефективність.

4. Реальний вплив на швидкість моніторингу та реагування

Модуль моніторингу та автоматизації у системі забезпечує відстеження ситуації в реальному часі. Швидкість моніторингу, зокрема здатність системи швидко реагувати на інциденти (наприклад, затримки в транспорті або непередбачувані зміни попиту), має вирішальне значення для швидкості прийняття рішень.

Після впровадження системи можна також оцінити її вплив на бізнес-метрики, такі як середній час доставки або час перерозподілу товарів на складах. Це дасть нам всебічне розуміння того, як система оптимізує процеси і зменшує час, необхідний на прийняття рішень у різних логістичних сценаріях.

5. Оцінка точності прогнозів та її вплив на швидкість

Зниження похибки прогнозів (наприклад, середньоквадратичної помилки) з кожним новим запуском означає, що наша модель навчання з часом стає більш точною. Це дозволяє приймати рішення швидше, оскільки зростає впевненість у правильності прогнозів, що зменшує потребу в додаткових перевірках.

6. Адаптивність системи до змін у реальному часі

У контексті логістики важливо, щоб система швидко адаптувалася до будь-яких змін, таких як зміни в погодних умовах або нова інформація про трафік. Створена система може забезпечувати швидке оновлення маршрутів та рекомендацій на основі нових даних, і цей фактор може бути ключовим для оцінки швидкості прийняття рішень.

4.2 Оцінка економічного ефекту та потенційних вигод для підприємства

Розглянемо економічний ефект та потенційні вигоди для підприємства після вводу системи автоматизації прийняття рішень.

1) Зменшення витрат на логістику

Система оптимізації транспортних маршрутів (на основі алгоритму мурашиної колонії) дозволяє ефективно знижувати витрати, пов'язані з логістичними операціями. Завдяки оптимізації маршрутів підприємство може:

- **Оптимізувати маршрути**, тим самим знизити витрати на паливе, обслуговування транспорту та амортизацію техніки. Це дозволяє зменшити

транспортні витрати на певний відсоток залежно від кількості маршрутів і частоти перевезень.

- Оптимізувати графіки доставки, що дозволить уникнути простоїв та скоротити час доставки.
- Використовувати менше ресурсів для доставки, що позитивно впливає на загальні операційні витрати.

2)Покращення швидкості прийняття рішень

Автоматизація процесів прийняття рішень знижує час, який витрачається на аналіз даних та прийняття відповідних заходів.

Впровадження такої системи забезпечує:

- Миттєве реагування на зміну умов, наприклад, на підвищення попиту або зміну погодних умов, що впливає на логістику.
- Вчасне прийняття рішень, що підвищує конкурентоспроможність підприємства і знижує ризики пов'язані з можливими затримками.
- Система дозволяє приймати рішення швидше та з меншою кількістю помилок. Це забезпечує скорочення часу простоїв, очікувань та неефективних операцій.
- **Економія за рахунок зменшення простоїв:** можна оцінити як зниження непрямих витрат на простої обладнання та персоналу.

3)Зростання доходів через поліпшення рівня обслуговування

Система дозволяє більш точно планувати терміни доставки та зменшувати їх відхилення. Це значно підвищує рівень обслуговування клієнтів та впливає на дохід:

- Завдяки своєчасній доставці та оптимальному обслуговуванню клієнтів можна підвищити рівень задоволеності та утримання клієнтів, що позитивно вплине на дохід підприємства.
- Підвищується надійність поставок, що дозволяє клієнтам отримувати товари в обумовлені терміни.
- Формула для розрахунку додаткового доходу може виглядати так:

$$\Delta D \text{ клієнти} = N \text{ нові клієнти} \times R \text{ середній дохід}$$

де N нові клієнти — приріст клієнтів завдяки покращенню обслуговування, R середній дохід — середній дохід з одного клієнта.

4) Потенціал для масштабування та гнучкість

- Використання модульної архітектури надає можливість розширення системи без значних додаткових витрат, а також забезпечує адаптивність під нові бізнес-вимоги.
- Це знижує потребу в частих модернізаціях, що в перспективі може заощадити кошти та ресурси.

5) Потенційні економічні вигоди в цифрах

Вартість впровадження такої системи може бути значною, але її окупність проявляється в економії часу і ресурсів:

- **Економія на витратах на транспорт:** до 10-15% через оптимізацію маршрутів.
- **Зменшення затрат на зберігання:** до 5-10% через зниження зайвих запасів.
- **Підвищення продуктивності праці:** працівники логістичного відділу можуть зосередитися на стратегічних завданнях замість виконання рутинних задач.

Потенційні вигоди для підприємства

1. Покращення обслуговування клієнтів:

- Оптимізовані маршрути та краще прогнозування попиту дозволяють вчасно задовольняти потреби клієнтів. Це підвищує рівень обслуговування та знижує кількість затримок у постачаннях, що може збільшити лояльність клієнтів.

2. Гнучкість і адаптивність:

- Система адаптується до змін в умовах на ринку чи на складі, допомагаючи швидко реагувати на коливання попиту чи зміну транспортних умов.

3. Збільшення конкурентоспроможності:

- Зниження витрат на логістику та підвищення ефективності сприяє зниженню собівартості продукції, що дає можливість встановлювати більш конкурентоспроможні ціни або збільшувати маржу.

4. Зменшення вуглецевого сліду:

- За рахунок оптимізації маршрутів можна зменшити витрати на паливе та, відповідно, знизити рівень викидів CO₂, що покращує екологічний імідж компанії.

Такий економічний ефект, доповнений вигодами для підприємства, допомагає створити значну додану вартість для підприємства, покращуючи його фінансові показники, імідж та збільшуючи конкурентоспроможність на ринку.

Розділ 5. ЕРГОНОМІКА ІТ ТА ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ РОЗРОБКИ. ФОРМУВАННЯ РЕКОМЕНДАЦІЙ ЩОДО ВПРОВАДЖЕННЯ ТА МАСШТАБУВАННЯ РОЗРОБЛЕНОЇ СИСТЕМИ НА РЕАЛЬНІ ВИРОБНИЧІ ПРОЦЕСИ

5.1 Вимоги до програмного забезпечення та основні підходи до його проектування з погляду користувача

Розробка системи автоматизації прийняття рішень для логістики повинна базуватися на ключових вимогах і підходах, що забезпечують ефективність, точність прогнозування, безперервну роботу в режимі реального часу, доступність, зручність та безпеку даних. Нижче описано основні вимоги, що сприятимуть високій якості та надійності системи.

→ Інтуїтивно зрозумілий інтерфейс користувача:

Інтерфейс повинен бути структурованим та легким для розуміння. Це дозволить користувачам з легкістю переглядати інформацію про маршрути, прогнози, час доставки та інші критично важливі дані. Зручність у використанні дозволяє операторам та менеджерам швидко знаходити необхідні інструменти й функції, що прискорює процес прийняття рішень.

→ Прогнозування та оптимізація в режимі реального часу:

Система має підтримувати обробку даних у реальному часі, що є надзвичайно важливим для логістики, де кожна хвилина може впливати на своєчасність доставок. Відповідно, система повинна миттєво оновлювати дані про рух транспорту, стан інвентарю, прогнози попиту та інші фактори, що впливають на управління ланцюгом постачання.

→ Гнучкість і масштабованість:

Система повинна мати можливість масштабуватися, щоб підтримувати різну кількість точок доставки, об'єм інвентарю та складність маршрутів. Гнучкість забезпечує можливість адаптуватися до різних потреб компанії, дозволяючи модифікувати параметри маршрутів, додавати нові фактори для аналізу тощо.

→ Точність прогнозів та оптимізації маршрутів:

Використання алгоритму мурашиної колонії в системі забезпечує оптимальне прокладання маршрутів, що зменшує витрати на паливо та знижує затримки доставки. Алгоритми прогнозування мають враховувати такі фактори, як попит, погода, трафік, що дозволяє приймати точніші рішення.

→ Захист даних і конфіденційність:

Дані, пов'язані з маршрутами, клієнтами, запасами та іншими аспектами логістичних операцій, мають бути захищеними від несанкціонованого доступу. Система повинна використовувати надійні методи шифрування даних та аутентифікації користувачів, що гарантує безпеку інформації навіть при роботі через зовнішні канали передачі.

→ Мінімізація часу на виконання завдань:

Здатність системи швидко виконувати алгоритмічні обчислення є критичною для прийняття своєчасних рішень. Це включає швидку обробку вхідних даних для отримання ефективних маршрутів, а також прогнозів на основі машинного навчання щодо часу доставки або потреби в ресурсах.

→ Продуктивність і ефективне використання ресурсів:

Система має бути оптимізованою для ефективного використання серверних та обчислювальних ресурсів, що дозволить швидко обробляти великі об'єми даних. Це особливо важливо для систем, що працюють з великим числом пунктів доставки або високим рівнем трафіку даних.

5.2 Ергономічні цілі і показники якості програмного продукту

Ергономічні цілі створюваної системи автоматизації прийняття рішень у логістиці зосереджені на забезпеченні зручності, швидкості доступу до функцій та мінімізації зусиль, необхідних для взаємодії користувачів із програмним забезпеченням.

★ **Інтуїтивність інтерфейсу.** Користувачі мають легко знаходити функції, такі як оптимізація маршрутів, моніторинг ресурсів чи прогнозування попиту.

Інтерфейс повинен забезпечувати зрозумілу навігацію з логічним розташуванням кнопок, форм і графічних елементів.

- ★ **Комфортність використання.** Система має бути зручною як для працівників складів, так і для менеджерів логістики. Це включає можливість адаптації робочого середовища до персональних уподобань, таких як вибір режиму відображення даних.
- ★ **Адаптивність.** Система повинна коректно працювати на різних пристроях (десктопах, планшетах), забезпечуючи однаковий досвід роботи незалежно від платформи чи роздільної здатності екрана.
- ★ **Ефективність взаємодії.** Інтерфейс має забезпечувати виконання основних операцій (наприклад, внесення даних про ресурси чи запуск аналітики) у мінімальну кількість кроків.
- ★ **Зворотний зв'язок.** Кожна дія користувача, наприклад, зміна налаштувань чи запуск алгоритму оптимізації, має супроводжуватися чіткими повідомленнями про успішне виконання або попередженнями про помилки.

Реалізація цих цілей і відповідність показникам якості дозволить створити систему, яка буде не лише функціональною, а й максимально зручною для кінцевих користувачів, що сприятиме підвищенню продуктивності праці та задоволенню потреб підприємства.

5.3 Основні характеристики, що враховуються при розробці інтерфейсу користувача

При розробці інтерфейсу користувача для системи автоматизації прийняття рішень у логістиці враховуються ключові характеристики, що забезпечують зручність, доступність і ефективність роботи з програмним забезпеченням.

❖ Консистентність дизайну.

Інтерфейс повинен мати єдиний стиль у всіх модулях системи: однакові шрифти, кольори та розташування кнопок. Це дозволить користувачам швидко

орієнтуватися у функціоналі незалежно від виконуваних завдань (оптимізація маршрутів, моніторинг чи аналітика).

❖ **Інтуїтивність.**

Використання зрозумілих позначень, іконок та підказок є критично важливим. Наприклад, кнопка для запуску алгоритму оптимізації має бути чітко позначеною та розташованою у логічному місці. Інтуїтивність знижує час навчання користувачів роботі з системою.

❖ **Естетичний**

вигляд.

Важливо створити приємний та професійний дизайн, що сприяє комфорту користувачів. Вибір кольорової гами повинен забезпечувати гарну видимість на моніторах різного типу.

❖ **Ефективність.**

Інтерфейс повинен бути побудований так, щоб виконання основних завдань (наприклад, перегляд звітів або редагування даних) займало мінімум часу та кроків. Це включає автоматичне заповнення форм, фільтри для даних тощо.

❖ **Доступність.**

Система повинна відповідати принципам універсального дизайну, щоб бути зручною для всіх користувачів, включаючи людей із обмеженими можливостями. Це передбачає підтримку масштабування тексту, використання клавіатурних шорткатів і підтримку екранних читалок.

5.4 Техніко-економічне обґрунтування розробки

Для успішної реалізації проекту зі створення системи автоматизації прийняття рішень у логістиці необхідно чітко визначити потенціал системи, оцінити її вигоди та ризики, а також спланувати етапи розробки й впровадження. Даний підрозділ містить докладний опис проекту, починаючи від резюме і закінчуючи оцінкою ризиків та страхуванням. У ньому визначені ключові аспекти, які допоможуть оцінити ринковий потенціал додатку, зрозуміти його конкурентні

переваги, розробити ефективну маркетингову стратегію та забезпечити фінансову стабільність проекту.

Резюме проекту

Назва: SmartChain

Місце розташування: Система є багатоплатформним рішенням, яке розгортається як хмарний сервіс для підприємств. Вона доступна через веб-інтерфейс для ПК та мобільних пристроїв, а також інтегрується у внутрішні ERP-або TMS-системи підприємств. Основна географія застосування — логістичні компанії та виробничі підприємства у великих промислових регіонах світу.

Мета: Основною метою проекту є суттєве скорочення витрат на логістику, зменшити час на прийняття рішень, покращити точність прогнозів і забезпечити оперативність у змінних умовах ринку. Це буде досягнуто шляхом впровадження зручної та надійної системи штучного інтелекту для автоматизації прийняття рішень.

Суть проекту: Проект передбачає розробку інноваційної системи автоматизації прийняття рішень у логістиці, яка використовує сучасні алгоритми оптимізації та прогнозування для забезпечення ефективного управління ланцюгами постачання, транспортуванням та розподілом ресурсів. Система орієнтована на автоматизацію складних процесів у сфері логістики з метою підвищення точності, зменшення витрат і прискорення прийняття рішень. Система пропонує широкий спектр функціональних можливостей для забезпечення зручності та безпеки користувачів:

- 1. Пошук і створення маршрутів доставки:** Користувачі можуть вводити початкові та кінцеві точки маршруту, наприклад, склад та кілька магазинів. Система автоматично генерує можливі маршрути, використовуючи оптимізаційні алгоритми (мурашина колонія). Користувачі можуть шукати вже створені маршрути інших користувачів, якщо це необхідно, наприклад, для економії часу або при використанні стандартних маршрутів.

2. **Оптимізація маршрутів для кількох транспортних засобів:** Якщо система використовується для управління кількома транспортними засобами (наприклад, для доставки в кілька точок одночасно), вона може автоматично розподіляти маршрути між транспортними засобами, щоб мінімізувати загальну відстань і час. Користувачі можуть вказати кількість транспортних засобів, і система автоматично запропонує оптимальні розподіли маршрутів.
3. **Прогнозування часу маршруту:** Користувачі можуть отримати точну оцінку часу, необхідного для виконання маршруту, з урахуванням безлічі факторів, таких як відстань, трафік, погодні умови та день тижня. Модель машинного навчання на основі даних про маршрут та інші змінні прогнозує час у дорозі, що допомагає користувачам планувати завдання та уникати затримок.
4. **Оцінка ефективності маршруту:** Після того як система запропонує оптимальний маршрут, користувачі можуть отримати звіт про його ефективність, включаючи загальну довжину маршруту та передбачуваний час у дорозі. Це допомагає прийняти рішення про те, який маршрут обрати.
5. **Інтерактивні сповіщення та оновлення:** Система може інформувати користувача про значні зміни в маршруті, наприклад, про підвищення рівня трафіку або зміну погодних умов, що може вплинути на час доставки.
6. **Спрощення планування:** Автоматичне генерування маршрутів та розрахунок часу дозволяють користувачеві легко планувати логістичні процеси без необхідності вручну рахувати маршрути та трафік.
7. **Забезпечення безпеки даних:** Користувачі можуть бути впевнені в захисті своїх даних завдяки вбудованим системам шифрування та безпеки, що відповідають сучасним вимогам щодо захисту інформації.

Переваги SmartChain:

- Оптимізація маршрутів та зниження витрат на транспорт завдяки використанню алгоритмів оптимізації, таких як мурашина колонія, система автоматично визначає найефективніші маршрути для транспорту, що

дозволяє знизити витрати на паливо та час, зменшуючи кількість порожніх пробігів і знижуючи витрати на обслуговування транспортних засобів.

- Автоматизоване прийняття рішень дозволяє уникнути людських помилок при розрахунках маршрутів, оптимізації вантажоперевезень і прогнозуванні термінів доставки, що забезпечує більш точні та надійні результати.
- Використання машинного навчання для прогнозування часу на маршруті і обробки різних параметрів, таких як погодні умови, рівень трафіку та інші фактори, дозволяє зменшити час очікування і підвищити ефективність роботи транспортної мережі.
- Завдяки оптимізації транспортних маршрутів та зменшенню кількості порожніх пробігів, система допомагає знизити викиди CO₂ і зменшити негативний вплив на навколишнє середовище.
- З автоматизованим плануванням доставок і прогнозуванням часу прибуття, клієнти можуть отримувати точніші терміни доставки і знижувати рівень затримок, що значно підвищує їх задоволеність.
- Система може легко масштабуватися в залежності від зростання потреб підприємства, дозволяючи адаптуватися до змін у вимогах або збільшенні обсягів перевезень.

Цільова аудиторія:

- ❖ **Логістичні компанії:** Підприємства, що займаються вантажними перевезеннями і оптимізацією транспортних процесів. Ці компанії можуть скористатися системою для покращення ефективності управління своїм автопарком і логістичними маршрутами, зниження витрат на паливо та підвищення швидкості доставки.
- ❖ **Малі та середні підприємства:** Бренди, які мають потребу в ефективному управлінні поставками та транспортними засобами, зокрема для доставки товарів до клієнтів або між складами. Вони можуть за допомогою цієї системи оптимізувати ланцюг поставок і зменшити витрати.

- ❖ **Інтернет-магазини та рітейлери:** Компанії, що займаються електронною комерцією і мають необхідність в ефективному управлінні запасами та доставкою товарів. Для них система автоматизованого планування маршрутів і управління запасами буде надзвичайно корисною, забезпечуючи швидку і економічну доставку.
- ❖ **Офісні працівники та керівники:** Люди, які займаються управлінською діяльністю та часто потребують швидкого і точного розрахунку часу доставки та логістичних процесів для ефективного управління компанією або підрозділом.
- ❖ **ІТ-компанії та стартапи:** Організації, які працюють з великими обсягами даних і мають потребу в автоматизації процесів прийняття рішень для зниження затрат та підвищення швидкості роботи. Вони можуть використовувати таку систему для інтеграції в свої платформи та поліпшення внутрішніх процесів.
- ❖ **Міські транспортні компанії:** Оператори громадського транспорту та доставки, які можуть інтегрувати цю систему для більш ефективного планування маршрутів і зменшення заторів у великих містах.
- ❖ **Екологічно свідомі споживачі:** Особи та організації, які зацікавлені в зменшенні свого екологічного впливу. Вони будуть зацікавлені в зменшенні викидів CO₂ та зменшенні витрат на паливо, що є важливим аспектом системи автоматизації в логістиці.

Очікувані результати:

- Зменшення заторів на дорогах завдяки більш ефективному плануванню маршрутів для транспорту
- Зниження витрат на паливо, обслуговування та утримання транспортних засобів
- Зменшення викидів CO₂ та інших шкідливих речовин
- Скорочення часу доставки, що покращує обслуговування клієнтів і знижує ризик затримок

- Забезпечення більш точного управління логістичними потоками
- Полегшення взаємодії між логістичними компаніями, підприємствами та споживачами
- Зменшення навантаження на транспортні мережі

Резюмуючи все вище сказане, цей проект має великий потенціал не лише в економічному, але й у соціальному та екологічному аспектах. Завдяки автоматизації логістичних процесів, оптимізації маршрутів та ефективному використанню транспортних ресурсів, він може значно зменшити затори, знизити витрати на паливо і обслуговування, а також сприяти покращенню екологічної ситуації в містах. Крім того, він може забезпечити підвищення якості обслуговування та зручності для користувачів, що в кінцевому рахунку підвищить рівень життя міських жителів.

Загалом, система автоматизації прийняття рішень в логістиці може стати важливим кроком до сталого розвитку міських транспортних і логістичних систем.

5.5. Опис проектованого продукту

Проектований продукт: система III SmartChain

SmartChain – це система, яка є інноваційним рішенням для автоматизації процесів прийняття рішень у сфері логістики, орієнтовану на оптимізацію ресурсів, планування маршрутів та управління запасами в реальному часі. Система використовує штучний інтелект для прогнозування попиту, оптимізації транспорту та складів, а також для оцінки ефективності різних логістичних операцій.

Призначена для компаній, які займаються логістикою, зберіганням та доставкою товарів, система забезпечує ефективне управління всіма етапами логістичних ланцюгів, включаючи планування, а також оптимізацію маршрутів для зниження витрат на транспорт і підвищення загальної продуктивності. Тим самим заощаджує гроші та підіймає валовий прибуток компаній, а також поліпшує якість обслуговування клієнтів.

Основні функції та можливості SmartChain:

- Оптимізація маршрутів та управління транспортними засобами:

Користувачі системи можуть автоматично планувати оптимальні маршрути для транспорту, з урахуванням різних параметрів, таких як відстань, дорожні умови, рівень трафіку та вартість пального. Система використовує алгоритми оптимізації, зокрема методи мурашиної колонії та машинного навчання, для пошуку найбільш ефективних шляхів доставки.

- Прогнозування часу доставки:

Модель машинного навчання (лінійна регресія) прогнозує час доставки в залежності від таких факторів, як відстань, рівень трафіку, погодні умови, день тижня та кількість зупинок на маршруті. Прогнозування часу доставки дає можливість точно оцінити витрати часу на кожен маршрут і планувати процеси логістики більш ефективно.

- Автоматичне оновлення маршрутів:

Система здатна автоматично оновлювати маршрути в разі зміни умов (наприклад, затори, погодні умови), щоб забезпечити максимальну ефективність доставки.

- Інтеграція з іншими системами:

Для забезпечення гнучкості і безперебійної роботи система має можливість інтеграції з іншими бізнес-платформами, такими як ERP або CRM-системи, що дозволяє отримувати актуальну інформацію про замовлення, ресурси та транспорт.

- Аналіз і рекомендації для оптимізації:

Після збору даних і їх аналізу, система надає рекомендації для оптимізації логістичних процесів, таких як змінення графіків доставки або вибір інших маршрутів для зниження витрат та часу на транспортування.

Користувацький інтерфейс

Система має чистий і функціональний інтерфейс, що дозволяє користувачам швидко і зручно взаємодіяти з усіма модулями системи. Всі екрани мають чітке меню навігації і прості форми введення, що знижує час на адаптацію до нових

користувачів. Інтерфейс дозволяє не тільки керувати поточними поїздками, але й отримувати прогностичні дані для оптимізації майбутніх маршрутів, що дозволяє максимально ефективно використовувати транспортні ресурси.

Оцінка ринку збуту

Оцінка ринку збуту для системи автоматизації прийняття рішень у логістиці (SmartChain) є критично важливим етапом, оскільки вона дозволяє визначити потенційний попит на продукт, зрозуміти конкурентне середовище, окреслити цільову аудиторію та передбачити можливі виклики. Нижче представлена детальна оцінка ринку збуту для "Common Way".

Аналіз цільової аудиторії

Аналіз цільової аудиторії приведено в таблиці 5.1.

Таблиця 5.1 – Аналіз цільової аудиторії

Вік	Керівники середньої та вищої ланки (35-55 років), логісти, аналітики, ІТ-фахівці, які відповідають за оптимізацію процесів у компаніях.
Місце проживання	Великі міста та промислові зони, де концентрується значна кількість логістичних операцій, а також регіони з активною торгівлею та виробництвом.
Професійний статус	Логістичні компанії, ритейл, виробники, дистриб'ютори, оператори складських і транспортних послуг, які мають значні обсяги перевезень і потребують автоматизації процесів.
Мотиви	Скорочення витрат на логістику, підвищення точності планування, зменшення часу простоїв, оптимізація транспортних маршрутів, підвищення конкурентоспроможності.

Географічний аналіз ринку

Основні регіони для впровадження системи автоматизації прийняття рішень у логістиці:

- Європа

Система буде найбільш ефективною у містах з розвинутою транспортною інфраструктурою та високим рівнем ділової активності. Київ, Львів, Варшава, Берлін, Амстердам і Париж є перспективними через необхідність оптимізації маршрутів доставки, зменшення транспортних витрат та підвищення ефективності роботи логістичних компаній.

– Північна Америка

У великих містах, таких як Нью-Йорк, Лос-Анджелес, Чикаго та Торонто, актуальність впровадження системи обумовлена великим обсягом вантажоперевезень і складною транспортною ситуацією. Система дозволить мінімізувати затримки, пов'язані із заторами, і покращити координацію в ланцюгах постачання.

– Азія

Мегаполіси, такі як Токіо, Сеул, Шанхай, та Сінгапур, демонструють високий попит на рішення для логістичної автоматизації через стрімке зростання електронної комерції та обмеження простору для транспортування. Система допоможе компаніям зменшити витрати на логістику, оптимізувати маршрути та забезпечити своєчасність доставок.

– Південна Америка

Такі міста як Сан-Паулу, Буенос-Айрес, Богота мають складну логістичну інфраструктуру через щільний міський трафік та потребують оптимізації транспортування товарів у міських умовах. Система може знайти попит серед компаній, які займаються доставкою в густонаселених районах.

– Африка

Багато африканських мегаполісів, такі як, Лагос, Кейптаун, Найробі стикаються з проблемами поганої інфраструктури та хаотичного руху. Система, яка автоматизує логістичні рішення, допоможе зменшити витрати та оптимізувати маршрути, що зробить її привабливою для місцевого бізнесу.

– Океанія

Основні регіони: Сідней, Мельбурн, Окленд.

Високий рівень розвитку транспорту та зростаючі очікування споживачів щодо швидкості доставки створюють попит на технологічні рішення, які підвищують ефективність логістики.

Розширення ринку

- Середні та малі міста

Після оптимізації роботи системи у мегаполісах доцільно спрямувати увагу на середні та малі міста. Тут також є значний потенціал, особливо для малих і середніх підприємств (МСП), які стикаються з логістичними проблемами:

Недостатня кількість ресурсів для ефективного управління логістикою. Відсутність доступу до складних IT-рішень через обмежений бюджет. Потреба в локальній оптимізації (наприклад, доставка продуктів або товарів у віддалені населені пункти).

- Близький схід

Такі міста, як Дубай, Ер-Ріяд, Доха де є розвинена транспортна інфраструктура, що потребує сучасних цифрових рішень для підвищення ефективності.

- Глобальні тренди

Вихід на ринки, що розвиваються (наприклад, Індія, Африка), де активно розширюється електронна комерція і зростає потреба в сучасних логістичних рішеннях. Співпраця з міжнародними партнерами, такими як великі транспортні компанії, для створення екосистеми взаємодії. Участь у міжнародних виставках та конференціях для демонстрації можливостей системи.

Аналіз попиту

1. Екологічна свідомість

Сучасні компанії дедалі більше орієнтуються на зменшення вуглецевого сліду та впровадження екологічно чистих технологій. Використання автоматизованих систем, сприяє оптимізації маршрутів перевезень, що зменшує споживання пального та викиди CO₂. Це особливо актуально для компаній, які працюють у регіонах з екологічними обмеженнями чи прагнуть покращити свій "зелений" імідж.

2. Економічний фактор

Зростання витрат на логістику, включаючи вартість пального, обслуговування транспортних засобів і оплату праці, мотивує підприємства до впровадження рішень, які підвищують ефективність. Створена система дозволяє знизити операційні витрати шляхом оптимізації маршрутів, автоматизації складування та зменшення часу на прийняття рішень.

3. Реакція на ринкові виклики

У сучасних умовах глобалізації бізнес стикається з постійно зростаючим попитом на оперативність та адаптивність. Система автоматизації забезпечує точне управління ресурсами, дозволяючи компаніям швидко реагувати на зміну ринкових умов, знижувати ризики дефіциту чи надлишку товарів.

4. Популярність цифрових рішень

Збільшення популярності технологій штучного інтелекту (ШІ) та автоматизації стимулює попит на інноваційні логістичні системи. Інтуїтивно зрозумілий інтерфейс та інтеграція системи з іншими бізнес-процесами роблять її зручною та доступною навіть для компаній, які раніше не використовували подібні рішення.

5. Потреба у підвищенні конкурентоспроможності

В умовах конкуренції підприємства прагнуть зменшити витрати та покращити якість послуг. Наша система дозволяє не лише оптимізувати витрати, але й підвищити швидкість обслуговування клієнтів за рахунок автоматизації та точного прогнозування.

Висновки: оцінка ринку збуту показує, що "SmartChain" демонструє значний потенціал для успішного впровадження та розвитку системи автоматизації прийняття рішень у логістиці. Постійне зростання попиту на ефективні, економічні та екологічні рішення в управлінні логістичними процесами створює сприятливі умови для інтеграції подібних систем у роботу підприємств. Правильна адаптація функціоналу до специфіки окремих регіонів, а також постійне вдосконалення та інтеграція з новими технологіями дозволять системі не лише зайняти стабільну позицію на ринку, але й стати одним із лідерів у сфері автоматизації логістики.

Стратегія маркетингу

Стратегія маркетингу є ключовим аспектом успішного запуску та розвитку будь-якого програмного продукту, включаючи систему автоматизації прийняття рішень у логістиці "SmartChain". В умовах зростаючої конкуренції та швидко змінюваних ринкових тенденцій, ефективна маркетингова стратегія дозволяє не лише залучити нових користувачів, але й утримувати їх, забезпечуючи сталий розвиток сервісу.

Цей підрозділ присвячений детальному аналізу методів просування додатку на ринку. Розглянемо основні аспекти, які сприятимуть збільшенню впізнаваності бренду, залученню цільової аудиторії та підвищенню лояльності користувачів.

Методи просування

– Соціальні мережі (Facebook, Instagram, LinkedIn, X(Twitter))

Для просування в соціальних мережах використовують три основні стратегії: таргетована реклама, контент-маркетинг, інтерактивні кампанії

Таргетована реклама у таких платформах, як LinkedIn, Facebook і Twitter, дозволяє орієнтуватися на менеджерів із логістики, підприємців та керівників компаній. Така реклама базується на професійних інтересах, галузевій приналежності та географічному розташуванні, що дає змогу залучати аудиторію, яка найбільше зацікавлена у впровадженні технологій для автоматизації.

Контент-маркетинг включає створення та публікацію експертних статей, блогів, відео та кейс-стаді про використання системи для вирішення актуальних проблем у логістиці. Це не лише підвищує довіру до продукту, а й демонструє його практичну цінність.

Інтерактивні кампанії, наприклад, онлайн-вебінари, віртуальні демонстрації роботи системи та безкоштовні консультації, допомагають потенційним клієнтам ознайомитися з функціоналом системи та зрозуміти її переваги. Використання таких методів сприяє залученню нових користувачів і формуванню довгострокових партнерських відносин.

– Партнерства з професійними асоціаціями та великими компаніями

Партнерства з професійними асоціаціями та великими компаніями відіграють ключову роль у просуванні системи автоматизації прийняття рішень у логістиці. Співпраця з професійними асоціаціями та галузевими мережами дозволить залучити досвідчених фахівців і керівників, які працюють у сфері логістики та управління ланцюгами поставок. Спільні семінари, тренінги, конференції та майстер-класи допоможуть підвищити обізнаність про систему серед професіоналів, а також сприятимуть створенню комунікаційної мережі з потенційними користувачами, які застосовуватимуть технології автоматизації в своїй професійній діяльності.

Партнерство з великими компаніями відкриває можливості для впровадження системи у широкомасштабних логістичних операціях. Завдяки корпоративним програмам можна адаптувати рішення під конкретні потреби компаній, наприклад, оптимізацію постачання, зменшення витрат на транспортування чи автоматизацію складування. Додатково, такі співпраці можуть включати спільну розробку аналітичних інструментів або індивідуальних модулів для автоматизації.

Залучення компаній, які прагнуть покращити свою екологічну ефективність і знизити викиди CO₂, також сприяє популяризації системи як інструменту для побудови екологічно відповідальних ланцюгів постачання. Партнерства дозволяють не лише розширити ринок збуту, але й сприяють формуванню довгострокових стратегічних відносин із ключовими гравцями ринку.

– **Рекламні кампанії в громадському транспорт**

Плакати та банери можуть значно підвищити впізнаваність системи автоматизації прийняття рішень у логістиці серед потенційних користувачів, особливо тих, хто зацікавлений у покращенні транспортування та оптимізації ресурсів. Плакати та банери в транспорті, таких як вантажівки, потяги чи порти, де логістика відіграє ключову роль, можуть демонструвати ключові переваги системи: зменшення витрат, покращення точності доставки та оптимізація маршрутів. Це привертатиме увагу як перевізників, так і компаній, які шукають нові технології для свого бізнесу.

Аудіо та відео-реклама, орієнтована на водіїв вантажівок чи працівників складів, може пояснювати, як система працює і які переваги вона приносить, наприклад, швидкість обробки даних або адаптивність до змінних умов на ринку. Такий підхід дозволить сформувати довіру до продукту та підвищить зацікавленість у його використанні.

Крім того, реклама може бути адаптована до специфічних потреб логістичних компаній. Наприклад, відеоролики з демонстрацією роботи системи в реальних умовах допоможуть потенційним клієнтам побачити її ефективність у дії. Розміщення реклами у місцях, де проходять ключові транспортні шляхи чи у вантажних вузлах, сприятиме залученню аудиторії, яка активно працює у сфері логістики.

– Підтримка та участь у місцевих екологічних ініціативах

Підтримка та участь у місцевих екологічних ініціативах може стати стратегічно важливим елементом для популяризації системи автоматизації прийняття рішень у логістиці. Співпраця з екологічними організаціями, які займаються оптимізацією ресурсів або скороченням шкідливих викидів, допоможе продемонструвати практичну користь даного рішення в контексті сталого розвитку.

Участь у спеціалізованих форумах, присвячених екологічним іноваціям у транспорті, дозволить привернути увагу до системи як до інструменту, що сприяє зниженню вуглецевого сліду завдяки оптимізації маршрутів, зменшенню порожніх рейсів та ефективному використанню транспортних засобів.

Організація спільних ініціатив, наприклад, пілотних проектів з "зеленими" логістичними компаніями або муніципальними транспортними службами, дозволить наочно продемонструвати ефективність системи. Такі проекти можуть включати інтеграцію рішення у системи моніторингу екологічних показників для оптимізації використання пального або створення інтелектуальних маршрутів для зменшення заторів.

Висновки: стратегія маркетингу для системи автоматизації прийняття рішень у логістиці “SmartChain” орієнтована на підвищення впізнаваності продукту

серед основних цільових аудиторій, зокрема підприємств, що займаються логістикою, постачальниками та транспортними компаніями. Застосування цифрових платформ для просування, співпраця з великими компаніями та організація семінарів і презентацій у сфері логістики сприятиме збільшенню зацікавленості та довіри до системи. Використання таргетованої реклами через соціальні мережі дозволить звертатися до бізнес-спільнот, які зацікавлені в автоматизації своїх процесів, оптимізації витрат і підвищенні ефективності. Партнерство з професійними асоціаціями та компаніями в галузі логістики допоможе залучити досвідчених спеціалістів і сприятиме розвитку ринку через впровадження інноваційних технологій в логістичні процеси. Підтримка екологічних ініціатив, таких як оптимізація транспорту та зниження викидів, також може стати важливою складовою маркетингової стратегії, підкреслюючи переваги системи в контексті сталого розвитку. В цілому, маркетингова стратегія сприятиме успішному просуванню системи на ринку, забезпечуючи її інтеграцію в основні логістичні процеси підприємств.

План виробництва додатку

Етап 1: Аналіз вимог та проектування архітектури системи Початковий етап включає детальний аналіз потреб користувачів та визначення основних функцій системи. Після збору вимог проводиться проектування архітектури, вибір оптимальних алгоритмів для прийняття рішень, а також визначення структури даних для обробки та зберігання необхідної інформації в пам'яті системи. Цей етап забезпечує чітке розуміння всіх наступних кроків розробки, що гарантує узгодженість та ефективність системи в майбутньому.

Етап 2: Розробка основних функціональних модулів На даному етапі реалізуються основні функції системи відповідно до визначених вимог. Це включає розробку інтерфейсу користувача, інтеграцію з іншими системами (ERP, WMS, TMS), а також реалізацію алгоритмів для оптимізації маршруту, управління запасами та прогнозування попиту. Кожен модуль тестується для впевненості в його працездатності та сумісності з іншими частинами системи.

Етап 3: Інтеграція з іншими компонентами системи Цей етап фокусується на інтеграції різних частин системи, що були розроблені на попередніх етапах. Він включає інтеграцію алгоритмів прогнозування та оптимізації з модулями прийняття рішень, а також налаштування зв'язку між різними модулями для забезпечення безперервної та ефективної роботи системи. Метою цього етапу є забезпечення злагодженої роботи всіх частин системи без необхідності інтеграції з зовнішніми сервісами або платформами.

Етап 4: Тестування та відладка системи На цьому етапі проводиться тестування системи на різних рівнях: тестування модулів, інтеграційне тестування та системне тестування. Виявлення та виправлення помилок гарантує, що система працюватиме стабільно та ефективно, забезпечуючи високий рівень точності прогнозування та оптимізації логістичних процесів.

Етап 5: Запуск пілотної версії та збір відгуків користувачів Після успішного тестування випускається пілотна версія системи для обмеженого кола користувачів. Збираються відгуки щодо функціональності, інтерфейсу та ефективності алгоритмів для прийняття рішень. Отримані дані допоможуть вдосконалити систему до її офіційного запуску.

Етап 6: Випуск стабільної версії та постійне оновлення Після успішного завершення всіх попередніх етапів та врахування отриманого фідбеку від пілотних користувачів, випускається стабільна версія системи. Після релізу система буде постійно оновлюватися, зокрема, оновлюватимуться алгоритми оптимізації та функції для інтеграції з новими платформами. Це забезпечить її адаптивність та актуальність для користувачів у довгостроковій перспективі.

Цей план виробництва дозволяє систематично та ефективно розвивати систему автоматизації прийняття рішень у логістиці, забезпечуючи її високу якість, відповідність вимогам бізнесу та успішну реалізацію на ринку. Кожен етап спрямований на досягнення максимального ефекту при розробці та впровадженні системи, починаючи з аналізу потреб користувачів і закінчуючи постійним вдосконаленням системи після запуску. Це дозволяє забезпечити її стабільну

роботу, підвищити ефективність процесів та створити конкурентні переваги на ринку логістичних технологій.

Організаційний та юридичний плани

Організаційний план

Команда проекту для системи автоматизації прийняття рішень у логістиці складається з кількох ключових ролей, які працюють спільно для досягнення цілей системи:

- **Керівник проекту**

Керівник проекту відповідає за організацію робочих процесів, координацію дій команди та взаємодію з усіма зацікавленими сторонами. Він контролює виконання завдань, забезпечує виконання бюджету проекту, вирішує проблеми, що виникають на етапах розробки, та відповідає за своєчасне досягнення поставлених цілей.

- **Розробники програмного забезпечення**

Ця група відповідає за створення та програмування функціональних можливостей системи автоматизації прийняття рішень. Вони реалізують алгоритми для оптимізації логістичних процесів, зокрема в управлінні запасами та маршрутах доставки. Розробники працюють над інтеграцією необхідних алгоритмів та забезпечують їхню ефективну роботу.

- **Аналітики даних**

Аналітики даних займаються збором та обробкою великих обсягів даних, необхідних для функціонування системи, таких як історичні дані про попит, стан запасів і маршрути доставки. Вони створюють моделі для прогнозування та аналізу, що дозволяє ефективно оптимізувати рішення в рамках логістичних задач.

- **Спеціалісти з оптимізації користувацького досвіду (UX)**

Спеціалісти з UX відповідають за створення чіткої та зрозумілої структури взаємодії користувачів з системою, щоб забезпечити максимальну ефективність у процесі використання. Вони оптимізують робочі процеси всередині системи, аналізують зручність виконання основних операцій (наприклад налаштування алгоритмів оптимізації маршрутів), працюють над мінімізацією кількості кроків для користувача, щоб знизити час, витрачений на прийняття рішень.

– **Маркетологи та спеціалісти з продажу**

Маркетологи займаються розробкою та реалізацією маркетингових стратегій для популяризації системи серед потенційних клієнтів. Вони проводять аналіз цільової аудиторії, розробляють рекламні кампанії, проводять комунікацію з підприємствами та забезпечують зворотній зв'язок для вдосконалення системи.

– **Спеціалісти з підтримки користувачів**

Ця команда забезпечує оперативну підтримку клієнтів, відповідає на технічні запити, допомагає налаштувати систему та вирішує питання, що виникають при її використанні. Вони також займаються збором відгуків користувачів і передають їх розробникам для подальшого вдосконалення продукту.

Ця команда, працюючи в тісній співпраці, мають забезпечувати успішну реалізацію та подальший розвиток системи автоматизації прийняття рішень у логістиці “SmartChain”

Юридичний план

Юридичний план включає в себе ряд важливих аспектів для забезпечення правової відповідності та захисту даних користувачів:

– **Реєстрація компанії та отримання необхідних ліцензій**

На першому етапі проекту важливо зареєструвати юридичну особу, що буде займатися розробкою та впровадженням системи. Компанія повинна отримати всі необхідні ліцензії та дозволи для роботи в сфері інформаційних технологій та автоматизації, що відповідають законодавству країни, де здійснюватиметься діяльність. Це може включати ліцензії на розробку програмного забезпечення та надання ІТ-послуг.

– **Забезпечення відповідності додатку нормам захисту даних користувачів**

Враховуючи, що система обробляє деякі технічні та інші дані, необхідні для оптимізації логістичних процесів. Для забезпечення відповідності вимогам GDPR та інших норм, важливо впровадити відповідні заходи для захисту інформації. Це може включати анонімізацію або псевдонімізацію даних, використання безпечних каналів зв'язку та надання користувачам права на доступ та видалення інформації, що стосується їхнього використання системи.

- Розробка користувацької угоди та політики конфіденційності

Розробка користувацької угоди є обов'язковим етапом для забезпечення правової безпеки проекту. Угода повинна включати умови використання системи, обов'язки користувачів та компанії, а також механізми вирішення спорів. Політика конфіденційності визначатиме, як саме обробляються та захищаються персональні дані користувачів. Ці документи повинні бути доступні для користувачів, і вони повинні давати згоду на їх умови при реєстрації в системі.

Стратегія фінансування

Стратегія фінансування для розробки системи автоматизації прийняття рішень у логістиці також передбачає кілька варіантів залучення коштів для реалізації проекту:

- Власні кошти

Використання власних коштів засновників або зацікавлених сторін є одним із варіантів для початкового етапу проекту. Засновники можуть використовувати свої особисті заощадження або залучати внутрішні ресурси компанії для фінансування початкової фази розробки та дослідження потреб користувачів у логістиці. Цей спосіб дозволяє зберегти повний контроль над проектом та ухвалювати рішення без зовнішнього впливу.

- Приватне фінансування

Залучення приватних інвесторів або венчурних капіталістів може бути необхідним для більш масштабного розвитку та впровадження технології. Інвестори можуть надати не тільки фінансування, але й експертний досвід у галузі логістики та технологій штучного інтелекту. Це допоможе прискорити етапи розробки та вивести продукт на ринок.

- Гранти та субсидії

Проект може скористатися грантами або субсидіями, доступними через державні програми або міжнародні фонди, які підтримують інноваційні технології в логістиці та штучному інтелекті. Ці фінансові ресурси можуть допомогти на етапах дослідження та тестування системи, зокрема для забезпечення екологічних аспектів і сталого розвитку технології.

– Розвиток технологічних хабів та акселераторів

Акселератори та інкубатори, орієнтовані на технологічні стартапи, можуть запропонувати не тільки фінансування, але й менторську підтримку, технічні ресурси та комерційні зв'язки. Це дозволить швидше пройти етапи тестування, вдосконалення продукту і виведення його на ринок. Деякі акселератори надають гранти або пільгові кредити для технологічних стартапів.

– Корпоративне партнерство та стратегічні інвестиції

Залучення фінансування від великих корпорацій або стратегічних партнерів, які можуть бути зацікавлені в впровадженні таких технологій у свою діяльність. Наприклад, великі логістичні компанії або інвестори, які працюють в сфері AI або логістичних технологій, можуть бути зацікавлені в партнерстві для фінансування проекту або навіть придбання частини продукту.

– Підписки та попередні замовлення

Можна запустити попередній маркетинг продукту і залучити кошти через попередні замовлення або підписки на майбутні версії системи. Це дасть можливість отримати фінансування ще до фактичного запуску продукту і дозволить сформувати початкову базу користувачів. У результаті інвестори можуть побачити попит і відразу більш охоче підтримати проект.

Висновки: стратегія фінансування є важливим інструментом для забезпечення стабільного функціонування проекту розробки додатку. Завдяки різноманітним підходам до фінансування, проект може знайти оптимальні умови для розвитку без необхідності надмірно залежати від одного джерела фінансування, що дозволить знизити ризики та забезпечити більш гнучкий процес запуску та реалізації.

Оцінка ризику та страхування

Оцінка ризиків дозволяє ідентифікувати потенційні загрози та проблеми, які можуть виникнути протягом розробки та експлуатації додатку. Це можуть бути технічні проблеми, зміни вимог користувачів, або навіть зміни у законодавстві.

На основі оцінки ризиків розробляються стратегії управління ризиками, які включають у себе уникнення, зменшення або прийняття ризиків. Це дозволяє

команді проекту заздалегідь приготуватися до можливих негативних сценаріїв і реагувати на них ефективно.

Через оцінку ризиків можна зменшити невизначеність і несподіваність у процесі розробки. Це сприяє стабільності та прогнозованій успішності проекту.

Види ризиків:

– Технічні ризики:

Наприклад, у процесі інтеграції з різними системами (якщо така інтеграція планується в майбутньому) можуть виникнути проблеми з сумісністю або стабільністю.

Зміни в технологічних вимогах або появи нових інструментів можуть вплинути на ефективність системи.

Невірні рішення, які система приймає через помилки в алгоритмах, можуть призвести до економічних втрат або зниження ефективності.

Через обробку великих обсягів даних користувачів, можливі ризики щодо збереження конфіденційності та безпеки.

– Бізнес-ризики:

Конкуренція, змагання з іншими аналогічними або дуже схожими системами може ускладнити привертання користувачів.

Монетизація, непрогнозовані складнощі у виборі оптимальної моделі монетизації (наприклад, підписка, реклама, платіжні комісії).

– Регуляторні ризики:

Якщо є зміни в законодавчих вимогах щодо обробки даних або впровадження автоматизації, це може вимагати значних змін у системі.

Невиконання вимог таких стандартів як GDPR може призвести до штрафів і втрати довіри до системи.

– Економічні ризики:

Якщо система не зможе досягти очікуваного рівня ефективності або економії, це може призвести до фінансових втрат.

Надмірні витрати на підтримку та оновлення системи можуть обмежити прибутковість.

– Організаційні ризики

Кадрові перешкоди – проблеми зі складом команди розробників, що можуть вплинути на терміни виконання проекту.

Маркетинг і просування – ризики, пов'язані з недооцінкою маркетингових витрат та стратегій для залучення користувачів.

Страхування

Для високотехнологічних систем, таких як розробка та впровадження системи автоматизації прийняття рішень у логістиці, виникають певні ризики, які потребують страхування. Це можуть бути юридичні аспекти, пов'язані з порушенням інтелектуальної власності, а також можливі помилки або недоліки в алгоритмах системи, які можуть призвести до фінансових збитків для клієнтів або партнерів. Страхування від відповідальності дозволяє покрити витрати на юридичні послуги та компенсацію шкоди.

Одним з основних ризиків для цієї системи є безпека даних, що стосуються транспортування та обробки інформації в реальному часі. Хоча система не збирає особисті дані, вона обробляє великі обсяги логістичних даних, що включають бізнес-процеси компаній і можуть містити чутливу інформацію, таку як маршрути транспортування, запасні частини та дані постачальників. Потенційні кіберзагрози, такі як хакерські атаки або витоки даних, можуть завдати серйозних фінансових і репутаційних збитків. Страхування може допомогти покрити ці витрати, забезпечуючи відшкодування збитків, відновлення даних і репутації.

Забезпечення захисту інфраструктури, такої як сервери та хмарні сервіси, на яких розгортається система, є ще одним критично важливим аспектом. Страхування охоплює ризики фізичних пошкоджень обладнання, крадіжок, природних катастроф і збоїв в роботі інфраструктури.

Наявність страхування демонструє серйозний підхід до безпеки та фінансової стабільності проекту, підвищуючи довіру з боку партнерів, замовників і інвесторів, та забезпечує ефективну готовність до вирішення можливих проблем.

Це дозволяє знизити ризики для системи та її користувачів, забезпечуючи стабільність і надійність при обробці великих даних у логістичних процесах.

Висновки: ефективне управління ризиками та належне страхування є важливими елементами для забезпечення стабільності та безперебійної роботи системи. Враховуючи специфіку обробки великих обсягів логістичних даних та потенційні загрози, що можуть виникнути в результаті технічних збоїв, кіберзагроз чи юридичних проблем, ці заходи здатні знизити можливі негативні наслідки.

Розрахунок витрат на створення програми

Під час виконання магістерської роботи необхідно було розробити систему автоматизації прийняття рішень у логістиці та розподілі виробничих ресурсів. Розрахунок економічної ефективності дозволить обґрунтувати доцільність розробки, розрахувати собівартість системи та оцінити час її окупності.

Розрахуємо вартість розробки системи.

Середня заробітна плата (Junior) складає 25 000 грн/міс, а вартість сучасного ПК складає 42000 грн. Вартість кіловат-години електроенергії складає 4,32 грн.

Витрати на допоміжні матеріали приведені в таблиці 5.2.

Таблиця 5.2 – Розрахунок витрат на «Матеріали і комплектуючі вироби»

Пункти витрат	Сума, грн./міс
1	2
Інтернет	3000,00
Всього матеріали	300,00

Вартість програми розраховується по формулі:

$$C_{\text{пр}} = (Z_{\text{зп}} + Z_{\text{сз}} + Z_{\text{зв}} + Z_{\text{е}} + Z_{\text{м}} + A_{\text{об}})T, \text{ де} \quad (5.1)$$

T – тривалість розробки, міс;

$Z_{\text{зп}}$ – основна і додаткова заробітна плата персоналу, грн.;

$Z_{\text{сз}}$ – відрахування на соціальні заходи (30% від основної і додаткової заробітної плати, грн.);

$Z_{\text{зв}}$ – загальногосподарські витрати (10% від основної заробітної плати, грн.);

$Z_{\text{е}}$ – витрати на електроенергію;

$Z_{\text{м}}$ – витрати на основні і допоміжні матеріали;

$A_{\text{об}}$ – амортизаційні відрахування на устаткування;

$Z_{\text{е}}$ при споживанні потужності 240 Вт, тривалості роботи на місяць складає $22 \cdot 8 = 176$ годин, вартість кіловат-години електроенергії 4,32 грн, з цього слідує:

$Z_e = 176 * 4,32 * 0,24 = 178 \text{ грн.}$	
Амортизаційні відрахування на устаткування розраховуються за формулою:	
$A_{об} = \frac{\text{Вартість ПК}}{\text{Срок використання (міс)}}$	(5.2)
$A_{об} = \frac{42000}{24} = 1750 \text{ грн.}$	

Витрати на розробку програмного забезпечення приведені в таблиці 5.3.

Таблиця 5.3 – Витрати на розробку програмного забезпечення

	Найменування витрат	Одиниця вимірювання	Кількість
	1	2	3
	Термін розробки	міс	2,5
	Основна та допоміжна заробітна плата	грн.	25000,00
	Відрахування на соціальні заходи	грн.	7500,00
	Загально-господарчі витрати	грн.	2500,00
	Витрати на головні та допоміжні матеріали	грн.	300,00
	Витрати на електроенергію	грн.	178,00
	Амортизаційні відрахування	грн.	1750,00

$$C_{пр} = (25000 + 7500 + 2500 + 178 + 300 + 1750) * 2,5 = 93070 \text{ грн.}$$

Розрахунок економічної ефективності

Розрахунок економічної ефективності системи автоматизації прийняття рішень у логістиці включає оцінку витрат на розробку та підтримку, порівняно з очікуваними вигодами від оптимізації логістичних процесів. Це допомагає обґрунтувати доцільність впровадження системи, розрахувати її собівартість та оцінити час окупності.

Ефективність також оцінюється на основі прогнозованого впливу системи на скорочення витрат, таких як транспортні витрати, час на прийняття рішень, а також зменшення кількості помилок у плануванні. Крім того, враховується попит на такі системи, конкурентне середовище в логістиці та можливі ризики, які можуть вплинути на успішність проекту.

Такий підхід дозволяє забезпечити обґрунтованість інвестицій і визначити потенційні шляхи розвитку системи з точки зору економічної ефективності.

Якщо враховувати, що в місяць як мінімум 3 клієнтів(підприємств) буде купляти додаток, тоді щомісячний мінімальний дохід становитиме 7500 грн. Термін окупності програмного забезпечення розраховується за наступною формулою:

$T = \frac{C_{\text{пр}}}{I}, \text{ де}$	(5.3)
---	-------

$C_{\text{сз}}$ – витрати на створення програми;

I – місячний дохід від продаж.

Тобто, максимальний термін окупності програмного забезпечення:

$T = \frac{93070}{7500} = 12 \text{ міс.}$	(5.4)
--	-------

Якщо кількість клієнтів в місяць збільшиться до 2, то термін окупності скоротиться вдвічі.

Висновки: ергономіка відіграє критичну роль у забезпеченні ефективності та зручності користування системою автоматизації прийняття рішень. Врахування принципів ергономіки дозволяє зменшити час на адаптацію користувачів до системи та покращити ефективність використання інтерфейсу. Техніко-економічне обґрунтування розробки системи дозволяє детально оцінити витрати на реалізацію та прогнозувати економічні вигоди від автоматизації, включаючи оптимізацію ресурсів і зменшення витрат на логістичні процеси. Ці аспекти разом сприяють успішному впровадженню системи в роботу підприємства та забезпечують її фінансову стійкість і конкурентоспроможність на ринку.

ВИСНОВКИ

В результаті виконання магістерської роботи було розроблено систему штучного інтелекту для прийняття рішень у логістиці та розподілі виробничих ресурсів. Ця система дозволяє користувачам оптимізувати шляхи доставки грузів враховуючи різні побічні фактори і знаходити ідеальний маршрут заощаджуючи час та гроші.

Під час роботи над магістерською роботою були пройдені такі етапи:

- Детальне дослідження предметної області, аналіз готових рішень та постановка задачі

Було проведено ретельний аналіз предметної області для визначення потреб та вимог користувачів у сфері логістики та розподілу виробничих ресурсів. Аналіз існуючих рішень на ринку показав їхні недоліки, такі як, відсутність адаптивних алгоритмів та низька ефективність в реальному часі. Це підкреслило актуальність створення нової системи автоматизації прийняття рішень, яка б враховувала ці вимоги та забезпечила високий рівень оптимізації та адаптивності. Було сформульовано чітке завдання для розробки системи, що включає не тільки автоматизацію логістичних процесів, а й ефективне розподілення ресурсів з урахуванням змінюваних умов.

- Проектування

Проектування системи автоматизації прийняття рішень у логістиці та розподілі виробничих ресурсів складалося з ескізного, технічного та робочого проектів. Для проектування була обрана модульна методологія проектування, що дозволило створити гнучку та масштабовану архітектуру системи. У ході ескізного проекту були розроблені прототипи інтерфейсу та основні функціональні вимоги системи, зокрема для управління транспортними маршрутами, автоматизації інвентаризації та прогнозування маршруту.

- Розробки і тестування функціональних модулів системи

Розробка та тестування функціональних модулів системи автоматизації прийняття рішень у логістиці та розподілі виробничих ресурсів включали створення та тестування основних модулів для оптимізації маршрутів, автоматизації

управління виробничими ресурсами та прогнозування шляху. Було розроблено модулі користувача, які дозволяють ефективно взаємодіяти з системою для планування ресурсів і прийняття рішень у реальному часі.

Проведено тестування системи для виявлення та виправлення помилок, перевірки точності прогнозів та оптимізації маршрутів, а також для забезпечення стабільної та надійної роботи в умовах реального часу. Тестування включало перевірку ефективності алгоритмів оптимізації, таких як мурашина колонія та машинне навчання для прогнозування часу, а також забезпечення безпеки.

- Опрацювання техніко-економічної частини та підрахунок собівартості програмного забезпечення.

Опрацювання техніко-економічної частини та підрахунок собівартості системи автоматизації прийняття рішень у логістиці та розподілі виробничих ресурсів включали детальний аналіз витрат на розробку програмного забезпечення. Це включало витрати на інструменти для розробки, програмне забезпечення для машинного навчання та оптимізації, а також витрати на оплату праці розробників, аналітиків та тестувальників. Також враховувались витрати на обслуговування та додаткові витрати.

Під час виконання проекту самостійно були опрацьовані основні компоненти системи автоматизації прийняття рішень та розподілу виробничих ресурсів, включаючи розробку алгоритмів для оптимізації транспортних маршрутів, управління запасами. Важливою частиною було створення ефективної моделі для обробки великих обсягів даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кочубей Д. Управління мережевою структурою ланцюгів постачання. Зовнішня торгівля: економіка, фінанси, право. 2019. № 3. С. 19-27.
[https://doi.org/10.31617/zt.knute.2019\(104\)0](https://doi.org/10.31617/zt.knute.2019(104)0)
2. Кирлик Н.Ю. (2021). «Штучний інтелект та його використання в логістичних процесах». Actual Problems in Economics. (243/244), 59-66.
3. Машинне навчання та штучний інтелект [Електронний ресурс] - Режим доступу: <https://wezom.com.ua/ua/blog/mashinne-navchannya-ta-shtuchniy-intelekt-rozbirajemo-yih-klyuchovi-vidminnosti>
4. Машинне навчання для ланцюжка постачання [Електронний ресурс] - Режим доступу: <https://wezom.com.ua/ua/blog/mashinne-navchannya-dlya-lantsyuzhka-postachannya-uspishni-keysy>
5. ChatGPT in Ukraine. [Електронний ресурс] - Режим доступу: [ChatGPT in Ukraine - Все про штучний інтелект в Україні](#)
6. Tchito Tcharga, C., Mih, T. A., Tchagna Kouanou, A., Fozin Fonzin, T., Kuetche Fogang, P., Mezatio, B. A., & Tchiotsop, D. (2021). Biomedical image classification in a big data architecture using machine learning algorithms. Journal of Healthcare Engineering, 2021,1- 11.
7. Gerardo Muros Anguita J., Díaz Olariaga O. Air cargo transport demand forecasting using ConvLSTM2D, an artificial neural network architecture approach. Case studies on transport policy. 2023. P. 101009. URL: <https://doi.org/10.1016/j.cstp.2023.101009> (date of access: 08.09.2024).
8. Kelleher J.D.. Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies / Kelleher J.D., Namee B.M, D'Arcy A. – The MIT Press, 2015. – 624 p.
9. Харченко В. О. Основи машинного навчання : навч. посіб. /В. О. Харченко. – Суми : Сумський державний університет, 2023. – 264 с.
10. Т.М. Басюк, В.В. Литвин, Л.М. Захарія, Н.Е. Кунанець. Машинне навчання: Навчальний посібник призначений для студентів, що навчаються за першим (бакалаврським) рівнем вищої освіти за спеціальностями галузі знань 12

- „Інформаційні технології”. – Львів: Видавництво «Новий Світ - 2000», 2019. - 335 с.
11. Олещенко, Л. М. Машинне навчання. Комп’ютерний практикум [Електронний ресурс] : навчальний посібник для студентів, які навчаються за спеціальністю 121 «Інженерія програмного забезпечення», освітня програма «Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем» / Л. М. Олещенко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 2,54 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2022. – 92 с.
 12. Columbus L. 10 Ways Machine Learning Is Revolutionizing Supply Chain Management // Forbes Available: <https://www.forbes.com/sites/louiscolumbus/2018/06/11/10-ways-machine-learning-isrevolutionizing-supply-chain-management/#45fc01a33e37>
 13. Knoll D., Prügulmeier M., Reinhart G. Predicting Future Inbound Logistics Processes Using Machine Learning. *Procedia CIRP* Vol. 52, 2016, P. 145-150
 14. Shapenko, Y., Kotova, S., Halona, I., Bilonoh, O., & Yaroshevskyy, V. (2023). Methods of forecasting freight transportation in logistics. *International Science Journal of Engineering & Agriculture*, 2(4), 79–86. <https://doi.org/10.46299/j.isjea.20230204.09>
 15. Гращенко І. С., Краснюк М. Т., Краснюк С. О. Гібридно-сценарне застосування інтелектуальних, орієнтованих на знання технологій, як важливий антикризовий інструмент логістичних компаній в Україні. Вчені записки ТНУ імені В. І. Вернадського. Сер. Економіка і управління. 2019. №1, т. 30 (69). С. 121–129.
 16. Korostin, O. (2024). Optimization of maritime transportation routes using artificial intelligence: analysis of opportunities and challenges. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (56), 31-38. <https://doi.org/10.36910/6775-2524-0560-2024-56-03>
 17. Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81- 106.
 18. Quinlan, J. Ross. C4.5: Programs for Machine learning. Morgan Kaufmann Publishers 1993.

19. Mosavi A., Varkonyi-Koczy A. R.: Integration of Machine Learning and Optimization for Robot Learning. *Advances in Intelligent Systems and Computing* 519, 349-355 (2017).
20. Bengio, Y.: Learning deep architectures for AI. *Foundations and trends in Machine Learning* 2, 1-127 (2009).
21. Яковлєв А. В. Інтелектуальна система підтримки прийняття рішення для організації гуманітарної логістики : пояснювальна записка до кваліфікаційної роботи здобувача вищої освіти на другому (магістерському) рівні, спеціальність 123 Комп'ютерна інженерія / А. В. Яковлєв ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків, 2024. – 105 с.
22. Grebennik I., Khriapkin O., Ovezgeldyyev A., Pisklakova V., Urniaieva I. (2019) The Concept of a Regional Information-Analytical System for Emergency Situations. In: Murayama Y., Velev D., Zlateva P. (eds) *Information Technology in Disaster Risk Reduction. ITDRR 2017*.
23. *IFIP Advances in Information and Communication Technology*, vol 516. Springer, Cham Scopus
24. Данилов О.І., Корягіна І.В. *"Основи логістики"*. Київ: Центр навчальної літератури, 2018.
25. Grebennik I., Reshetnik V., Ovezgeldyyev A., Ivanov V., Urniaieva I. (2019) Strategy of Effective Decision-Making in Planning and Elimination of Consequences of Emergency Situations In: Murayama Y., Velev D., Zlateva P. (eds) *Information Technology in Disaster Risk Reduction. ITDRR 2018. IFIP Advances in Information and Communication Technology*. Springer, Cham Scopus
26. Роботи на складах [Електронний ресурс] - Режим доступу: <https://wareteka.com.ua/blog/roboty-na-skladah-primery-avtomatizacii/>
27. Моделі й методи прийняття рішень: навч. посіб. / С.А. Ус, Л.С. Коряшкіна; М-во освіти і науки України, Нац. гірн. ун-т. – Д. : НГУ, 2014. – 300 с
28. Top 5 Uses of AI in Logistics- A Comprehensive Overview. [Електронний ресурс]. Режим доступу: <https://dfreight.org/blog/top-5-uses-of-ai-in-logistics/>

- 29.Смирнов І., Косарева Т. Транспортна логістика. Навчальний посібник. Центр навчальної літератури. 2019. 224 с.
- 30.Островська Г. Й., Островський О.Т. Застосування інтелектуальних інформаційних систем в контексті управління промисловими підприємствами. Маркетинг і цифрові технології. 2023. Том 7, № 1. С. 69-81. DOI: 10.15276/mdt.7.1.2023.5.
- 31.Островська Г. Й., Островський О.Т. Штучний інтелект в умовах сучасних підприємств та маркетингових кампаній: ефективні інструменти та перспективи розвитку. Маркетинг і цифрові технології. 2023. Том 7, № 3. С. 66-82. Doi: 10.15276/mdt.7.3.2023.5
- 32.Матвійчук А.В. Штучний інтелект в економіці: нейронні мережі, нечітка логіка : монографія . – К. : КНЕУ, 2011. – 439 с
- 33.Герман В. А. Аналіз інформаційних технологій в транспортних системах "Розумних міст" : кваліфікаційна робота освітнього рівня „Бакалавр“ „126 — інформаційні системи та технології“ / В. А. Герман. — Тернопіль : ТНТУ, 2023. — 45 с.
- 34.Hisham, A., Harin, S.: Deep Learning – the new kid in Artificial Intelligence. (2017)
- 35.Глибоке навчання [Електронний ресурс] - Режим доступу: [https://uk.wikipedia.org/wiki/%D0%93%D0%BB%D0%B8%D0%B1%D0%BE%D0%BA%D0%B5_%D0%BD%D0%B0%D0%B2%D1%87%D0%B0%D0%BD%D0%BD%D1%8F#:~:text=%D0%AF%D0%BA%D1%89%D0%BE%20%D0%BF%D0%BE%D0%B3%D0%BB%D1%8F%D0%BD%D1%83%D1%82%D0%B8%20%D0%BD%D0%B0%20%D0%B3%D0%BB%D0%B8%D0%B1%D0%BE%D0%BA%D0%B5%20%D0%BD%D0%B0%D0%B2%D1%87%D0%B0%D0%BD%D0%BD%D1%8F,%D1%82%D0%B0%20%C2%AB%D0%BD%D0%B0%D0%B9%D0%B3%D0%BB%D0%B8%D0%B1%D1%88%D0%B5%C2%BB%20\(%D0%B0%D0%BD%D0%B3%D0%BB](https://uk.wikipedia.org/wiki/%D0%93%D0%BB%D0%B8%D0%B1%D0%BE%D0%BA%D0%B5_%D0%BD%D0%B0%D0%B2%D1%87%D0%B0%D0%BD%D0%BD%D1%8F#:~:text=%D0%AF%D0%BA%D1%89%D0%BE%20%D0%BF%D0%BE%D0%B3%D0%BB%D1%8F%D0%BD%D1%83%D1%82%D0%B8%20%D0%BD%D0%B0%20%D0%B3%D0%BB%D0%B8%D0%B1%D0%BE%D0%BA%D0%B5%20%D0%BD%D0%B0%D0%B2%D1%87%D0%B0%D0%BD%D0%BD%D1%8F,%D1%82%D0%B0%20%C2%AB%D0%BD%D0%B0%D0%B9%D0%B3%D0%BB%D0%B8%D0%B1%D1%88%D0%B5%C2%BB%20(%D0%B0%D0%BD%D0%B3%D0%BB)
- 36.Deng and D. Yu. Deep Learning: Methods and Applications. Foundations and Trends in Signal Processing, 2013, vol. 7, nos. 3–4, pp. 197–387.

- 37.Турило А.М., Захарченко О.О. *"Логістичний менеджмент"*. Київ: КНЕУ, 2019.
- 38.Richards G. "Warehouse Management: A Complete Guide to Improving Efficiency and Minimizing Costs in the Modern Warehouse". Kogan Page, 2017.
- 39.Tompkins J.A., White J.A., Bozer Y.A., Tanchoco J.M.A. "Facilities Planning". Wiley, 2018.
- 40.Shestakovych, M. O., & Shabatura, Y. V. (2024). Застосування нейронних мереж для оптимізації процесу переходу інформаційних систем з монолітною архітектурою в мікросервісну. *Forestry Education and Science: Current Challenges and Development Prospects. International Science-Practical Conference, October 23-25, 2024, Lviv, Ukraine.*
<https://doi.org/10.36930/conf150.5.14>
- 41.Іванищева А.В. Сучасні тенденції розвитку та інновації в логістиці [Електронний ресурс]:
<http://dspace.onu.edu.ua:8080/bitstream/handle/123456789/10594/344-362.pdf;jsessionid=59A8E8103F0B42E95F350C69F0C8D8CF?sequence=1>
- 42.Новіцький П.С. Еволюція логістичного бізнесу через призму використання інформаційних технологій. Здобутки та досягнення прикладних та фундаментальних наук ХХІ століття. Черкаси: МЦНД, 2022. URL:
<https://orcid.org/0000-0002-7300-5262>
- 43.Поляк Е. Вплив ІТ-технологій на процеси оптимізації логістики в Україні. Євроінтеграція України та економічна безпека держави: збірник наукових праць. Вип. 1 (28), 2022. DOI: 10.24144/2078-1431.2022.1(28).150-156
- 44.ПЕЛЕХ, К., & ДЕМИДЧУК, Л. (2024). Інтернет речей в технологіях сучасних вантажних перевезень (транспортній логістиці). *Herald of Khmelnytskyi National University. Economic Sciences*, 334(5), 106-111.

ДОДАТОК А. Опис програми

A.1 Загальні відомості

Позначення та найменування програми: Систему штучного інтелекту для прийняття рішень у логістиці та розподілі виробничих ресурсів. Найменування програми «SmartChain».

Програмне забезпечення, необхідне для функціонування програми: для функціонування програми необхідно мати планшет, комп'ютер або телефон.

Мови програмування, на яких написана програма: програма написана у середовищі програмування PyCharm на мові Python.

A.2 Функціональне призначення

Система призначена для користувачів підприємств, що працюють у сфері логістики та виробництва. Вона дозволяє автоматично оптимізувати маршрути транспорту, управляти запасами та ефективно розподіляти ресурси.

A.3 Структура програми з описом функцій складових частин і зв'язки між ними

1. Аналіз запиту на оптимізацію:

Вхідні дані: пункт відправлення, пункт призначення, дата, вимоги до часу доставки та інші критерії.

Опис: Користувач або система вводить дані про вимоги до маршруту (наприклад, транспорт, час доставки тощо). Цей етап відповідає за первинне опрацювання запиту.

Зв'язки: Вхідні дані передаються в систему оптимізації для подальшого розрахунку.

2. Оптимізація маршрутів (за допомогою мурашиної колонії або іншого алгоритму):

Вхідні дані: координати точок (склади, магазини), кількість транспортних засобів.

Опис: Алгоритм оптимізації (наприклад, алгоритм мурашиної колонії) генерує найкращі маршрути для доставок, враховуючи задані умови.

Зв'язки: Після виконання оптимізації результат (найкращий маршрут)

передається в систему для подальшої обробки.

3. Прогнозування часу доставки (на основі машинного навчання):

Вхідні дані: оптимізований маршрут, погодні умови, рівень трафіку, день тижня.

Опис: Модель машинного навчання прогнозує час доставки на основі історичних даних про маршрути та вплив різних факторів (погода, трафік).

Вихідні дані: результат оптимізації маршрутів, прогнозований час доставки, витрати на ресурси.

ДОДАТОК Б. Керівництво користувача

Для роботи розробленої Системи «SmartChain» повинна бути файл програми, який буде завантажений у хмарне середовище у якому доступ можна отримати тільки за допомогою коду. Користувачем системи буде компанія, яка придбала програму у постачальника і розробника.

Запуск програми здійснюється за допомогою файлу, який знаходиться у хмарному середовищі.

Далі користувач у самій програмі сам може ввести дані, які потрібні для розрахунку оптимізації маршруту. Приклад на рис. Б 1.1

```
119 # Основний блок
120 > if __name__ == "__main__":
121     # Приклад даних
122     locations = ['Склад', 'Магазин №1', 'Магазин №2', 'Магазин №3', 'Магазин №4']
123     coordinates = [
124         0: [0, 0], # Склад
125         1: [10, 10], # Магазин №1
126         2: [20, 20], # Магазин №2
127         3: [30, 30], # Магазин №3
128         4: [40, 40] # Магазин №4
129     ]
130
131     # Навчання моделі машинного навчання
132     ml_model = train_ml_model()
133
134     # Отримуємо оптимальний маршрут за допомогою мурашиної колонії
135     best_route, best_distance = route_optimization(locations, coordinates)
136
137     # Прогнозуємо час на основі знайденого маршруту
138     predicted_time = predict_route_time(ml_model, best_route, coordinates, day_of_week=6, traffic_level=1, weather=2)
```

Рис. Б 1.1 Дані

```

import numpy as np
import networkx as nx
import random
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

# 1. Алгоритм мурашиної колонії для пошуку оптимального алгоритму
def route_optimization(locations, coordinates, num_vehicles=1, iterations=100,
alpha=1.0, beta=2.0,
                        evaporation_rate=0.5):
    # Створюємо граф для моделювання точок доставки
    G = nx.complete_graph(len(locations))
    for (u, v) in G.edges():
        # Відстань між точками
        G.edges[u, v]['distance'] = np.linalg.norm(
            np.array(coordinates[u]) - np.array(coordinates[v])) # Евклидова відстань

    # Ініціалізація феромонів
    pheromones = {edge: 1 for edge in G.edges}

    best_route = None
    best_distance = float('inf')

    for _ in range(iterations):
        all_routes = []
        for vehicle in range(num_vehicles):
            route = [0] # Початкова точка
            unvisited = set(range(1, len(locations)))

```



```
while unvisited:
```

```
    current = route[-1]
```

```
    probabilities = []
```

```
    for neighbor in unvisited:
```

```
        pheromone = pheromones[(min(current, neighbor), max(current, neighbor))]
```

```
        distance = G.edges[current, neighbor]['distance']
```

```
        probability = (pheromone ** alpha) * ((1 / distance) ** beta)
```

```
        probabilities.append((neighbor, probability))
```

```
    # Нормалізація ймовірностей
```

```
    total = sum(p for _, p in probabilities)
```

```
    probabilities = [(node, p / total) for node, p in probabilities]
```

```
    # Вибір наступної точки за ймовірностями
```

```
    next_node = random.choices([node for node, _ in probabilities],
```

```
                               weights=[p for _, p in probabilities])[0]
```

```
    route.append(next_node)
```

```
    unvisited.remove(next_node)
```

```
    # Оновлення феромонів на пройденому маршруті
```

```
    pheromones[(min(current, next_node), max(current, next_node))] *= (1 -
evaporation_rate)
```

```
    # Замкнути маршрут, повернувшись до початкової точки
```

```
    route.append(route[0])
```

```
    all_routes.append(route)
```

```
# Оцінка маршрутів та оновлення феромонів
```

```

for route in all_routes:
    route_distance = sum(G.edges[route[i], route[i + 1]]['distance'] for i in
range(len(route) - 1))
    if route_distance < best_distance:
        best_distance = route_distance
        best_route = route

# Посилення феромонів на найкращих маршрутах
for i in range(len(route) - 1):
    edge = (min(route[i], route[i + 1]), max(route[i], route[i + 1]))
    pheromones[edge] += 1 / route_distance

return best_route, best_distance

# 2. Модель машинного навчання (лінійна регресія) для передбачення часу
def train_ml_model():
    # Генерація даних з більш реалістичним розподілом часу в дорозі
    distances = np.random.uniform(5, 50, size=100) # Випадкові відстані (км) між 5 і
50
    traffic_levels = np.random.randint(1, 5, size=100) # Рівень трафіку від 1 до 4
    weather_conditions = np.random.randint(1, 3, size=100) # Погодні умови (1 -
сприятливі, 2 - несприятливі)
    days_of_week = np.random.randint(0, 6, size=100) # Дні тижня від 0 до 6
    num_stops = np.random.randint(3, 10, size=100) # Кількість зупинок

    # Цільова змінна — реалістичний час у дорозі з урахуванням різних факторів
    y = distances * 2 + traffic_levels * 10 + weather_conditions * 15 +
np.random.normal(0, 10, size=100)

    # Об'єднання всіх ознак у матрицю

```

```
X = np.column_stack([distances, days_of_week, traffic_levels, weather_conditions,
num_stops])
```

```
# Поділ даних на тренувальну та тестову вибірку
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

```
# Створення та навчання моделі лінійної регресії
model = LinearRegression()
model.fit(X_train, y_train)
```

```
# Оцінка точності моделі
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')
```

```
return model
```

```
# 3. Використання моделі машинного навчання для передбачення часу маршруту
def predict_route_time(model, route, coordinates, day_of_week, traffic_level, weather):
```

```
    # Перетворимо маршрут на ознаки
    distances = [np.linalg.norm(np.array(coordinates[route[i]]) -
np.array(coordinates[route[i + 1]])) for i in
        range(len(route) - 1)]
    total_distance = sum(distances)
```

```
# Ознаки: відстань, день тижня, трафік, погода, кількість зупинок
features = np.array([total_distance, day_of_week, traffic_level, weather, len(route)])
features = features.reshape(1, -1)
```

```

# Передбачення часу маршруту з використанням моделі
predicted_time = model.predict(features)
predicted_time = max(predicted_time[0], 0) # Забезпечення невід'ємного значення
return predicted_time

# Основний блок
if __name__ == "__main__":
    # Приклад даних
    locations = ['Склад', 'Магазин №1', 'Магазин №2', 'Магазин №3', 'Магазин №4']
    coordinates = {
        0: [0, 0], # Склад
        1: [10, 10], # Магазин №1
        2: [20, 20], # Магазин №2
        3: [30, 30], # Магазин №3
        4: [40, 40] # Магазин №4
    }

    # Навчання моделі машинного навчання
    ml_model = train_ml_model()

    # Отримуємо оптимальний маршрут за допомогою мурашиної колонії
    best_route, best_distance = route_optimization(locations, coordinates)

    # Прогнозуємо час на основі знайденого маршруту
    predicted_time = predict_route_time(ml_model, best_route, coordinates,
day_of_week=6, traffic_level=1, weather=2)
    print(f'Predicted time for the best route: {predicted_time} minutes')
    print("Best route found:", best_route)
    print("Distance of best route (km):", best_distance)

```