

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «магістр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

**«Розробка бібліотеки C# для оптимізації процесів управління товарними
потокami в облікових системах торгових підприємств»**

Виконав студент групи ЗППЗм-23-1
РОМАНЧУК Роман Васильович

Керівник роботи:
ЄФРЕМОВ Юрій Миколайович

Рецензент:
ВЛАСЕНКО Олег Васильович

Житомир – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

зав. кафедри інженерії програмного
забезпечення

Тетяна ВАКАЛЮК

«01» жовтня 2024 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **РОМАНЧУК Роман Васильович**

Керівник роботи: **ЄФРЕМОВ Юрій Миколайович**

Тема роботи: **«Розробка бібліотеки C# для оптимізації процесів управління
товарними потоками в облікових системах торгових підприємств»,**

затверджена Наказом закладу вищої освіти від **«01» жовтня 2024 р. № 521/с**

Вихідні дані для роботи:

Предметом дослідження є аналіз алгоритмів, методів та бібліотек машинного навчання для оптимізації логістичних процесів в торгових підприємствах в умовах ринку.

Консультанти з магістерської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Єфремов Ю.М.	02.10.2024р.	02.10.2024р.
2	Єфремов Ю.М	02.10.2024р.	02.10.2024р.
3	Єфремов Ю.М	02.10.2024р.	02.10.2024р.

РЕФЕРАТ

Кваліфікаційна магістерська робота полягає в дослідженні ефективності впровадження автоматизованих систем машинного навчання в торгових підприємствах для покращення ефективності роботи підприємств зокрема логістичних та складських систем які на сьогодні склалися в Україні.

У дослідженні представлено оцінка актуальних досліджень та розробки нових рішень для оптимізації товарообороту, зменшення витрат на доставку та ефективного управління залишками товарів на складах.

Магістерська робота складається із вступу, трьох розділів та висновків, списку використаних джерел, що налічує 66 сторінок, 14 ілюстрацій, 8 таблиць, та 15 посилань на зовнішні джерела.

КЛЮЧОВІ СЛОВА: МАШИННЕ НАВЧАННЯ, ПРОГНОЗУВАННЯ ПОПИТУ, ОПТИМІЗАЦІЯ ЛОГІСТИКИ, ERP-СИСТЕМИ, AutoML, РОЗПІЗНАВАННЯ, НЕЙРОННА МЕРЕЖА, МАШИННЕ НАВЧАННЯ, АВТОМАТИЗАЦІЯ.

Abstract

The qualification master's thesis consists of a study dedicated to the effectiveness of implementing automated machine learning systems in trading enterprises to improve the efficiency of enterprises, in particular logistics and warehouse systems that have developed in Ukraine today.

The study presents an assessment of current research and development of new solutions for optimizing product turnover, reducing delivery costs and effectively managing product stock in warehouses.

The master's thesis consists of an introduction, three chapters and conclusions, a list of sources used, which includes 66 pages, 14 illustrations, 8 tables, and 15 links to external sources.

KEYWORDS: MACHINE LEARNING, DEMAND FORECASTING, LOGISTICS OPTIMIZATION, ERP SYSTEMS, AutoML, RECOGNITION, NEURAL NETWORK, MACHINE LEARNING, AUTOMATION

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ	9
1.1. Технології штучного інтелекту в оптимізації логістичних процесів ...	9
1.1.1. Огляд алгоритмів для прогнозування попиту та оптимізації логістики	9
1.1.2. Аналіз існуючих бібліотек машинного навчання.....	21
1.2. Аналіз сучасних методів управління товарними потоками, використання ERP-системи для управління бізнес-процесами	25
Висновки по розділу 1	31
РОЗДІЛ 2. РОЗРОБКА ПРОГРАМНОЇ БІБЛІОТЕКИ	32
2.1. Інтеграція бібліотек штучного інтелекту з обліковими системами....	32
2.2. Обґрунтування вибору методів прогнозування та програмних засобів для реалізації програмного продукту	34
2.3. Реалізація програмних модулів бібліотеки	40
2.4. Інтеграція бібліотеки з ERP-системами.....	43
Висновки до розділу 2	51
РОЗДІЛ 3. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ РОБОТИ СИСТЕМИ.....	52
3.1. Тестування та результати впровадження	52
3.2. Ефективність застосування бібліотеки на підприємствах	54
3.3. Переваги та обмеження розробленого рішення.....	56
Висновки до розділу 3	59
ВИСНОВКИ.....	60
СПИСОК ЛІТЕРАТУРИ.....	62
ДОДАТКИ.....	64
Додаток А.....	65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ERP – (Enterprise Resource Planning) – система планування ресурсів підприємства.

LSTM – (Long Short-Term Memory) – довготривала короткострокова пам'ять, тип рекурентної нейронної мережі.

XGBoost – (Extreme Gradient Boosting) – бібліотека для реалізації градієнтного бустингу.

RNN – (Recurrent Neural Network) – рекурентна нейронна мережа.

ML.NET – (Machine Learning .NET) – кросплатформна бібліотека для машинного навчання від Microsoft.

ОСП – (Облікова Система Підприємства) – ERP-система, чи будь яка інша комп'ютерна система, в якій здійснюється облік та планування.

BAS – (Business Automation Software) – платформа для створення комплексних систем автоматизації бізнесу.

ІІІ – штучний інтелект.

ВСТУП

Актуальність теми. Сьогодні торговим підприємствам для забезпечення прибутковості діяльності та можливості втриматися на ринку доводиться працювати в умовах високої конкуренції та швидкої зміни кон'юнктури ринку.

Особливо ускладнилася система логістики в умовах воєнного стану в Україні, це стосується як ланцюгів доставки так і швидкості умов транспортування товарів від постачальників. Тому однією з ключових аспектів у сфері торгівлі, що вимагає особливої уваги, є управління товарними потоками, що включає закупівлі та продажі та швидке реагування на зміни факторів в системі логістики.

На даний час в багатьох підприємствах в Україні впроваджено ERP системи та спеціалізовані обліковані програми для управління товарними потоками в торгівлі (облікова система підприємства – ОСП) від різних розробників програмного забезпечення (BAS, SAP, IT-Enterprise та ін.). Всього на ринку IT послуг, які поставляють такі систем діє більше двох десятків постачальників до яких входять як Українські компанії так і закордонні. Ці системи як правило максимально конфігуровано під реальну структуру бізнес процесів на кожному конкретному підприємстві

Як правило такі системи для розрахунку та оптимізації товарних потоків як правило використовують стандартні, часто обмежені алгоритми які не дозволяють враховувати значну кількість факторів, або взагалі не мають такого програмного інструментарію.

Впровадження систем штучного інтелекту (ШІ) у ці процеси дозволяє значно підвищити швидкість та точність прогнозування потреб у товарах, оптимізувати запаси та зменшити витрати.

Оптимальним рішенням що дозволяє мінімально впливати на структуру діючих конфігурацій облікових систем є підключення зовнішнього програмного модуля. Розробка бібліотеки для підключення до облікової системи підприємства з метою використання технологій ШІ є актуальним завданням, яке сприятиме підвищенню ефективності управління товарними потоками.

Метою даного дослідження є даної роботи є дослідження застосування принципів AutoML для виконання функцій прогнозування та оптимізації товарного потоку в підсистемах закупівель та продажів в умовах ринку України.

Для досягнення цієї мети були поставлені наступні **завдання**:

1. Аналіз існуючих методів прогнозування та оптимізації товарних потоків.
2. Вибір та обґрунтування найбільш ефективних алгоритмів ШІ для застосування у даній сфері.
3. Розробка архітектури бібліотеки та її інтеграція з ОСП .
4. Реалізація обраних алгоритмів у вигляді програмних модулів.
5. Тестування БШІ на реальних даних підприємства та оцінка її ефективності.

Об'єктом дослідження є автоматизація процесу управління товарними потоками в підсистемах закупівель та продажів підприємства із застосування принципів AutoML в умовах ринку України.

Предметом дослідження є методи та алгоритми штучного інтелекту, що застосовуються для прогнозування та оптимізації процесів товарообороту торгового підприємства в ринкових умовах України, а також програмні засоби, що реалізують ці методи у вигляді бібліотеки на мові програмування C#.

Новизна дослідження є створення бібліотеки для оптимізації процесів управління товарними потоками враховуючи специфіку торгових підприємств.

Практичне значення полягає у використанні бібліотеки C# для оптимізації товарообороту у підприємстві оптової торгівлі.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

1.1. Технології штучного інтелекту в оптимізації логістичних процесів

1.1.1. Огляд алгоритмів для прогнозування попиту та оптимізації логістики

Сьогодні підприємствам України доводиться працювати в дуже динамічному ринковому середовищі, де на споживчий попит впливає широкий спектр факторів, включаючи глобальні економічні процеси, технологічну креативність і непередбачені ситуації, такі як пандемії та геополітичні конфлікти, що значно ускладнює роботу бізнесу, та відповідно процесів планування та прогнозування попиту особливо в частині експортно-імпорту товарів.

Не маючи системи ефективного прогнозування попиту та логістичних факторів компаніям досить зладжено працювати в сучасних реаліях, тому прогнозування та оптимізація логістики є ключовими елементами управління ресурсами, дозволяє ефективно керувати запасами, знижувати витрати та покращувати рівень обслуговування клієнтів.

Раніше, до появи технологій штучного інтелекту, для прогнозування використовувалися статистичні методи та експертні оцінки, зокрема: метод експоненціального згладжування - враховує тренди та сезонність у виборці даних; метод найменших квадратів - відбувається в побудові лінійних регресійних моделей для оцінки залежності між змінними; каузальні моделі – будуються з метою відслідкування причинно-наслідкових зв'язків між факторами, що впливають на попит; експертні оцінки – базуються на думках та знаннях експертів у галузі.

Проте, ці методи хоч і дозволяли отримувати певні результати - все ж мали ряд обмежень: були трудомісткими, вимагали значних знань статистики та не завжди могли врахувати велику кількість факторів, що впливають на попит.

Розвиток обчислювальної техніки та збільшення обсягів даних у зв'язку з розширенням масштабів бізнесу, стало актуальним використовувати більш складні та точні методи прогнозування.

Враховуючи що двома нерозривно взаємопов'язаними і в якійсь мірі агнтагоністичними процесами є продажі (попит на товари) з одного боку та

закупівлі (логістика та складське зберігання) з іншого боку, далі подання опис алгоритмів надається за наступними сегментами.

Для передбачення попиту на товари або послуги на основі історичних даних, трендів, сезонності та інших факторів використовуються алгоритми прогнозування, групи: методів часових рядів, регресійних методів.

До методів часових рядів відносяться: метод ковзної середньої (Moving Average), експоненційне згладжування (Exponential Smoothing), ARIMA (Autoregressive Integrated Moving Average) [9].

Ковзна середня - метод розрахунку базується на загальному середньому значенні за певний період часу, добре працює для згладжування випадкових коливань, хоча недостатньо точно регулює тенденцію та сезонність.

Експоненційне згладжування – показник розраховується з враховуванням ваги попередніх значень, причому більш свіжим даним надається більша вага. Цей метод має різні варіації, які враховують тренд та сезонність (Holt-Winters).

ARIMA (Autoregressive Integrated Moving Average) – це метод побудований на автокореляції та інтеграції даних. Особливо добре підходить для прогнозування часових рядів з трендовою сезонністю.

До регресійних методів відносяться: лінійна регресія (Linear Regression), множинна регресія (Multiple Regression).

Лінійна регресія (Linear Regression). Метод відслідковує лінійну залежність між змінною (попит), якає залежною, та незалежними змінними (ціна, реклама, сезонність).

Множинна регресія (Multiple Regression) – для прогнозування попиту використовує кілька незалежних змінних.

Для оптимізації процесів логістики, зокрема: маршрутизації, управління запасами та складських операцій використовується група алгоритмів оптимізації логістики.

До складу цієї групи відносять різні типи транспортних задач, з них наприклад: задача комівояжера, задача маршрутизації, жадібні алгоритми, генетичні алгоритми, алгоритми мурашиної колонії, алгоритми локального

пошуку, а саме:

- задача комівояжера полягає в знаходженні найкоротшого маршруту, який проходить через всі задані пункти по одному разу та повертається у вихідний пункт;
- задача маршрутизації транспорту вирішується узагальненням TSP для кількох транспортних засобів з різними обмеженнями (вантажопідйомність, час доставки);
- жадібні алгоритми - прості алгоритми, які на кожному кроці обирають локально оптимальне рішення;
- генетичні алгоритми - еволюційні алгоритми, які імітують природний відбір для знаходження оптимального рішення.

Досить оригінальні але досить ефективними алгоритмами є алгоритми мурашиної колонії, який імітують поведінку мурашок для знаходження найкоротших шляхів.

Слід зазначити також алгоритми локального пошуку – які починають обробку деякого початкового рішення та поступово покращують його, переходячи до сусідніх рішень.

Для точного визначення оптимального рівня запасів та мінімізації витрат на зберігання використовуються алгоритми оптимізації управління запасами що дозволяє ефективно керувати товарооборотом підприємства. Використання цих алгоритмів дає ефективніше планувати закупівлі, покращити обслуговування клієнтів та збільшити прибутковість.

До них відносяться: модель економічного розміру замовлення, модель з точкою повторного.

Модель економічного розміру замовлення по своїх суті являє метод визначання оптимального розміру замовлення, який мінімізує загальні витрати на зберігання та замовлення запасів.

Цікавою також є модель точки повторного замовлення, в рамках якої визначають оптимальний момент, з якого необхідно зробити нове замовлення, щоб уникнути дефіциту.

Для автоматизації та вдосконалення процесів на складі використовується група алгоритмів оптимізації складських операцій. За допомогою цих інструментів виконується аналіз великих обсягів даних про замовлення, наявність товарів, розташування складських зон та інші параметри, а також для визначення оптимального шляху руху товарів, розміщення продукції на стелажах та розподіл завдань між працівниками. Завдяки цьому вдається значно скоротити час виконання замовлень, знизити витрати на зберігання та підвищити точність виконання операцій.

До цієї групи відносяться: алгоритми розміщення товарів, алгоритми сортування.

Алгоритми розміщення товарів - оптимізують розміщення товарів на складі для мінімізації часу на збирання замовлень.

Алгоритми сортування - ефективно сортують товари для швидкої обробки.

Вказані алгоритми використовуються торговими підприємствами для вирішення ряду практичних завдань.

Проте справжнім революційним проривом в частині точності прогнозування попиту та вирішення задач логістики стала поява технологій штучного інтелекту (ШІ) та методів автоматизованого машинного навчання (AutoML).

Системне використання механізмів AutoML дозволяє підприємству ефективно передбачати тенденції та своєчасно коригувати свою діяльність відповідно до змін ринку, своєчасно виявляти закономірності в даних, будувати складні моделі та адаптуватися до змін умов в автоматичному режимі.

Використання AutoML дозволяє отримувати точні прогнози майбутнього попиту. Алгоритми машинного навчання здатні виявляти складні нелінійні залежності в даних, що дозволяє отримувати більш точні прогнози. При цьому усі процеси побудови моделей та отримання прогнозів виконуються автоматично, що звільняє аналітиків від повсякденної одноманітної роботи.

Найбільш розповсюдженими алгоритмами для прогнозування попиту та оптимізації логістичних процесів на сьогодні є: лінійна регресія, дерева рішень, Random Forest, Boosting, різні типи нейронних мереж.

Лінійна регресія – являє собою алгоритм, що дає можливість встановити зв'язок між залежною змінною (попит) та одним або кількома незалежними змінними (час, ціна, сезонність) яку можна представити лінійним рівнянням.

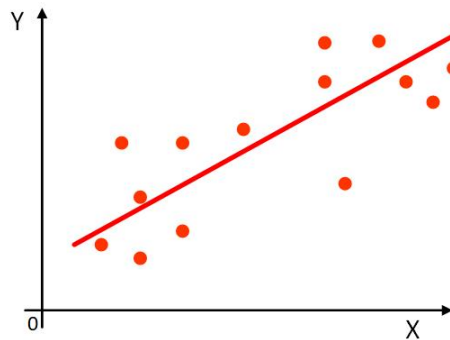


Рис. 1.1. Приклад графіку лінійної регресії

На Рис. 1.1 вісь X - незалежна змінна, вісь Y = залежна змінна (вихід), лінія регресії - оптимальна лінія для моделі.

Завдання алгоритму - це пошук лінії, яка найкраще підходить для з'єднання точок сукупності, як показано на прикладі графіка вище (Рис. 1.1.1).

Багатовимірна лінійна регресія визначається таким на рівнянням яке описується наступним чином:

$$y = b_0 + b_1x_1 + b_2x_2 + \dots b_nx_n$$

де, $x_1, x_2 \dots x_n$ – це незалежні вхідні змінні. Коефіцієнти $b_1, b_2 \dots b_n$ описують кореляцію змінних. b_0 - це значення залежної змінної, якщо це незалежні змінні дорівнюють 0.

Проста лінійна регресія має лише одну вхідну незалежну змінну, отже її рівняння спрощується:

$$y = b_0 + bx \text{ або } y = a + bx$$

Застосування функції втрат або середньоквадратичної помилки (MSE) дозволяє покращити точність моделі, в частині оптимізації коефіцієнтів, що можна зробити з:

$$MSE = \sum_{i=1}^N \text{відхилення}_i^2 = \sum_{i=1}^N (y_i - f(x_i))^2 = \sum_{i=1}^N (y_i - a - b \cdot x_i)^2$$

де:

y_i - фактичне значення залежної змінної,

N - загальна кількість експериментальних точок.

Дерева рішень – це різновид моделей у вигляді дерева, у якій кожен вузол представляє атрибут, а гілки – можливі значення цього атрибута.

Зазначений алгоритм добре працює з нелінійними залежностями і може обробляти як числові, так і дані які є категоріальними.

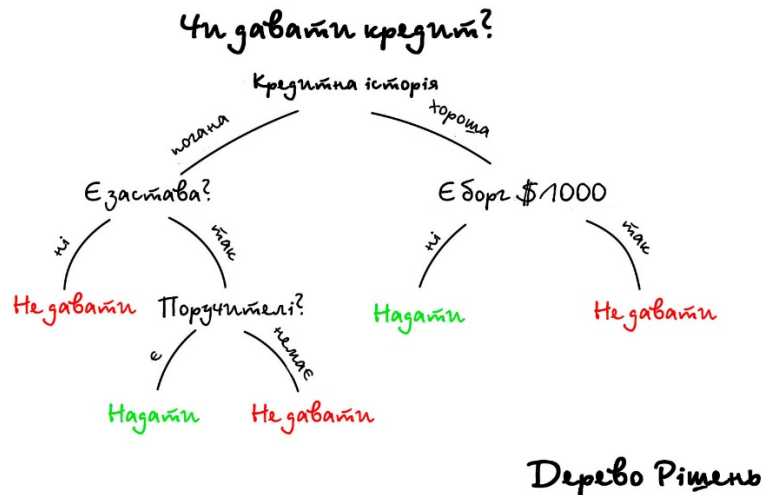


Рис. 1.2. Приклад схеми дерева рішень

На рис 1.2 представлена приклад схеми дерева рішень яке описує видачу кредиту з огляду на кредитну історію. Модель «Дерево рішень» призначена для вирішення завдань ідентифікації групи, до якої належить предмет на основі заданих характеристик.

Кожен вузол такого дерева призначений для перевірки кожної окремої характеристики, при цьому від цього вузла відходить по одній гілці, кожна з яких веде до наступного вузла або до кінцевого результату – категорії, до якої відноситься предмет. Тобто, ми рухаємося по дереву, відповідаючи на питання про характеристики об'єкта, поки не дістанемося до відповідного “листа” – категорії.

Найбільш відомих алгоритми побудови дерев рішень – це CART (Classification and Regression Tree).

Алгоритм CART можна легко впоратися з проблемами регресії та класифікації. Він формує двійкові дерева, де кожен вузол може мати лише дві гілки-нащадки. Це вивчає інтерпретацію моделі, але іноді може призвести до меншої глибини дерева, ніж інші методи.

Випадковий ліс (Random Forest) - ансамбль алгоритмів, який будує множину дерев рішень і об'єднує їхні прогнози для підвищення точності .

В такій моделі рішення приймаються на кожному «вузлу» дерева. При цьому визначення розподілу об'єктів на класи та регресії по правилах які розроблені для кожного такого дерева.

Такі дерева будують на згенерованих випадкових підмножинах даних для навчання та випадково вибраних характеристиках. Цей підхід дозволяє уникнути перенавчання, яке часто відбувається з окремими деревами рішень, і підвищує точність прогнозування.

Після вибору деревами варіанту класу приймається загальне рішення що клас що отримав найбільшу кількість голосів вважається кінцевим прогнозом. Таке взаємоузгодження дозволяє лісу досягти високої точності та стійкості до помилкових величин у даних. Random Forest ефективно опрацьовують великі набори даних, виявляють нелінійні залежності між вхідними даними та автоматично вибирають найважливіші ознаки.

Boosting – алгоритм «ансамблю» моделей, який послідовно будує моделі, і приділяє більше уваги помилковим прогнозам попередніх.

Цей підхід у машинному навчанні діє як стратегічний об'єднувач: він об'єднує кілька простих моделей, таких як дерев'яні рішення, в єдину потужну систему (див. рис. 1.3). Суть у тому, що кожна нова модель навчається на помилках помилки, і в результаті ми отримуємо ансамбль, здатний ефективно розв'язувати навіть найскладніші завдання класифікації та регресії [18].

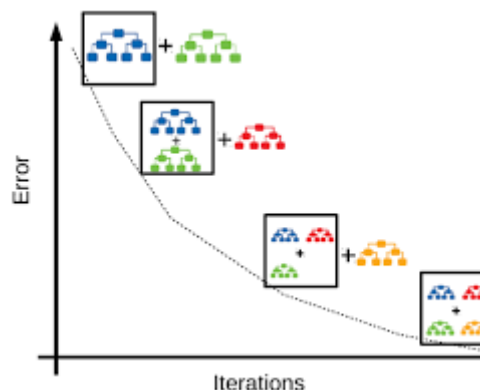


Рис. 1.3 Графічна модель алгоритму Boosting

Процес (Boosting) починається з навчанням першої (слабкої) моделі, далі формують наступну модель, для спроби виправити помилки попередньої.

Кожному об'єкту, що важливо, в навчальній вибірці присвоюється вага, якою відображається його вагу для навчання наступної моделі.

Об'єкти, які були неправильно класифіковані попередніми моделями, отримують більшу вагу, а ті, які були класифіковані правильно, – меншу (нові моделі зосереджуються на найбільш складних прикладах).

Алгоритм робить відносно точні прогнози на різних типах даних завдяки об'єднанням кілька моделей Він часто використовується для вирішення складних задач, таких як класифікація зображень, виявлення аномалій та прогнозування часових рядів.

Має такі «підвиди: AdaBoost, Gradient Boosting, XGBoost та LightGBM. Кожен з них має свої особливості, які необхідно налаштувати для досягнення оптимальних результатів.

AdaBoost – полягає в тому, що на кожному кроці алгоритм акцентується на тих об'єктах, які були класифіковані з помилками попередніми моделями. Отже, наступні моделі фіксуються на найскладніших прикладах. AdaBoost ефективний для багатьох задач класифікації.

Gradient Boosting – дозволяє використовувати різні функції втрат, та покроково навчає нову модель, для мінімізації градієнту функції втрат на попередньому ансамблі. Це дає йому значну адаптивність до різних типів даних і задач.

XGBoost – це алгоритм градієнтного бустингу, який оптимізований для роботи з великими обсягами даних на потужних серверах. XGBoost використовує ряд оптимізованих алгоритмів для розрахунку градієнтів і формування дерев рішень а також включає в себе ряд спеціальних методів регулювання, що запобігає стану перенавчання моделі.

LightGBM – відзначається, як така що оптимізована для роботи з великими даними, має ряд іновацій - ріст дерев з листя, гістограмні алгоритми, що дозволяє

йому досягти високої швидкості навчання і низького споживання ресурсів комп'ютера.

LightGBM і XGBoost – є конкурентами по функціоналу та можливостям і часто використовуються для задач, де швидкість навчання є критичним фактором.

Нейронні мережі – це «аналог біологічних нейронних мереж» і здатні виявляти складні неочевидні стохастичні залежності вхідних даних.

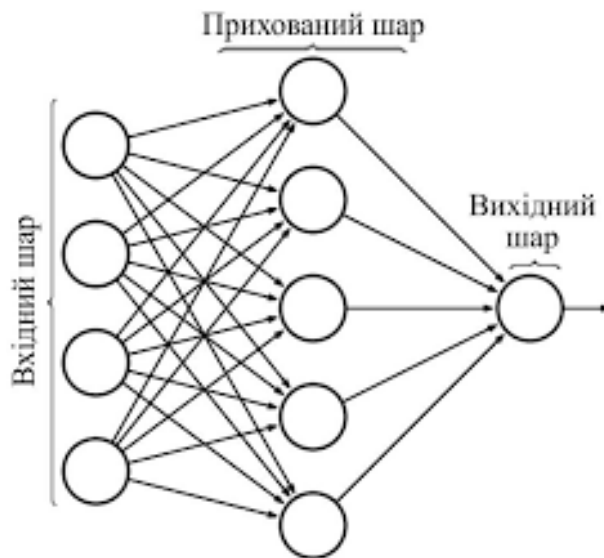


Рис. 1.4. Архітектура нейронної мережі

Це математичні моделі сформовані з вузлів, які є взаємопов'язаними і обробляють інформацію подібно до нейронів і здатні навчатися на великих обсягах даних, виявляти закономірності та робити прогнози (Рис. 1.4).

«Нейрони» отримують вхідні дані та обробляють їх з наступною передачею результату наступним вузлам моделі.

Нейронні мережі відзначаються здатністю до адаптивного навчання - самостійно калібруються на навчених даних, покращують свою точність в процесі виконання завдань.

Налічується багато варіантів нейронних мереж, кожна з них зі своїми особливостями та застосовуються для виконання різних типу завдань.

Так згорткові нейронні мережі (CNN) широко використовуються для обробки зображень.

Рекурентні нейронні мережі (RNN) – застосовують для обробки послідовностей текст або мови.

Генеративні змагальні мережі (GAN) – найбільш придатні для створення нових даних, зокрема, зображень або музики.

Спектр використання має широкий перелік галузей: для розпізнавання зображень і мови, машинного перекладу, медичної діагностики, фінансового прогнозування, розробки автономних автомобілів та багато іншого.

На жаль нейронні мережі мають ряд обмежень - для навчання ефективних моделей потрібні великі обсяги даних і значні обчислювальні ресурси; процес навчання нейронних мереж може бути досить складним і вимагати глибоких знань в цій галузі.

Незважаючи на це, нейронні мережі - одна з найбільш перспективних технологій сьогодення. Вони відкривають нові можливості в різних сферах людської діяльності і продовжують активно розвиватися, стають все більш досконалими і доступними, тому набули великого розповсюдження.

З урахуванням завдань товарообороту слід окремо більш докладно зупинитися на рекурентних нейронних мережах (RNN).

RNN - запам'ятовують попередню інформацію, що робить їх потужнішими за традиційні нейронні мережі, які обробляють інформацію окремо одну від однієї. Також, здатні працювати з контекстом та вибирати варіант рішення на основі попередньої інформації. завдяки чому найбільш підходять для завдань, які вимагають комплексного імпорту даних, наприклад, обробки людської мови, аналізу часових рядків і створення тексту.

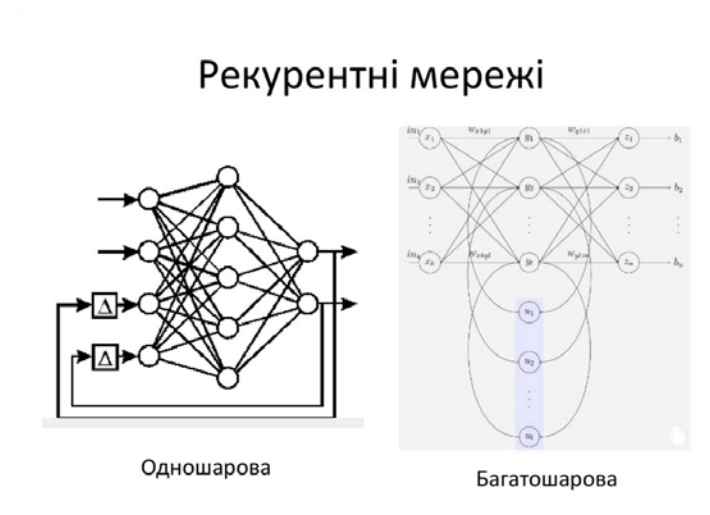


Рис. 1.5 Принципова схема рекурентних мереж

Базовим елементом RNN є петлі зворотного зв'язку, які дозволяють інформації циркулювати всередині мережі (рис. 1.5) чим створюється ефект "пам'яті", оскільки мережа може зберігати інформацію про попередні стани.

RNN виявляють довгострокові залежності в даних, що є важливим для розуміння складних структур мови або прогноз майбутніх значень в часових рядах.

Однак, ці моделі стикаються з проблемою зникаючих градієнтів, яка ускладнює навчання мереж на довгих послідовностях, для чого були розроблені різні модифікації, такі як довгострокова короткочасна пам'ять (Long Short-Term Memory, LSTM) та одиниці «входів» (Gated Recurrent Units, GRU). Ці архітектури дозволяють мережам краще запам'ятовувати інформацію на тривалих відрізках часу і ефективніше навчатися на складних завданнях [19].

Застосування RNN охоплює широкий спектр областей. У природній обробці мови RNN використовуються для машинного перекладу, аналізу настроїв, генерації тексту та розпізнавання мови. В аналізі часових рядів RNN застосовуються для прогнозування цін на акції, погоди та інших явищ, що змінюються з часом. Крім того, RNN використовуються в системах рекомендацій, розпізнаванні образів та багатьох інших областях, де важливо враховувати послідовність даних.

Отже, рекурентні нейронні мережі - це потужний інструмент що дозволяє створювати моделі, здатні навчатися на послідовностях даних і виявляти складні залежності між факторами. Завдяки своїм можливостям, RNN все частіше

застосовують в різних сферах діяльності, стимулюючи розвиток штучного інтелекту.

Характерною особливістю є те, що алгоритми нейронних мереж можуть адаптуватися до змінних умов, що особливо важливо в динамічному ринковому середовищі та дозволяють стабільно і швидко обробляти великі обсяги інформації.

З використанням даних технологій для прогнозування попиту на товари значно підвищується ефективність заходів по оптимізації закупівлі товарів, уникати дефіциту або надлишку товарів на складі, мінімізувати витрати на транспортування, скорочення часу логістики та підвищення їх ефективності, зменшення витрат на зберігання товарів, прискорення обробки замовлень та підвищення ефективності роботи складу, прогнозування інтересів клієнтів та пропонування їм найбільш релевантних товарів.

Вибір конкретного алгоритму залежить від ряду факторів, таких як: типу даних; складності задачі; доступних ресурсів; вимог до точності.

Слід зазначити що для ефективного використання алгоритмів необхідно мати якісні та достовірні дані.

Важливим аспектом також постійний моніторинг та корегування. Моделі прогнозування та оптимізації потребують постійного моніторингу та корегування з урахуванням змін в ринкових умовах та поведінці споживачів.

Для ефективного використання результатів прогнозування та оптимізації необхідно інтегрувати відповідні системи з іншими бізнес-системами компанії (ERP, CRM, WMS).

Застосування алгоритмів прогнозування попиту та оптимізації логістики є ключовим фактором успіху для сучасних торгових компаній. Правильний вибір та використання цих алгоритмів дозволяє значно підвищити ефективність бізнесу та забезпечити конкурентні переваги.

1.1.2. Аналіз існуючих бібліотек машинного навчання

Застосування методів штучного інтелекту машинного навчання для оптимізації логістичних процесів набуває дедалі більшої значущості в умовах сучасної бізнес-середовища.

З метою вирішення задач логістичної оптимізації доступний широкий спектр бібліотек та інструментальних засобів машинного навчання, зокрема найбільш потужні і популярні представлено нижче.

Бібліотека TensorFlow, open-source бібліотека машинного навчання, розробки Google.

TensorFlow може працювати на різних платформах, від мобільних пристроїв до кластерів серверів. Існує багато ресурсів, навчальних матеріалів та підтримки для розробників, програмування можна здійснювати на таких мовах як Python (основна мова), C++, Java та інші. Наявна TensorBoard для візуалізації та аналізу моделей

Для ефективного використання TensorFlow потрібні знання математики, статистики та програмування. Складно налагоджувати та оптимізувати моделі.

Бібліотеку використовують для різного роду аналізу та прогнозування товарообороту.

Зокрема вона надає можливість здійснювати аналіз історичних даних про продажі, сезонності, акції та зовнішні фактори для прогнозування майбутнього попиту на товари. Це дозволяє оптимізувати запаси та уникнути дефіциту або надлишку товарів. TensorFlow часто використовують торговими мережами для прогнозування попиту на певні товари в різних сегментах ринку по регіонах, враховуючи місцеві особливості та кон'юнктуру.

Не менш важливим здатність бібліотеки до оптимізації логістики. Алгоритми TensorFlow можуть враховувати багато факторів, таких як відстань, затори, час доставки та вартість, для оптимізації маршрутів руху товарів. Бібліотека застосовується для динамічної оптимізації маршрутів кур'єрів в режимі реального часу, враховуючи зміни дорожньої ситуації.

Бібліотеку застосовують також для оптимізації управління складом. Аналіз даних про рух товарів на складі для оптимізації розміщення товарів, мінімізації часу на збір та відвантаження замовлень.

На основі доступних джерел прикладом використання бібліотеки можна навести американську компанію UPS (United Parcel Service), що здійснює свою діяльність на експрес-доставці та логістиці. Компанія UPS використовує TensorFlow для оптимізації маршрутів логістики, що дозволяє зменшити витрати на паливо та покращити час доставки.

Оремо слід зазначити такий програмний продукт як Keras. Це API для нейронних мереж, написаний на Python, працює поверх бібліотек машинного навчання, зокрема TensorFlow, та призначений для спрощення процесу створення та навчання моделей глибокого навчання. Хоча Keras сам по собі не є повноцінною бібліотекою для оптимізації логістики, він є потужним інструментом для побудови моделей, які можуть бути використані для вирішення різних логістичних задач.

PyTorch – відкрита бібліотека машинного навчання розроблена групою дослідження штучного інтелекту компанії Facebook на основі відкритої бібліотеки машинного навчання та наукових обчислень Torch, що написаний на скриптовій мові програмування Lua. Основний інтерфейсом бібліотеки розроблений під мову Python, підтримуються також і для C++.

Бібліотека дозволяє прогнозувати точний часу доставки замовлень клієнтам та використовується інтернет-магазинами для надання клієнтам більш точних термінів доставки, є ефективною також для аналізу зображень для контролю якості, використовуються моделі комп'ютерного зору на основі PyTorch для автоматичного контролю якості товарів на складі або під час транспортування. Використовується для прогнозування попиту та оптимізації запасів з використанням рекурентних нейронних мереж (RNN) для врахування часових залежностей.

Компанія Walmart яка являє собою багатонаціональну корпорацію роздрібною торгівлі застосовує PyTorch для прогнозування попиту на товари, щоб оптимізувати запаси та зменшити витрати на зберігання.

Scikit-learn - це бібліотека написана на Python та широко використовує NumPy, що надає прості та ефективні інструменти для машинного навчання, такі як класифікація, регресія, кластеризація, зменшення розмірності та вибір моделей.

Бібліотека використовується у випадках коли виникає потреба у кластеризації клієнтів. Сегментація клієнтів за їх купівельною поведінкою для персоналізації пропозицій та оптимізації логістичних процесів. З її допомогою прогнозують також відтік клієнтів. Здійснюється аналіз даних про клієнтів для прогнозування ймовірності їх відтоку та вжиття заходів для їх утримання.

Переваги бібліотеки - простота використання API, велика колекція алгоритмів, докладна документація з прикладами.

Недоліки - обмежена підтримка глибокого навчання, не призначена для дуже складних задач.

Scikit-learn ефективна для аналізу даних про замовлення, виявлення закономірностей та аномалій даних необхідних для оптимізації ланцюга поставок.

Statsmodels – бібліотека Python, що надає інструменти для статистичного моделювання та аналізу даних.

Бібліотека містить широкий спектр статистичних моделей та варіантів тестів, зручний набір інструментів для аналізу даних та перевірки робочих гіпотез. Використовується для аналізу часових рядів для прогнозування попиту з використання моделей ARIMA та інших методів для прогнозування попиту на товари на основі історичних даних. Застосовують також для виконання регресійного аналізу для визначення факторів, що впливають на вартість доставки: Аналізують дані для визначення основних факторів, що впливають на вартість доставки, таких як відстань, вага, тип транспорту та інші.

Однак, також є недолік, який полягає у менш орієнтованості на машинне навчання порівняно з TensorFlow та PyTorch.

Accord.NET – це потужний фреймворк для машинного навчання, обробки зображень, аудіо та сигналів на мові C#, є частиною .NET Framework компанії

Microsoft. Містить велику кількість алгоритмів, включаючи класифікацію, регресію, кластеризацію, оптимізацію та нейронні мережі.

Використовується для прогнозування попиту та оптимізації запасів на основі алгоритмів регресії, оптимізації маршрутів із застосуванням алгоритмів оптимізації та знаходження найкоротших або найефективніших маршрутів доставки; сегментації клієнтів за географічним розташуванням або частотою замовлень для оптимізації логістичних операцій.

Однак, має наступні недоліки, які полягають у сповільненій розробці фреймворку, хоча він також залишається функціональним; поступатися за продуктивністю оптимізованим бібліотекам на C++ або Python (TensorFlow або PyTorch).

ML.NET – це кросплатформенний фреймворк машинного навчання від Microsoft, розроблений спеціально для .NET-розробників на мові C#. Він дозволяє створювати кастомні моделі машинного навчання та інтегрувати їх у існуючі .NET-додатки. Також, тісна інтеграція з .NET-екосистемою, простота у використанні для .NET-розробниками; підтримує різні алгоритми машинного навчання; дозволяє розгортати моделі на різних платформах (Windows, Linux, macOS). Проте, може поступатися за гнучкістю та кількістю доступних алгоритмів порівняно з більш зрілими фреймворками, такими як TensorFlow або PyTorch.

За допомогою ML.NET прогнозують час доставки нових замовлень, оптимізують складські операції з розміщення товарів та скорочення часу на збирання замовлень, виявляють нетипові події у логістичних процесах, (таких як затримки доставки або втрата вантажу).

Підбір бібліотеки залежить від обраного завдання, наявних ресурсів та кваліфікованості команди. Для підвищення ефективності застосовують комбіноване використання різних бібліотек для досягнення найкращих результатів. Наприклад, для складних задач глибокого навчання можна використовувати TensorFlow або PyTorch, а для більш простих задач аналізу даних та статистичного моделювання – scikit-learn або Statsmodels.

C# хоча й не є найпопулярнішою мовою для розробки ІІІ, оскільки лідерство тримають Python та R, все ж має низку бібліотек, які можна використовувати для створення інтелектуальних систем, зокрема для оптимізації логістичних процесів.

З огляду ефективність створення програм для ERP-систем підприємств України більш ефективним зокрема в масштабуванні програмних продуктів є використання бібліотек написаних на мові C#.

C# та платформа .NET існують вже досить давно, що забезпечує стабільність, велику кількість документації та зрілу екосистему, забезпечує високу продуктивність оскільки програма компілюється в машинний код, що важливу для обробки великих обсягів даних в ERP-системах

Багато підприємств в Україні використовують Windows Server та інші продукти Microsoft, тому інтеграція з ними за допомогою C# є більш простою та ефективною.

Наявні потужні інструменти розробки, а саме Visual Studio - потужний IDE з розвиненими засобами відлагодження, профілювання та тестування, що спрощує розробку складних ERP-систем.

Важливим фактором є актуальність бібліотек машинного навчання оскільки Microsoft активно підтримує .NET, що гарантує подальший розвиток та оновлення платформи

1.2. Аналіз сучасних методів управління товарними потоками, використання ERP-системи для управління бізнес-процесами

Управління товарними потоками охоплює всі етапи руху товарів від постачальника до кінцевого споживача, включаючи закупівлю, зберігання, транспортування, розподіл та продаж. Сучасні методи управління товарними потоками спрямовані на оптимізацію цих процесів з метою мінімізації витрат, підвищення швидкості доставки та покращення рівня обслуговування клієнтів [14].

Усі методи будуються на таких головних принципах організації процесу:

Just-in-Time (JIT) "Точно вчасно" - концепція, яка передбачає отримання товарів лише тоді, коли вони необхідні для виробництва або продажу. Це дозволяє мінімізувати запаси на складі та зменшити витрати на зберігання.

Lean Manufacturing (Бережливе виробництво) - методологія, спрямована на усунення всіх видів втрат у виробничому процесі, включаючи надлишки запасів, час очікування, зайві переміщення та дефекти.

Agile Supply Chain (Гнучкий ланцюг поставок) - концепція, яка передбачає швидку адаптацію ланцюга поставок до змін у попиту та ринкових умовах.

Demand-Driven Supply Chain (Ланцюг поставок, керований попитом) - підхід, який базується на прогнозуванні попиту та реагуванні на реальний попит, а не на прогнозах виробництва.

Cross-Docking - процес, при якому товари відразу після отримання на склад перевантажують на транспортні засоби для доставки клієнтам, минаючи етап зберігання.

Vendor Managed Inventory (VMI) «Управління запасами постачальником» - коли постачальник бере на себе відповідальність за управління запасами на складі покупця.

ABC-аналіз - метод класифікації товарів за їх значенням для компанії, що дозволяє оптимізувати управління запасами для різних категорій товарів.

RFID (Radio-Frequency Identification) - технологія радіочастотної ідентифікації, яка дозволяє відстежувати рух товарів в реальному часі.

Автоматизація складських процесів - використання автоматизованих систем зберігання та видачі товарів, роботизованих систем та іншого обладнання для оптимізації складських операцій.

ERP-системи — це як невидимий двигун, що приводить у рух сучасний бізнес. Вони об'єднують усі ключові процеси компанії в одну суперструктуру, що працює як єдиний мозок, що керує всіма фінансами до логістики. Це більше, ніж просто програмне забезпечення — це серце компанії, яке забезпечує, щоб усі частини працювали в синхронії, без зайвих витрат часу та ресурсів.

ERP дозволяє приймати рішення швидше й точніше, використовуючи актуальні дані, які під рукою 24/7. Від автоматизації закупівель до управління запасами та логістикою — ці системи оптимізують шкідливу операцію, скорочують витрати, виключають дефіцит і гарантують, що клієнт завжди отримує.

Одною з ключових переваг ERP є інтеграція. Усі відділи компанії працюють як єдиний організм, використовуючи одну загальну базу даних. Це не просто знижує ризик помилок, це створює середовище, де кожен співробітник має доступ до необхідної інформації в потрібний момент. Таке з'єднання технологій і процесів створює бізнес, який думає швидше, рухається ефективніше і навпаки

ERP-системи беруть на себе виконання багатьох рутинних завдань, звільняючи співробітників для виконання більш творчих та аналітичних функцій. Наприклад, ERP-системи автоматизують процеси обробки замовлень, управління відносинами з постачальниками, відстеження руху товарів на складі та планування доставки. Це дозволяє значно скоротити час на виконання цих завдань та підвищити їхню точність.

ERP-системи — це ваш незамінний партнер у боротьбі з рутинними процесами. Вони беруть на себе всю «чорну роботу», звільняючи команди для виконання завдань, які дійсно мають значення — аналітики, стратегічного планування та творчих проривів. Автоматизоване оброблення замовлень, бездоганна співпраця з постачальниками, контроль за логістикою на складі та точне планування доставки — все це стає реальністю за допомогою ERP. Ви отримуєте більшу точність, менше часу на марні зусилля і жодних компромісів.

Сьогодні ERP — це не просто система, а мозок сучасного підприємства. Вона оптимізує процеси, підвищує ефективність до нового рівня та тримають вашу компанію в авангарді ринку. Завдяки таким технологіям ви не просто досягаєте стратегічні цілі, а створюється фундамент для стабільного та тривалого зростання. У світі де швидкість і точність вирішують усе, ERP — це ваш квантовий стрибок у майбутньому бізнесі.

Ключові функціональні можливості ERP-систем, це підсистеми управління: закупівлями, запасами, виробництвом, продажами, фінансами, персоналом.

Рішення щодо вибору ERP-системи – це стратегічний крок, який може змінити фінансову політику компанії. Це не просто вибір програмного забезпечення, а інвестиція у ефективність і конкурентоспроможність на роки вперед. Ідеальна система – це та, яка враховує масштаби бізнесу, нюанси індустрії, фінансові рамки та унікальні бізнес-процеси, що перед конкурентами.

Перше, ERP-система має забезпечувати повний набір функціональних можливостей, необхідних для ефективного управління всіма аспектами бізнесу компанії. Це включає в себе модулі для управління фінансами, обліку, виробництвом, логістикою, управління персоналом та іншими ключовими процесами. Важливо, щоб система не просто автоматизувала рутинні операції, а й оптимізувала їх, підвищуючи загальну ефективність.

Друге, сучасні компанії використовують широкий спектр інформаційних систем, тому здатність ERP-системи до інтеграції з іншими програмними продуктами є критично важливою. Система повинна легко обмінюватися даними з системами бухгалтерського обліку, CRM-системами, системами управління складськими запасами та іншими додатками, забезпечуючи єдину інформаційну картину для прийняття рішень.

ERP-система повинна володіти здатністю масштабованості, яка забезпечує її властивість відповідати зростаючим запитам бізнесу. Це означає, що система повинна безперешкодно застосовуватися до збільшення обсягів діяльності, кількості користувачів, а також до впровадження нових функцій. Завдяки масштабованості компанія отримує можливість уникнути потреби в заміні системи при розширенні

Вартість ERP-системи – це ще один важливий фактор, який слід враховувати при виборі. Вартість включає в себе не тільки ліцензійні платежі, але й витрати на впровадження, навчання персоналу та подальшу технічну підтримку.

Окрему увагу варто відзначити якістю технічної підтримки, що її пропонує постачальник ERP-системи. Надійний сервіс повинен включати оперативне усунення технічних несправностей, професійні консультації, навчання персоналу та регулярне оновлення програмного забезпечення. Компетентний постачальник

стане запорукою стабільної роботи системи, допомагаючи бізнесу максимально розкрити його потенціал та сприяти його подальшому розвитку

Остаточний вибір ERP-системи є складним і багатогранним завданням, що забезпечує глибоке вивчення специфіки діяльності підприємства та ретельний аналіз наявних рішень на ринку. Впровадження правильно обраної системи здатне суттєво оптимізувати бізнес-процеси, порушити операційні витрати та створити основу для сталого

Застосування сучасних методів організації товарних потоків у поєднанні з впровадженням ERP-системи стає ключовим інструментом підвищення продуктивності та конкурентоспроможності підприємств.

На сьогодні в Україні найбільше розповсюдження отримали ERP-системи, деякі з яких наведені нижче.

BAS ERP – це українська система, орієнтована на середній і великий бізнес. Вона включає модулі для обліку, управління виробничими процесами, логістики та CRM. Система легко інтегрується з інструментами аналізу даних і звітності через API та спеціалізоване розширення. BAS ERP враховує вимоги українського законодавства, що робить її оптимальною для вітчизняних підприємств.

SAP ERP – це глобальне рішення, німецької компанії SAP, яке підтримує масштабування та індивідуальні налаштування. SAP має потужні інструменти для бізнес-аналітики, фінансового прогнозування та управління складними ланцюгами постачання. Пропонує широкий спектр можливостей для інтеграції через стандартизовані протоколи.

Висока вартість впровадження не робить SAP популярним серед великих підприємств, які потребують комплексних рішень.

Microsoft Dynamics 365 – це комплексна платформа, яка включає ERP та CRM функціонал. Вона пропонує гнучкі можливості налаштування та інтеграції з іншими продуктами Microsoft. Dynamics 365 підходить для компаній різного розміру, від малого бізнесу до великих корпорацій. Має потужний функціонал, гнучкість, інтеграція з екосистемою Microsoft, зручний інтерфейс. Недолік - вища

вартість порівняно з BAS ERP, може потребувати значних витрат на впровадження та налаштування.

Odoo – це ERP-система з відкритим кодом бельгійської компанії, яка пропонує широкий спектр модулів для управління різними бізнес-процесами. Odoo відома своєю гнучкістю та можливістю кастомізації, відкритий код, гнучкість, модульна структура, відносно низька вартість, проте, може потребувати значних зусиль на налаштування та кастомізацію, менша кількість готових рішень для українського ринку порівняно з BAS ERP.

IT-Enterprise – це українська ERP-система, яка пропонує широкий функціонал для управління різними аспектами діяльності підприємства. Вона орієнтована на підприємства різних галузей, включаючи виробництво, торгівлю та послуги. Добре адаптована до українського ринку, має широкий функціонал, можливість кастомізації. На даний час менша поширеність порівняно з BAS ERP та міжнародними аналогами.

Oracle ERP – ERP-система від міжнародного гіганта в сфері програмного забезпечення Oracle і являє собою інтегрований хмарний комплекс застосунків для управління фінансами, закупками та портфелем проєктів для малого, середнього та великого бізнесу

BJet – хмарна ERP-система, спільна розробка Odoo та української компанії BusinessJetPro. Примітною особливістю є можливість побудови облікової системи компанії на базі колекції функціональних модулів.

OneBox - це українська хмарна платформа, яка пропонує комплексний набір інструментів для автоматизації та управління різноманітними бізнес-процесами. Вона діє за принципом конструктора так як і BJet.

Також присутні на ринку систем та займають невеликий сегмент в Україні наступні ERP-системи: Універсал української компанії СофтПро, індійська open-source система ERPNext, шведські системи IFS Applications та HansaWorld.

Висновки по розділу 1

Розглянута роль ERP у забезпеченні інтеграції та аналітики. ERP-системи дозволили об'єднати всі ключові бізнес-процеси в єдиній інформаційній системі. це значно покращує підвищену прозорість даних, сприяє скороченню ручної роботи та покращенню аналітики обробки.

Інтеграція ERP із технологіями машинного навчання, забезпечує ще більшу ефективність, дозволяючи проводити складний аналіз і формувати точні прогнози.

РОЗДІЛ 2. РОЗРОБКА ПРОГРАМНОЇ БІБЛІОТЕКИ

2.1. Інтеграція бібліотек штучного інтелекту з обліковими системами

Інтеграція бібліотек AutoML (автоматизоване машинне навчання) з ERP-системами є складним процесом, що вимагає остаточного планування та реалізації. Основними технічними аспектами цієї інтеграції є використання API та COM-інтерфейсів

Перед інтеграцією необхідно вивчити, який API надає ERP-система (наприклад, BAS, SAP,). Це може бути RESTful API, SOAP або інші протоколи. Бібліотека повинна мати можливість взаємодіяти з API ERP-системи. Це включає розробку відповідних методів для отримання та передачі даних.

Аутентифікація та авторизація: забезпечення безпечного доступу до API через механізми аутентифікації та налаштування правого доступу. ERP-системи які розміщуються не серверах кожного окремого підприємства, можуть підключатися через COM-інтерфейси для інтеграції із зовнішніми додатками. можливо перевірити наявність такої підтримки.

Якщо зареєструвати бібліотеку AutoML як COM-об'єкта, то це дозволяє ERP-системі запускати методи бібліотеки разом із власними вбудованими методами та функціями. Реєстрація включає створення відповідних записів у системному реєстрі. Необхідне також визначення методів та властивостей, які будуть доступні через COM, з урахуванням специфіки ERP-системи. На першому етапі необхідно виконати налаштування середовища, забезпечення сумісності версії програмного забезпечення для коректної інтеграції роботи. Далі обов'язковою є перевірка коректності обміну даними між ERP-системою та бібліотекою AutoML, включаючи тестування за різними сценаріями використання.

Важливо також провести аналіз швидкодії інтегрованої системи та внесення деяких оптимізацій для забезпечення ефективної роботи.

Документування повинна забезпечити докладний опис процесу інтеграції, використаних API та COM-інтерфейсів, а також інструкції для розробників та користувачів.

Необхідно забезпечити навчання персоналу, проведення тренінгів для співробітників щоб забезпечити використання функцій наданих інтегрованою системою.

Утримання цих кроків забезпечить успішну інтеграцію бібліотек AutoML з ERP-системами, що призведе до підвищення ефективності бізнес-процесів та конкурентоспроможності підприємства.

Впровадження AutoML у бізнес-процеси може суттєво підвищити ефективність та конкурентоспроможність підприємства.

Інтеграція бібліотеки машинного навчання в бізнес-процеси – це багатогранний процес, який вимагає використання підходу. Вона передбачає не лише адаптацію існуючих технологій, але й перегляд організаційних процесів. Впровадження AutoML супроводжує недостатність аналізу та реінжинірингу поточних бізнес-операцій, що може включати зміну процедур, перерозподіл завдань і оновлення стандартів. Для цього необхідно детально дослідити робочі процеси, щоб забезпечити їх сумісність із сучасними технологіями.

Один із ключових викликів — інтеграція нових рішень із чинними ІТ-системами. Проблеми сумісності, складні інтерфейси чи обмеження масштабованості можуть стати серйозними бар'єрами. Вирішення цих питань вимагає проведення технічного аудиту та створення плану модернізації. Паралельно викликати потребу у значних фінансових вкладеннях, тому важливо оцінити економічну доцільність і розробити стратегію впровадження, яка дозволить оптимізувати витрати.

Важливу роль у процесі розвитку людський фактор. Нерідко працівники виявляють опір змінам через страх перед новим або побоювання щодо втрати робочих місць. У цьому контексті критичною стає прозора комунікації: працівникам необхідно пояснити мету змін, акцентувати переваги та залучити їх до впровадження. Ця замученість сприяє подоланню опори і створює сприятливі умови для адаптації. Успіх впровадження залежить від якісного планування. Ретельно розроблений план має охоплювати технічні, фінансові та людські аспекти, забезпечуючи чітке виявлення про наступність дій, забезпечення ресурсів

та наявні ризики. Розпочати варто з пілотних проєктів, які після оцінки реальних можливостей, виявити слабкі місця та вдосконалити робочі процеси перед масштабним впровадженням.

Ефективне використання AutoML неможливо без залучення фахівців у галузь машинного навчання. Їхній досвід дозволяє оптимізувати вибір алгоритмів, налаштування моделей та рішення складних технічних завдань. Ця співпраця суттєво знижує ризики невдач і вашої якості впроваджуваних рішень. Значення має також роботу з персоналом. Співробітники повинні розуміти, чому запроваджуються нові технології, які вигоди вони принесуть, і відчувати свою важливість у цьому процесі. Відкрита комунікація полегшує формування позитивного ставлення до змін і полегшує їх впровадження.

Загалом, успішна інтеграція бібліотеки машинного навчання забезпечує багатосторонній підхід, який поєднує технічну підготовку, фінансовий аналіз, пілотне тестування, залучення професійних кадрів та активну роботу з людським фактором. Урахування всіх цих аспектів дозволяє компаніям ефективно адаптувати інновації, подолати виклики та досягти очікуваних результатів.

2.2. Обґрунтування вибору методів прогнозування та програмних засобів для реалізації програмного продукту

AutoML радикально змінює підхід до створення моделей прогнозування, створюючи процес доступним навіть для тих, хто не є експертами в машинному навчанні. Ця технологія автоматизує вибір алгоритмів, налаштування гіперпараметрів та підготовку даних, що дозволяє бізнесам легко інтегрувати її у свої процеси. Результат? Точніше прогнозування, ефективніше управління і, як наслідок, значно кращі бізнес-результати.

AutoML інтегрується з ERP-системами, можливості компанії зростають, якщо ще більше. Це забезпечує миттєвий аналіз великих обсягів даних, що дозволяє передбачати запит, оптимізувати запаси та будувати сценарії, які підвищують ефективність. Підприємства підтримують інструмент, який швидше ухвалить поточні рішення, зменшить витрати та залишиться конкурентоспроможними.

Головні виклики, з якими стикаються торговельні підприємства, — це неточні прогнози, перевищення запасів та додаткові витрати на логістику. Впровадження AutoML змінює правила гри. Воно автоматизує процеси прогнозування та оптимізації, допомагаючи скоротити витрати, уникати надлишків і створювати бізнес-модель, яка працює значно ефективніше.

Бібліотека має бути здатна миттєво обробляти великі масиви даних, забезпечуючи постійне оновлення прогнозів і рекомендацій. Це дає бізнесу можливість діяти на випередження. Точність моделей, які генерують систему, є критично важливою. Вони повинні бути досить ефективними, щоб вирішити задачі прогнозування попиту чи оптимізації запасів з максимальною точністю. Але точність — це лише частина картини. Адаптивність стає ключовою, коли дані змінюються. Сезонність, тенденції, нові фактори — усе це бібліотека втрачена, залишаючись актуальною має навіть у динамічних умовах. Сучасна платформа має бути масштабованою [10].

Завдяки автоматичному вибору алгоритмів і налаштувань, бібліотека може працювати навіть без залучення експертів. Це включає автоматичний збір, очищення та підготовку даних для подальшого моделювання. Усе це відбувається у фоновому режимі, дозволяючи зосередитися на результатах.

AutoML змінює логістику, спрощуючи процеси, які раніше вимагали глибокої експертизи в машинному навчанні. Ця технологія відкриває можливості створення потужних великих моделей навіть для компаній, які не мають у своєму розпорядженні команди дата-сайєнтистів.

Одним із ключових факторів, що робить AutoML таким ефективним, є автоматизація підбору алгоритмів і гіперпараметрів. Система аналізує дані та самостійно обирає найкращі рішення, наприклад, використовуючи XGBoost для прогнозування запиту або LSTM для аналізу часових рядів. Це не тільки прискорює процес розробки моделей, але й забезпечує їхню високу точність. Крім того, AutoML повністю змінює підхід до роботи з даними, автоматизуючи процеси очищення, нормалізації та трансформації. У сфері, де дані часто є хаотичними чи неповними, це гарантує їхню високу якість і створює основу для впровадження

ефективних рішень. Його застосування в логістиці охоплює багато напрямків — від прогнозування запиту та оптимізації маршрутів доставки до управління запасами. AutoML працює з такими алгоритмами, як ARIMA, LSTM, рішення дерев чи градієнтний бустинг, щоб забезпечити найбільш ефективні результати для кожного конкретного завдання.

Важливою особливістю AutoML є можливість порівняння результатів різних моделей. Система автоматично обчислює метрики якості, такі як MAE, RMSE та R2, що дозволяє вибрати найкращу модель для конкретного завдання. Це забезпечує високу точність прогнозів і прийняття обґрунтованих рішень. Інтеграція AutoML з ERP-системами дозволяє працювати з великими обсягами даних в реальному часі, що є критично важливим для сучасних логістичних операцій. Це забезпечує швидке прийняття рішень і адаптацію до змін ринку.

Підсумовуючи, AutoML є потужним інструментом для автоматизації складних завдань в логістиці. Він дозволяє компаніям підвищити ефективність своїх операцій, знизити витрати і приймати більш обґрунтовані рішення. Завдяки автоматизації підбору алгоритмів, гіперпараметрів і підготовки даних, AutoML робить машинне навчання доступним для широкого кола користувачів.

Для вирішення задач оптимізації товарообороту найчастіше застосовуються наступні методи:

XGBoost - метод градієнтного бустингу, який дає високу точності прогнозів, є методом швидких обчислень із здатністю обробляти великі обсяги даних.

Функція втрат $L(\Theta)$ мінімізується шляхом додавання нових дерев:

$$L(\Theta) = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega(f_k), \text{ де}$$

l — функція втрат,

Ω — регуляризаційний член для запобігання перенавчанню.

Використовуючи історичні дані продажів і фактори, такі як сезонність, ціни та акції, XGBoost допомагає передбачити майбутній попит із точністю до 95%.

LSTM (Long Short-Term Memory). Являє собою рекурентну нейронну мережу (RNN), яка ефективно працює з часовими рядами завдяки здатності запам'ятовувати довгострокові залежності між даними.

Оновлення станів пам'яті відбувається через гейт-системи:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t,$$

де C_t — стан пам'яті,

f_t — гейт забування,

i_t — вхідний гейт,

$\tilde{C}_t$ — новий стан.

Аналіз часових рядів для прогнозування попиту на різні товари в різні сезони.

Наприклад, LSTM може прогнозувати продажі залежно від історичних даних і маркетингових кампаній.

ARIMA (AutoRegressive Integrated Moving Average) - класичний метод аналізу та прогнозування часових рядів. Він поєднує авторегресію (AR), інтеграцію (I) та ковзне середнє (MA).

Модель ARIMA(p, d, q) розраховується рівнянням:

$$Y_t = c + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q},$$

де p — порядок AR,

d — порядок диференціювання,

q — порядок MA.

При прогнозуванні попиту на товари з урахуванням сезонності ARIMA дозволяє аналізувати історичні дані продажів і виявляти тренди. Порівняння характеристик алгоритмів моделей представлено в таблиці 2.1.

Таблиця 2.1

Порівняння ефективності моделей

Метрика	XGBoost	LSTM	ARIMA
Точність прогнозів	Висока (до 95%)	Висока (до 92%)	Середня (до 85%)
Сезонність	Частково враховує	Добре враховує	Добре враховує
Обсяг даних	Великі обсяги	Великі обсяги	Маленькі обсяги
Швидкість	Висока	Середня	Низька
Складність	Середня	Висока	Низька

Дані таблиці показують що, XGBoost є оптимальним для великих табличних даних, LSTM — для складних часових рядів, а ARIMA — для простих часових рядів із сезонністю. Вибір моделі залежить від специфіки завдання та доступних даних.

Розробка бібліотеки на C# з використанням ML.NET для вирішення завдань прогнозування товарообігу є виправданою в кількох аспектах, враховуючи особливості завдання та потреби бізнесу.

Використання C# та ML.NET забезпечить сумісність з існуючими системами багатьох українських компаній, які використовують .NET технології. Сучасні системи на платформі BAS, підтримують COM-інтерфейси, що дозволяє бібліотеці на C# без проблем інтегруватися з ними.

Бібліотека може бути використана як самостійний компонент, який легко інтегрувати в різні системи управління даними, такими як BAS, CRM чи ERP

ML.NET — це потужний інструмент, створений Microsoft для .NET-розробників, який відкриває нові горизонти в аналізі даних і прогнозуванні. Це не просто бібліотека, а ціла екосистема, яка здатна інтегрувати машинне навчання в існуючі процеси. Особливо це актуально для українського бізнесу, де оптимізація товарообороту може стати ключем до конкурентної переваги.

Гнучкість ML.NET — одна з її найсильніших сторін. Бібліотека швидко адаптується під різні задачі, підтримуючи як базові алгоритми по типу регресії чи класифікації, а також і розробку кастомних моделей, що враховують специфіку конкретного. Наприклад, сезонність продажів, ефективність маркетингових дій чи навіть вплив зовнішніх факторів можуть бути інтегровані разом в аналіз. Це рішення, яке не просто автоматизує процеси, а й робить їх розумними.

AutoML в ML.NET — це інструмент, який перетворює процес побудови моделей машинного навчання на щось максимально доступне. За допомогою AutoML можна швидко вибрати і налаштувати оптимальну модель навіть без глибоких технічних знань. Бібліотека легко взаємодіє з табличними даними, зокрема з CSV-файлами, і безперешкодно інтегрується з іншими системами, такими як BAS.

Інтеграція з LINQ і сучасними інструментами C# робить обробку даних природною і зрозумілою. Модульна структура платформи — це не просто про гнучкість, а про безмежні можливості. Завдання будь-якої стійкості, чи то прогнозування попиту, оптимізація запасів або виявлення аномалій, знаходять тут своє елегантне рішення.

Оскільки ML.NET — це відкритий і безкоштовний інструмент, додаткові витрати на ліцензії залишаються в минулому. Платформа без проблем інтегрується з іншими бібліотеками та хмарними сервісами, а автоматичний підбір алгоритмів оптимізує процес виконання завдань. Результат - максимальна ефективність і мінімум зайвого клопоту.

Поєднання C# та Visual Studio створює середовище, яке можливо розширює межі, забезпечуючи плавну інтеграцію ML.NET у сучасні системи. ML.NET є універсальним інструментом для машинного навчання, який може бути застосований в різних сферах бізнесу. Його гнучкість, простота використання та інтеграція з .NET-екосистемою роблять його відмінним вибором для українських компаній, які бажають оптимізувати свої процеси за допомогою штучного інтелекту.

ML.NET — це інструмент, який змінює правила гри в логістиці, де рішення базуються на даних. Уявлення про те, як цей факт може стати основою для ухвалення рішень, вже не є фантазією — це реальність, яка розгортається завдяки машинному навчанню. Можливість обробляти величезні обсяги інформації та виявляти приховані закономірності відкриває нові горизонти для оптимізації логістики.

Прогнозування запиту стало одним із ключових сценаріїв використання ML.NET. Історичні дані більше не просто архів, а актив, що формує майбутнє. Завдяки аналізу продажів минулих періодів створені моделі, здатні точно передбачати тенденції. Це дозволяє компаніям підтримувати баланс між дефіцитом і надлишками товарів. Не потрібно більше турбуватися про порожні поліси або зайві витрати на зберігання — оптимізація запасів стає результатом роботи моделі, яка безпомилково містить рішення. Логістика стає ефективнішою, коли

впроваджуються інноваційні технології. Наприклад, алгоритми машинного навчання через ML.NET здатні обробляти величезні обсяги даних – від відстаней і часу в дорозі до вартості палива й стану дороги. Це створює нову реальність для доставки, де маршрути залишаються не просто коротшими, а розумнішими.

Іншим інструментом є аналіз запасів. Системи на основі ML.NET просто не збирають, а формують стратегії. Наприклад, виявлення сезонних змін попиту чи розпізнавання трендів дозволяє уникнути перевантаження складів або їх порожнечі. В остаточному підсумку це про баланс – мати саме те, що потрібно, саме тоді, коли це потрібно

Ще однією важливою сферою застосування ML.NET є виявлення аномалій. Аналізуючи великі обсяги даних, модель може виявити незвичні або негативні патерни, які можуть свідчити про проблеми в логістичних процесах. Наприклад, ML.NET може виявити помилки в даних, випадки шахрайства або інші відхилення від норми. Це дозволяє компаніям швидко реагувати на проблеми та запобігати потенційним втратам.

2.3. Реалізація програмних модулів бібліотеки

Розробку модуля прогнозування товарних потоків із використанням AutoML здійснено засобами VisualStudio 2022, до складу якого включено бібліотеки ML.NET.

Для виконання прогнозування попиту та оптимізації управління запасами з урахуванням витрат була розроблена бібліотека-обгортка MLNetCOMWrapper.dll, яка базується на потужностях фреймворку ML.NET.

Ця бібліотека дозволяє створювати та застосовувати моделі машинного навчання для прогнозування майбутнього попиту на продукцію, що дає змогу оптимізувати рівні запасів та знизити витрати на зберігання та логістику.

Структура модуля бібліотеки представлена на рис. 2.1.

Модуль SalesOptimization реалізує систему автоматизованих продажів із використанням прогнозування бібліотеки ML.NET та включає основні компоненти, які забезпечують роботу з даними, навчання моделей, прогнозування та збереження результатів.

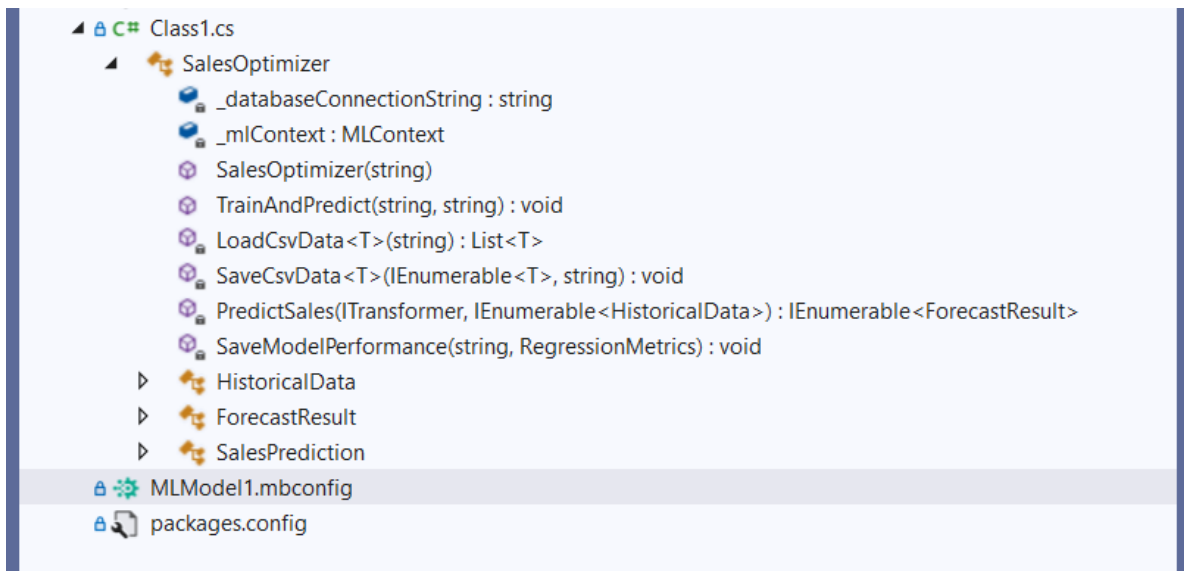


Рис. 2.1. Структура проекту бібліотеки

Опишемо основні компоненти модуля. Клас-обгортка (SalesOptimizer) - головний клас модуля, який виконує основні операції завантаження даних, навчання моделі, прогнозування, збереження результатів у базу даних або CSV-файл. Забезпечує взаємодію з бібліотекою ML.NET через MLContext та інтеграцію з базою даних через підключення SQL. Завантаження та обробка даних (LoadCsvData, SaveCsvData) - обробляє дані з файлів CSV, перетворюючи їх в об'єкти для моделей. Навчання моделі (TrainAndPredict) - використовує AutoML для автоматичного вибору найкращої моделі регресії. Прогнозування (PredictSales) – створює прогнози для товарів на основі навчальної моделі. Збереження результатів (SaveModelPerformance) - фіксує результати роботи моделі в базі даних для подальшого аналізу.

Завантаження даних здійснюється з CSV-файли або бази даних.

Функція LoadCsvData - зчитує дані із зовнішніх джерел, перетворюючи їх в об'єкти типу HistoricalData.

Оброблені вхідні дані передаються на AutoML для виконання розрахунків прогнозу.

Використовується компонент CreateRegressionExperiment, який виконує розділення даних на навчальну, тестову вибірки та підбір алгоритму та оптимізація гіперпараметрів та збереження найкращої моделі в файлі.

Історичні дані передаються у функцію PredictSales. Використовується метод CreatePredictionEngine, який виконує прогноз на основі навчальної моделі, формує список результатів у вигляді об'єктів ForecastResult, зберігає результати

Прогнози зберігаються у форматі CSV за допомогою SaveCsvData або у базу даних через метод SaveModelPerformance.

До ключових класів відносимо HistoricalData - представляє структуру вхідних даних: SalesVolume: (обсяг продажів), StockLevel (рівень запасів), DayOfWeek, Holiday (дані про день продажу). Використовується для навчання моделі; ForecastResult - містить прогнозовані дані: PredictedSales: (прогнозований обсяг продажів), ConfidenceLevel (довірчий рівень), GeneratedOn (дані створення прогнозу), SalesPrediction (відображає результат прогнозу для кожного запису).

Навчання моделі - використовуються історичні дані, які обробляються через AutoML

Алгоритми які можуть бути застосовані бібліотекою - SDCA, LightGBM, FastTree.

Показниками оцінки метрики які включено до функцій бібліотеки для оцінка точності прогнозів є: MAE, RMSE, MAPE.

Особливості архітектури бібліотеки наведено у таблиці 2.2.

Таблиця 2.2.

Особливості архітектури бібліотеки

Особливість	Пояснення
Модульність	Кожен компонент реалізує окрему функціональність (завантаження даних, прогнозування, оцінка), легко розширюється для додавання нових алгоритмів або джерел даних.
Інтеграція	Взаємодія з базами даних через ADO.NET та обмін даними через файли CSV.
Автоматизація	Використання AutoML зменшує потребу в ручному налаштуванні моделі.
Розширюваність	Можливість підключення інших форматів даних (JSON, XML).

До переваг віднесемо автоматизацію вибору моделей, яка зменшує потребу в експертизі машинного навчання, підтримка інтеграції з ERP-системами та сторонніми джерелами даних, може обробляти великі набори даних з мінімальними витратами ресурсів.

2.4. Інтеграція бібліотеки з ERP-системами

Технічна реалізація інтеграції програмного забезпечення через API та COM-інтерфейси є сучасним підходом до створення складних інформаційних систем, що забезпечує ефективність, масштабованість та гнучкість взаємодії між компонентами.

Інтеграція через API та COM-інтерфейси є ключовою частиною сучасної розробки програмного забезпечення, яка відкриває нові горизонти для взаємодії між системами. API (Application Programming Interface) можна встановити як набір чітко прописаних правил — договір між програмними компонентами. Він задає, як одна програма може звернутися до іншої з певним запитом і очікувати відповідь, гарантуючи передбачуваність та ефективність процесу. Цей стандарт об'єднує системи, виробляючи їх взаємодію прозорою і зручною.

У свою чергу COM (Component Object Model) підходить для інтеграції з іншого кута. Це бінарний стандарт, який дозволяє створювати компонентні об'єкти для повторного використання в різних програмах. Його сила може у здатності працювати на низькому рівні, забезпечуючи доступ до функціональних компонентів із максимальною гнучкістю. Це як створення будівельних блоків, які можна адаптувати до найрізноманітніших сценаріїв використання.

Обидва ці підходи мають свої унікальні сили, а їхнє правильне з'єднання відкриває шлях до створення ефективних і масштабованих рішень. API забезпечує легкість та універсальність інтеграції, тоді як COM забезпечує можливість глибокої взаємодії з компонентами системи. Технічна реалізація інтеграції через ці інтерфейси стає не лише викликом, а й нагодою для інноваційного переосмислення взаємодії між програмними продуктами.

Аналіз характеристик інтерфейсів представлено в таблиці 2.3.

Порівняння API та COM-інтерфейсів

Характеристика	API	COM-інтерфейси
Рівень абстракції	Вищий, більш абстрактний	Нижчий, більш низькорівневий
Технології	REST, SOAP, GraphQL, RPC	C++
Переносимість	Вища	Нижча
Складність розробки	Простіша, зазвичай використовують стандартні протоколи	Більш складна, вимагає глибокого розуміння технології COM
Продуктивність	Може бути нижчою через мережеві протоколи	Вища, оскільки немає мережевих затримок

COM-інтерфейси пропонують декілька переваг при розробці ERP-систем, особливо тих, що працюють без хмарних технологій. Їхня здатність забезпечувати пряму взаємодію між компонентами всередині однієї операційної системи суттєво оптимізує обробку даних, що є критично важливим для операцій, які оперують великими обсягами інформації. Такий підхід дозволяє досягти вищої продуктивності та ефективності системи.

Крім того, COM-інтерфейси надають розробникам більш глибокий рівень контролю над об'єктами та процесами, що є особливо корисним при створенні складних інтеграцій. Цей рівень деталізації дозволяє реалізувати більш витончені механізми безпеки та автентифікації.

Ще однією перевагою COM-інтерфейсів є їхня довга історія використання. Завдяки цьому існує велика кількість готових компонентів та бібліотек, що підтримують цей стандарт. Це значно спрощує процес розробки та дозволяє використовувати вже готові рішення для вирішення типових задач.

У межах інтеграції із системами машинного навчання використання COM-інтерфейсів надає значні переваги, зокрема в результатах розробки модулів для збору, обробки й аналізу даних, які інтегруються з ERP-системами. Завдяки цьому підходу забезпечується оптимізація процесів, розширюється процедура інсталяції та масштабування, а також покращується в

Так, використання СОМ-інтерфейсів у модулях дає можливість інтегрувати бібліотеку як частину стандартної бібліотеки ERP-системи, що істотно зменшує потребу в додаткових налаштуваннях. Це, у свою чергу, допоможе зручності впровадження та масштабованості. Крім того, завдяки СОМ-інтерфейсам модуль забезпечує ефективну взаємодію з базами даних, бібліотеками машинного навчання та іншими системними компонентами, дозволяючи інтегрувати різноманітні елементи в єдину функціональну систему. Окремої уваги заслуговує функціональність візуалізації та збереження даних, яка реалізується через СОМ-інтерфейс. Завдяки цьому забезпечується доступність і зручність роботи з отриманими результатами, включно з їх збереженням і використанням для прогнозу.

В цілому СОМ-інтерфейси виступають ефективним інструментом для створення ERP-системи, забезпечуючи високу продуктивність, гнучкість і можливість інтеграції з іншими технологіями.

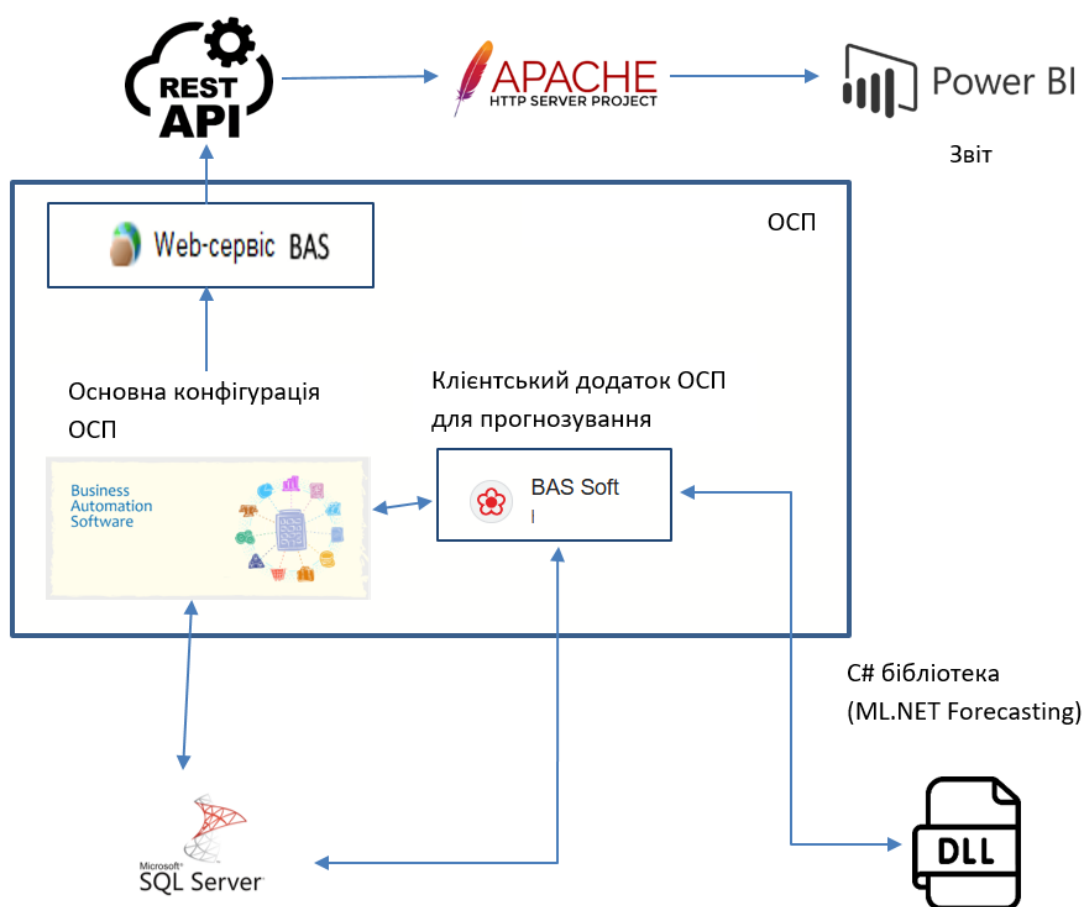


Рис. 2.2. Схема архітектури системи

Загальна структура архітектури комплексу програм підсистеми прогнозування товарообороту представлена на рис.2.2.

Для організації ETL-процесу даних системи обробка формує ряд таблиць про історичні дані а також результативні таблиці. З наведених нижче таблиць лише результуюча результативні формуються як додаткові і інсталюється в систему разом з обробкою шляхом стандартного процесу поновлення. Решта таблиць формується обробкою як проміжні віртуальні таблиці сформовані на основі історичних даних які накопичені у власних таблицях бази даних системи.

Опис структури таблиці SalesHistory (Історичні Дані) представлено в таблиці 2.4 і містить інформацію про минулі продажі та фактори, що впливають на запит.

Таблиця 2.4

SalesHistory (Історичні Дані)

Поле	Тип	Опис
SaleID	int	Унікальний ідентифікатор продажу
Date	data	Дата продажу
ProductID	int	Ідентифікатор товару
CustomerType	varchar(20)	Тип клієнта (Роздріб/Опт)
QuantitySold	int	Кількість проданих одиниць
Season	varchar(20)	Сезон (Зима/Літо/Весна/Осінь)

Таблиця віртуальна і формується обробкою з таблиць ERP-системи (Рис. 2.3).

Nº	Date	Product	CustomerType	CustomerTypeID	Season	Seaso...	ProductID	QuantityS...	SupplyID
1	03.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	1	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
2	04.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	4	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
3	07.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	2	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
4	08.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	4	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
5	09.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	-1	ГТД по импорту ГІ-000000017 от 09.05.2023 5:59:59
6	09.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	4	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
7	10.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	-1	ГТД по импорту ГІ-000000010 от 01.04.2023 5:59:59
8	10.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	3	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
9	11.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	1	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
10	15.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ПряміПродажі	1	Осінь	4	000013171	6	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
11	03.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ТорговіМережі	2	Осінь	4	000013171	12	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
12	07.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ТорговіМережі	2	Осінь	4	000013171	12	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
13	09.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ТорговіМережі	2	Осінь	4	000013171	12	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
14	10.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ТорговіМережі	2	Осінь	4	000013171	12	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
15	14.10.2024 0:0...	Майонез легкий 25% TM REMIA 250 мл	ТорговіМережі	2	Осінь	4	000013171	60	ГТД по импорту ГІ-000000032 от 21.05.2024 5:00:04
16	01.10.2024 0:0...	Сир Гауда нарізаний 150г, жир 48% TM Afa	ПряміПродажі	1	Осінь	4	000013211	13	ГТД по импорту АЗ-000000050 от 09.09.2024 5:59:59
17	03.10.2024 0:0...	Сир Гауда нарізаний 150г, жир 48% TM Afa	ПряміПродажі	1	Осінь	4	000013211	7	ГТД по импорту АЗ-000000050 от 09.09.2024 5:59:59

Рис. 2.3. Приклад формування віртуальної таблиці засобами ERP-системи

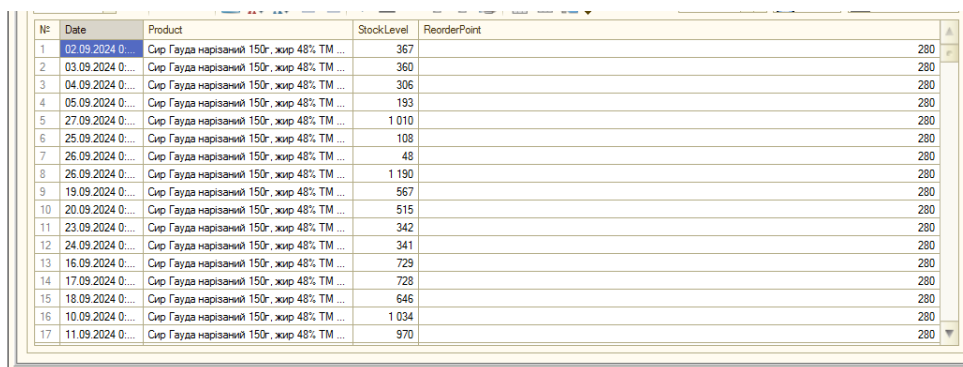
Дані про залишки товарів на складах збираються в таблиці InventoryLevels (таблиця 2.5.)

Таблиця 2.5

Структура таблиці InventoryLevels

Поле	Тип	Опис
InventoryID	int	Унікальний запис ідентифікатора
Date	data	Дата запису
ProductID	int	Ідентифікатор товару
StockLevel	int	Рівень запасів
ReorderPoint	int	Точка повторного замовлення

Таблиця InventoryLevels, віртуальна і формується обробкою з таблиць ERP-системи (рис 2.4).



Nr	Date	Product	StockLevel	ReorderPoint
1	02.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	367	280
2	03.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	360	280
3	04.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	306	280
4	05.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	193	280
5	27.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	1 010	280
6	25.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	108	280
7	26.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	48	280
8	26.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	1 190	280
9	19.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	567	280
10	20.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	515	280
11	23.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	342	280
12	24.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	341	280
13	16.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	729	280
14	17.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	728	280
15	18.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	646	280
16	10.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	1 034	280
17	11.09.2024 0...	Сир Гаура нарізаний 150г, жир 48% ТМ ...	970	280

Рис 2.4. Приклад формування віртуальної таблиці засобами ERP-системи

Дані про постачання товарів накопичуються в таблиці SupplyChainDetails (таблиця 2.6).

Таблиця 2.6

Структура таблиці SupplyChainDetails

Поле	тип	Опис
SupplyID	int	Унікальний ідентифікатор поставки
Date	data	Дата поставки
ProductID	int	Ідентифікатор товару
SupplierCountry	varchar(50)	Країна постачальник
SupplyChannel	varchar(20)	Канал (Канал1/Канал2)
TransitTime	int	Час доставки (дні)
DeliveryCost	decimal(10,2)	Вартість доставки

Таблиця ForecastResults Містити результати прогнозування запиту, згенеровану бібліотекою AutoML (таблиця 2.7).

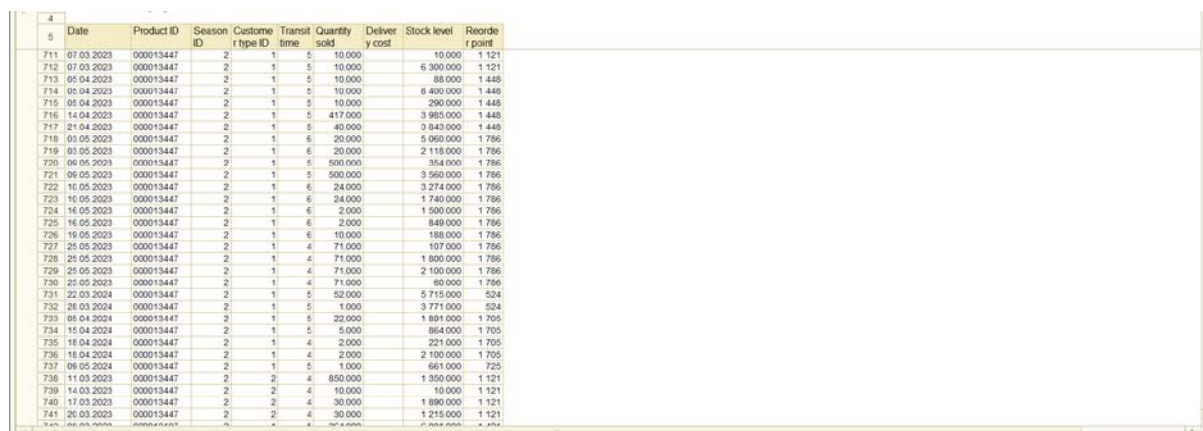
Таблиця 2.7

Структура таблиці ForecastResults (Результати прогнозу)

Поле	Тип даних (SQL)	Опис
ForecastID	varchar(9)	Унікальний ідентифікатор прогнозу (первинний ключ).
ProductID	varchar(9)	Ідентифікатор продукту (зовнішній ключ, посилання на таблицю Products).
ForecastDate	data	Дата, на яку зроблено прогноз.
PredictedSales	decimal(15,3)	Прогнозований обсяг продажу.
ConfidenceLevel	decimal(5,2)	Рівень довіри до прогнозу (наприклад, 95.00 для 95% довірчого інтервалу).
ModelUsed	varchar(50)	Назва або тип моделі AutoML, використаної для прогнозу.
GeneratedOn	timestamp	Дата та час генерації прогнозу.

Ця таблиця реалізована в системі наступним чином. Для зберігання даних dataset а також для зберігання прогнозних даних в конфігурації системи створено реєстр накопичення AutoML_ForecastResults.

Дані dataset у вивантажується в .csv файл обміну в наступному форматі (Рис. 2.5).



Date	Product ID	Season	Customer ID	Transit time	Quantity sold	Delivery cost	Stock level	Reorder point
07.03.2023	000013447	2	1	5	10.000		10.000	1.121
07.03.2023	000013447	2	1	5	10.000		6.300.000	1.121
05.04.2023	000013447	2	1	5	10.000		88.000	1.448
05.04.2023	000013447	2	1	5	10.000		8.400.000	1.448
05.04.2023	000013447	2	1	5	10.000		290.000	1.448
14.04.2023	000013447	2	1	5	417.000		3.985.000	1.448
21.04.2023	000013447	2	1	5	40.000		3.843.000	1.448
03.05.2023	000013447	2	1	6	20.000		5.060.000	1.786
03.05.2023	000013447	2	1	6	20.000		2.118.000	1.786
06.05.2023	000013447	2	1	5	500.000		354.000	1.786
06.05.2023	000013447	2	1	5	500.000		3.560.000	1.786
10.05.2023	000013447	2	1	6	24.000		3.274.000	1.786
10.05.2023	000013447	2	1	6	24.000		1.740.000	1.786
16.05.2023	000013447	2	1	6	2.000		1.500.000	1.786
16.05.2023	000013447	2	1	6	2.000		849.000	1.786
16.05.2023	000013447	2	1	6	10.000		188.000	1.786
25.05.2023	000013447	2	1	4	71.000		107.000	1.786
25.05.2023	000013447	2	1	4	71.000		1.800.000	1.786
25.05.2023	000013447	2	1	4	71.000		2.100.000	1.786
25.05.2023	000013447	2	1	4	71.000		60.000	1.786
25.05.2024	000013447	2	1	5	52.000		5.715.000	524
26.03.2024	000013447	2	1	5	1.000		3.771.000	524
06.04.2024	000013447	2	1	5	22.000		1.891.000	1.705
15.04.2024	000013447	2	1	5	5.000		864.000	1.705
15.04.2024	000013447	2	1	4	2.000		221.000	1.705
18.04.2024	000013447	2	1	4	2.000		2.100.000	1.705
06.05.2024	000013447	2	1	5	1.000		661.000	725
11.05.2023	000013447	2	2	4	850.000		1.350.000	1.121
14.03.2023	000013447	2	2	4	10.000		10.000	1.121
17.03.2023	000013447	2	2	4	30.000		1.890.000	1.121
20.03.2023	000013447	2	2	4	30.000		1.215.000	1.121

Рис. 2.5. Структура даних dataset до вивантаження в .csv файл обміну

Нижче (рис. 2.6) представлено створену обробку яка є частиною конфігурації Управління торговим підприємством в процесі формування dataset у ForecastResults.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	Date	Product	CustomerT	ProductID	QuantitySo	Revenue	Season	Month	SupplyID	SaleID	CustomerT	SeasonID				
1	03.10.2024	Майонез лєПряміПрода	000013171	1	33,25	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
2	04.10.2024	Майонез лєПряміПрода	000013171	3	109,8	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
3	07.10.2024	Майонез лєПряміПрода	000013171	2	65,9	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
4	08.10.2024	Майонез лєПряміПрода	000013171	2	65,9	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
5	09.10.2024	Майонез лєПряміПрода	000013171	-1	0	Осінь	01.10.2024	ГТД по имп Возврат тов	1	4						
6	10.10.2024	Майонез лєПряміПрода	000013171	2	66,5	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
7	10.10.2024	Майонез лєПряміПрода	000013171	2	65,9	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
8	10.10.2024	Майонез лєПряміПрода	000013171	-1	0	Осінь	01.10.2024	ГТД по имп Возврат тов	1	4						
9	10.10.2024	Майонез лєПряміПрода	000013171	1	33,25	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
10	10.10.2024	Майонез лєПряміПрода	000013171	2	65,9	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
11	10.10.2024	Майонез лєПряміПрода	000013171	1	36,6	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
12	10.10.2024	Майонез лєПряміПрода	000013171	2	73,2	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
13	10.10.2024	Майонез лєПряміПрода	000013171	4	146,4	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
14	03.10.2024	Майонез лєТорговіМер	000013171	12	560,4	Осінь	01.10.2024	ГТД по имп Реализации	2	4						
15	07.10.2024	Майонез лєТорговіМер	000013171	12	462,6	Осінь	01.10.2024	ГТД по имп Реализации	2	4						
16	09.10.2024	Майонез лєТорговіМер	000013171	12	560,4	Осінь	01.10.2024	ГТД по имп Реализации	2	4						
17	10.10.2024	Майонез лєТорговіМер	000013171	12	560,4	Осінь	01.10.2024	ГТД по имп Реализации	2	4						
18	14.10.2024	Майонез лєТорговіМер	000013171	60	2 313	Осінь	01.10.2024	ГТД по имп Реализации	2	4						
19	01.10.2024	Сир Гауда і ПряміПрода	000013211	1	90,75	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
20	01.10.2024	Сир Гауда і ПряміПрода	000013211	5	453,75	Осінь	01.10.2024	ГТД по имп Реализации	1	4						
21	01.10.2024	Сир Гауда і ПряміПрода	000013211	1	90,75	Осінь	01.10.2024	ГТД по имп Реализации	1	4						

Рис. 2.6. Обробка сформована програмними засобами ERP-системи підприємства

Оскільки обробка є частиною конфігурації а також у зв'язку з тим, що для роботи системи необхідно включити систему підприємства додаткові об'єкти а саме:

- довідник «Регіони» для фіксації країн походження товарів а також каналів постачання;
- реєстр (таблицю) «Товари в дорозі» для фіксації часу на доставку товарів на склади підприємства з моменту відвантаження у постачальника;
- внести зміни у стандартний документ системи ВМД по імпорту з тим щоб фіксувати рух товарів від постачальника в процесі оформлення декларації в реєстрі «Товари в дорозі»;
- додатковий реєстр AutoML_ForecastResults для накопичення історичних даних для навчання бібліотеки а також фіксації прогнозів. Це також дозволяє переглядати звіти стандартним звітом конфігурації «Універсальний звіт».

Усі зазначені об'єкти інсталиються в конфігурацію за допомогою стандартного пакету «Бібліотека стандартних підсистем» до якого вони включені і об'єднані в підсистему AutoML (Рис 2.7).

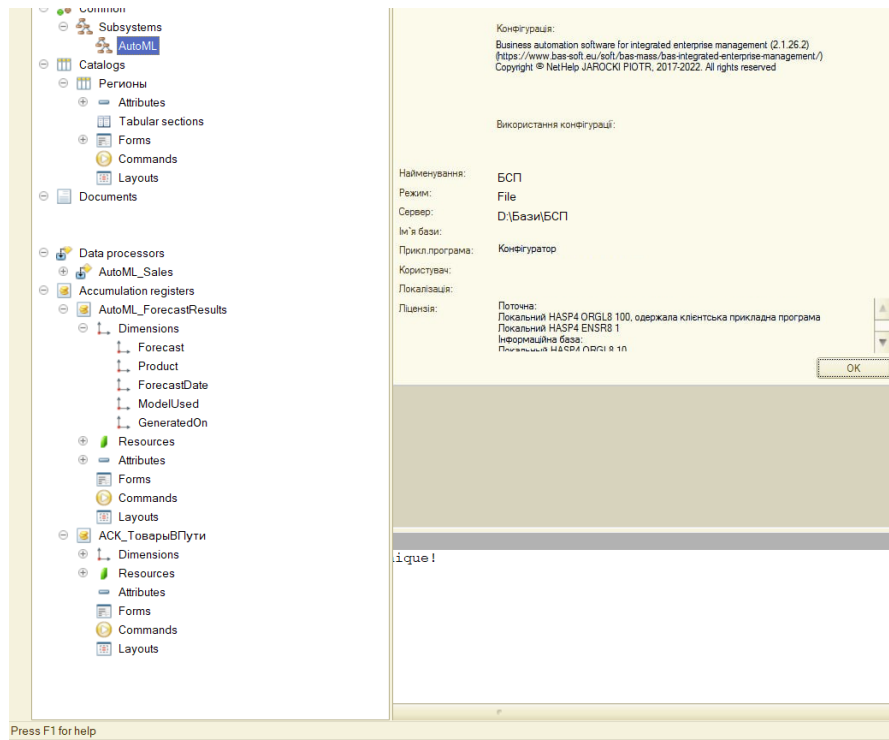


Рис. 2.7. Структура підсистеми AutoML в частині пакету Бібліотеки стандартних підсистеми

Висновки до розділу 2

Використання технології AutoML дозволяє досягти високої точності прогнозування, що важливо для таких ключових сфер бізнесу, як управління запасами та планування виробничих процесів.

Надійність і точність таких прогнозів є визначальними факторами для забезпечення ефективності операцій підприємств, адже саме від цього залежить оптимальне розподілення ресурсів і мінімізація витрат. При цьому значно зменшується обсяг ручної праці, необхідної для налаштування моделей, підвищуючи загальну продуктивність аналітичних процесів.

Інтеграція рішення з ERP-системами через API та COM-інтерфейси забезпечує достатню гнучкість і масштабованість а підприємства отримують можливість адаптувати такі технології до своїх специфічних вимог. Така конфігурація інтеграції дозволяє зберігати існуючу інформаційну інфраструктуру, мінімізуючи затрати на її перебудову, водночас підвищується функціональність та продуктивність систем.

Універсальність таких модулів також дозволяє використовувати їх у ширшому спектрі задач, таких як: адаптація модулів для аналізу продажів, оптимізації логістичних витрат; управління складськими запасами, яке є складною і багатогранною задачею, також може бути значно спрощене завдяки використанню прогнозних моделей. При цьому враховується широкий спектр факторів, що впливають на попит, і пропонувати оптимальні рішення.

РОЗДІЛ 3. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ РОБОТИ СИСТЕМИ

3.1. Тестування та результати впровадження

Тестування бібліотеки триває для оцінки її ефективності у прогнозуванні попиту та оптимізації логістичних процесів.

Основними завдання тестування є: перевірка точності прогнозів попиту; аналіз впливу впровадження на скорочення складних витрат; оцінка відповідності очікуваних і фактичних результатів.

Опишемо етапи тестування.

Підготовка тестового середовища - використовуються історичні дані з таблиці SalesHistory та InventoryLevels, при цьому формується навчальний набір даних для моделі ML.NET та вибирається тестовий період (наприклад, 3-6 місяців).

Валідація моделі - модель тестується на дані, які не використовуються в процесі навчання (hold-out test set), за такими метриками оцінки точності: середня абсолютна похибка (MAE) - середнє абсолютне відхилення між прогнозованими та фактичними значеннями; середньоквадратична помилка (RMSE) - корінь середнього квадратичного відхилення; середня абсолютна відсоткова помилка (MAPE) - середнє відсоткове відхилення.

Тестування на реальних даних - прогнози порівнюються з фактичними даними продажів та оцінюється вплив на точність прогнозів, скорочення складських витрат та ефективність управління логістикою.

Аналіз інтеграції з ERP-системою забезпечує глибоку перевірку точності та коректності обміну даними між бібліотекою машинного навчання ML.NET і внутрішньою ERP-системою підприємства. Такий підхід дозволяє забезпечити зручність інформації на всіх етапах логістичних і управлінських процесів, що є критично досягнутим для досягнення стратегічних цілей компанії. Інтеграція з урахуванням ключових технічних вимог, де особливу увагу приділено точності обробки великих обсягів

Впровадження бібліотеки машинного навчання на базі ML.NET стало значним кроком у розвитку технологічного потенціалу підприємства. Ця

бібліотека дозволила не тільки автоматизувати прогнозування ключових показників, але й запровадити новий рівень аналізу даних, що допомогло уникнути ситуацій із дефіцитом товарів або їх надлішковим накопиченням. У сучасних умовах ринкової конкуренції така оптимізація дає змогу зменшити операційні витрати, зменшуючи фінансові ризики, пов'язані з неефективним управлінням запасами.

Крім того, інтеграція ML.NET з ERP-системою позитивно вплинула на загальну ефективність логістичних процесів. Завдяки ідеальному аналізу й прогнозуванню попиту підприємство отримало можливість раціоналізувати ланцюг постачання, забезпечивши безперебійність доставки товарів. Як результат, це сприяло скороченню часу реагування на зміни, що має вирішальне значення для підтримки конкурентоспроможності.

Окремо варто відзначити вплив на клієнтське задоволення, яке суттєво покращилося за рахунок стабільності постачань. Споживачі відзначають надійність і своєчасність отримання всіх товарів, що стало можливим за допомогою інтегрованих алгоритмів прогнозування попиту та управління запасами. Ця стабільність формує довіру клієнтів і зміцнює репутацію підприємства

Таким чином, використання ML.NET як базової технології для інтеграції з ERP-системою демонструє не лише технологічну інноваційність, а й стратегічну цілеспрямованість підприємства. Упровадження таких рішень підтверджує здатність компанії адаптуватися до викликів сучасного бізнес-середовища, використовуючи передові інструменти для досягнення довгострокових цілей.

3.2. Ефективність застосування бібліотеки на підприємствах

Бібліотека машинного навчання, впроваджена для прогнозування попиту та оптимізації логістичних процесів, забезпечує значний економічний ефект і забезпечення точності прогнозів. Розглянемо ці аспекти детальніше.

Підвищення точності прогнозування стало одним із ключових результатів запровадження інноваційних технологій у нашій діяльності. До моменту інтеграції спеціалізованої бібліотеки для прогнозування ми викладалися на традиційні підходи, зокрема статичні методи, такі як ковзне середнє або експертні оцінки. Ці методи, хоча й малий певний рівень достовірності, забезпечували середню точність результатів лише на рівні близько 70%. Це створювало обмеження для ефективного прийняття рішень і вимагало пошук нових рішень для підвищення ефективності аналітичних процесів.

Із впровадженням сучасних моделей машинного навчання вдалося суттєво змінити час. Новий підхід дозволяє переходити від статичних до динамічних методів аналізу, які краще враховують змінність та складність зовнішніх і внутрішніх факторів. Як результат, значно зросла точність прогнозування, що підтверджується показниками, отриманими під час реалізації відповідних експериментів і практичного застосування моделі. Це стало потужним доказом ефективності новітніх технологій у сфері аналітики.

Позитивний ефект від запровадження таких інструментів відчутний на всіх рівнях. Завдяки підвищенню точності прогнозування стало можливим зменшити кількість помилкових рішень, оптимізувати процеси планування та ефективніше розподіляти ресурси. Крім того, більш надійні прогнози дозволили підвищити рівень довіри до аналітичних результатів з боку цікавих сторінок, що в умовах сучасної конкурентної боротьби є критичним місцем.

Таким чином, впровадження бібліотеки прогнозування стало не лише технічним оновленням, а й стратегічним кроком, що дозволив зміцнити позиції та підвищити конкурентоспроможність. Ефективність цього рішення обґрунтовується на досягненнях машинного навчання, які довели свою здатність забезпечити нову якість у прогнозуванні, недоступну традицію.

Ці підтверджені результати забезпечують та доцільність подальшого розвитку та інтеграції подібних технологій у практичну діяльність. Інновації у сфері машинного навчання відкриваються перед нами новими перспективами, дозволяючи значно підвищити точність, швидкість і гнучкість у прийнятих рішеннях, що є етапом на шляху до досягнення.

Економія витрат. Можливість скорочення складських витрат, оптимізація рівня запасів на складі дозволила знизити середній рівень очікуваних перехідних залишків на 25% , витрат на зберігання на 15-20% за рахунок поточних прогнозів.

Зменшення витрат на транспортування. Завдяки прогнозу обсягів замовлень за рахунок часу доставки скорочено витрати на термінову логістику на 10% .

Оптимізація закупівель. Завдяки точному визначенню зменшено кількість товарів, які закупаються "про запас", що знизило очікувані витрати на 5-10% (таблиця 3.1) .

Таблиця 3.1

Оцінка ефективності впровадження проекту

Критерій	До впровадження	Після впровадження
Точність прогнозів	~70%	90-93%
Середній рівень запасів	Високий через недостатню точність прогнозів	Зниження на 25%
Складські витрати	Високі через надлишкові залишки	Швидкість на 15-20%
Частота дефіциту товарів	Часті випадки дефіциту	Значне скорочення дефіциту
Витрати на термінову доставку	Високі через неправильне планування	Скорочення на 5-10%
Задоволеність клієнтів	Середній рівень через частину затримки	Підвищення завдяки стабільному постачанню

3.3. Переваги та обмеження розробленого рішення

Розроблене рішення, яке включає використання бібліотеки машинного навчання для прогнозування попиту та оптимізації логістичних процесів, має низку переваг, але також має певні обмеження, які варто отримати. Аналіз цих аспектів і рекомендацій щодо вдосконалення ефективного розкриття повного потенціалу впровадженого підходу.

Розроблене рішення з прогнозування попиту демонструє високу ефективність в оптимізації логістичних процесів підприємства. Застосування сучасних алгоритмів машинного навчання, таких як XGBoost, ARIMA та LSTM, дозволяє отримувати точні прогнози попиту на товари, що є критично важливим для ефективного управління запасами.

Оптимізація яка досягнута завдяки застосуванню системи, сприяє оптимізації ризиків дефіциту товарів та зменшенню витрат на їх зберігання, а також дозволяє значно скоротити транспортні витрати шляхом оптимізації маршрутів доставки та обсягів поставок.

Інтеграція системи з ERP-системами забезпечує автоматизацію процесів збору, обробки та аналізу даних, що підвищує оперативність прийняття рішень та знижує вплив людського фактору. Модульна структура системи дозволяє легко адаптувати її до зростання обсягів бізнесу та розширення номенклатури товарів.

Одним з ключових переваг запропонованої системи є її універсальність. Вона може бути успішно інтегрована в інформаційні системи підприємств з різним рівнем автоматизації, що мінімізує витрати на її впровадження.

Моделі машинного навчання, незважаючи на свою потужність, мають ряд обмежень, які необхідно враховувати при їх розробці та впровадженні. Одним з ключових обмежень є залежність від якості даних. Моделі потребують великих обсягів високоякісних, точних та актуальних даних для навчання. Помилки або прогалини в історичних даних можуть призвести до неточних прогнозів та ускладнити процес прийняття рішень.

Ще одним важливим аспектом є потреба в технічній експертизі. Налаштування, моніторинг та вдосконалення моделей машинного навчання вимагає значних знань та досвіду в галузі машинного навчання. Для малих та середніх підприємств залучення команди фахівців може бути складним завданням через високу вартість таких послуг.

Негативний вплив на результати роботи моделі машинного навчання, мають раптові зміни на ринку. Такі події, як економічні кризи, пандемії чи зміна регуляторного середовища, можуть суттєво вплинути на поведінку системи, а моделі, навчені на даних до цих подій, можуть не адекватно відображати нову реальність.

Нарешті, важливим фактором є значні початкові витрати, пов'язані з розробкою, впровадженням та навчанням персоналу. Ці витрати можуть бути особливо відчутними для підприємств з обмеженим бюджетом. Крім того, необхідно враховувати постійні витрати на підтримку та оновлення моделей, що також може бути значним фінансовим тягарем.

Таким чином, хоча моделі машинного навчання мають великий потенціал для вирішення складних задач, застосування їх пов'язане з рядом обмежень, які необхідно враховувати при прийнятті рішення про їх використання.

Для підвищення точності та ефективності моделі прогнозування попиту необхідно здійснити комплекс заходів, спрямованих на покращення якості даних, розширення функціональних можливостей моделі, підвищення її адаптивності та забезпечення ефективного моніторингу та інтерпретації результатів.

Одним з ключових аспектів є покращення якості даних. Регулярна перевірка та очищення даних від помилок і аномалій є необхідною умовою для навчання якісної моделі. Автоматизація процесу виправлення помилок дозволить значно скоротити час на підготовку даних і підвищити надійність результатів.

Розширення функціональних можливостей моделі передбачає інтеграцію додаткових факторів, які можуть впливати на попит. Це можуть бути як

зовнішні фактори (економічні показники, погодні умови, дії конкурентів), так і поведінкові дані клієнтів. Врахування складних змін у реальному часі за допомогою аналітики в реальному часі дозволить підвищити актуальність прогнозів. Автоматизація процесу оновлення моделі на нових даних забезпечить її постійну адаптацію до змін ринкових умов.

Для підвищення адаптивності моделі до різних типів даних та змінних умов рекомендується використовувати ансамблеві моделі, які поєднують в собі кілька алгоритмів. Це дозволить підвищити точність прогнозів та зробити модель більш стійкою до шумів у даних.

Моніторинг та інтерпретація результатів є невід'ємною частиною процесу використання моделі. Розробка системи моніторингу дозволить оцінювати якість прогнозів, виявляти відхилення та своєчасно вносити необхідні корективи. Надання бізнес-користувачам зручних інструментів для інтерпретації прогнозів (дашборди, аналітичні звіти) сприятиме ефективному використанню моделі в прийнятті рішень.

Для зменшення витрат на впровадження моделі рекомендується використовувати хмарні рішення для зберігання даних і навчання моделей. Це дозволить знизити початкові інвестиції в інфраструктуру. Інтеграція готових бібліотек та API прискорить процес розробки та впровадження моделі.

Навчання персоналу є важливим етапом впровадження моделі. Організація тренінгів для співробітників, які будуть працювати з системою, забезпечить її ефективне використання та досягнення максимальної віддачі від інвестицій.

Висновки до розділу 3

Проведене тестування продемонструвало високу точність прогнозів попиту за рахунок використання моделей машинного навчання.

Оцінка результатів показала зниження складських витрат та оптимізацію рівня запасів зі скороченням перехідних залишків, зменшення витрат на транспортування рахунок точного прогнозу обсягів замовлень і часу доставки а також скорочення витрат на закупівлі «про запас» на за рахунок оптимізації обсягів замовлень.

Успішна інтеграція бібліотеки з ERP-системою забезпечила автоматизацію обміну даними та точну передачу результатів прогнозування.

Автоматизація процесів прогнозування та інтеграції забезпечила зниження потреби в ручному втручанні та зменшення впливу людського фактору.

Простота впровадження та масштабованість системи позбавляє її використання для розширення асортименту товарів і обробки великих обсягів даних.

Стабільність і точність системи роботи підвищують конкурентоспроможність підприємства.

Ефективність системи залежить від якості даних, які використовують для навчання моделей.

Моделі важко адаптуються до різких змін ринку, таких як економічна криза або надзвичайний результат тому важливим є впровадження автоматичного донавчання моделей для забезпечення актуальності прогнозів.

Впровадження бібліотеки машинного навчання на базі ML.NET дозволило створити умови для зниження витрат та оптимізації логістичних процесів.

ВИСНОВКИ

Автоматизоване машинне навчання (AutoML) стає інструментом, який відкриває безпрецедентні можливості для оптимізації логістичних процесів українських підприємств. Його застосування дозволяє не тільки підвищити ефективність управління ланцюгами постачання, але й істотно знизити витрати, водночас підвищуючи рівень задоволення клієнтів. Це відбувається за допомогою алгоритмів здатності AutoML обробляти величезні обсяги даних, знаходячись у них із закритими закономірностями, які раніше могли залишитися поза вартістю. У результаті підприємства зможуть будувати більше точні прогнози, автоматизувати процеси та значно підвищити якість своїх послуг.

Впровадження AutoML в логістиці створює новий рівень ефективності при прогнозуванні попиту на продукцію. Використовуючи сучасні алгоритми, такі як XGBoost, ARIMA та LSTM, підприємства можуть точно оцінювати майбутні потреби більшого ринку. Це не лише уникнути дефіциту товарів або недостатніх запасів, а й забезпечити суттєву економію ресурсів. Здатність аналізувати попередні дані в поєднанні з урахуванням зовнішніх факторів дозволяє українським компаніям краще адаптуватися до змінної ринкової ситуації, що має критичне значення для конкурентоспроможності.

Інтеграція AutoML в логістичні процеси також відкриває перспективи для оптимізації маршрутів доставки. Завдяки цьому підприємству можна скоротити транспортні витрати, зменшити витрати пального та знизити екологічний слід, одночасно забезпечуючи швидку доставку продукції. Такий підхід дозволяє не лише підвищити ефективність роботи компаній, але й відповідати сучасним екологічним стандартам, які дедалі частіше залишаються фактором для споживачів.

Однак впровадження AutoML не обходиться без викликів. Крім того, якість даних, які використовують для навчання моделей, є вирішальним фактором. Неточні або неповні дані можуть привести до помилкових висновків і, як внаслідок, до неефективних рішень. Тому важливо впроваджувати системи регулярних перевірок, очищення та оновлення даних, які використовуються в

роботі алгоритмів машинного навчання. Крім того, адаптація моделей до змінних ринкових умов вимагає моніторингу їх ефективності регулярного та відповідного оновлення.

Ще одним аспектом є підготовка персоналу. AutoML хоча й пропонує інструменти для автоматизації, проте його впровадження вимагає участі спеціалістів, які розуміють основи роботи алгоритмів машинного навчання та здатні адаптувати їх до потреб конкретного підприємства. Для малих і середніх компаній це може стати серйозним викликом, більшість залучень експертів потребує значних ресурсів.

Звичайно, успіх у використанні AutoML залежить від готовності компаній інвестувати в розвиток компетенцій своїх співробітників. Навчання персоналу роботі з новими технологіями, інтеграція цих рішень у наявні ERP-системи, а також автоматизація процесів оновлення моделей є основними кроками для забезпечення ефективності таких впроваджень. Лише системний підхід дозволяє розкрити весь потенціал AutoML у логістичних процесах.

Враховуючи все вищезазначене, AutoML виступає потужним інструментом трансформації для українського бізнесу. Його застосування дозволяє компаніям досягти нових рівнів продуктивності, забезпечити адаптацію до змін ринку та отримати конкурентні переваги. Проте для цього необхідно викликати виклики, пов'язані з якими даними, підготовкою персоналу та адаптацією моделей. Лише за умов відповідного підходу до впровадження AutoML підприємство дозволить отримати максимальну користь від цієї технології, яка вже сьогодні змінює правила гри у світовій логістиці.

СПИСОК ЛІТЕРАТУРИ

1. Braun M., Krebs S., Flohr F., Gavrilu D. EuroCity Persons: A Novel Benchmark for Person Detection in Traffic Scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2019. Vol. 41, No 8. 1844 P.
2. C. Shan, S. Gong, and P. W. McOwan, Facial expression recognition based on local binary patterns: A comprehensive study, Image and Vision Computing, 2019. P. 803 – 816
3. Sanoar Hossain, Saiyed Umer, Ranjeet Kr. Rout, and M. Tanveer. 2023. Fine-grained image analysis for facial expression recognition using deep convolutional neural networks with bilinear pooling. Appl. Soft Comput. 134, C (Feb 2023). <https://doi.org/10.1016/j.asoc.2023.109997>
4. Howard A., Sandler M., Chu G. Searching for MobileNetV3. IEEE/CVF International Conference on Computer Vision : conference materials, Seoul, 27 Oct.-2 Nov. 2019. Seoul, 2019. 1314 P.
5. I. Swathi, S.S. Gantayat. Facial Expression Recognition Based On Emotion For Visually Challenged People. ICICSS, 2018. P. 123-134.
6. Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 2020, P. 94-101
7. Rainfall–runoff modelling using Long ShortTerm Memory (LSTM) networks / Kratzert F., Klotz D., Brenner C., Schulz K., Herrnegger M. Hydrology and Earth System Sciences. 2018. Vol. 22, № 11. P. 6005-6022.
8. Si Y. Using ARIMA model to analyse and predict bitcoin price / Yang Si. // DCP Business & Management. – 2022. – №34. – C. 1210–1216.
9. Stegmann M.B. Analysis and Segmentation of Face Images using Point Annotations and Linear Subspace Techniques. (Technical report IMM-REP-2002-22, Informatics and Mathematical Modelling, Technical University of Denmark, 2022).

10. Tereikovska, L. Neural network emotion recognition method based on facial images. Computer-integrated technologies: education, science, production, (40), 2020. С. 146-152. <https://doi.org/10.36910/6775-2524-0560-2020-40-22>
11. What is ACID in databases [Электронный ресурс] // Ian, Database.guide. – Режим доступа до ресурсу: <https://database.guide/what-is-acid-in-databases/>.
12. What is Blockchain Technology? How Does Blockchain Work? [Электронный ресурс] // SimplLearn. – 2023. – Режим доступа до ресурсу: <https://www.simplilearn.com/tutorials/blockchain-tutorial/blockchain-technology>.
13. What is OLAP (online analytical processing)? [Электронный ресурс] // Biscobing – Режим доступа до ресурсу: <https://www.techtarget.com/searchdatamanagement/definition/OLAP>.
14. What is OLTP? [Электронный ресурс] // IBM – Режим доступа до ресурсу: <https://www.ibm.com/topics/oltp>.
15. What is random forest? [Электронный ресурс] // IBM – Режим доступа до ресурсу: <https://www.ibm.com/topics/random-forest>.
16. Wu J. Short term return prediction of cryptocurrency based on XGBoost algorithm / J. Wu, X. Guo. // International Conference on Big Data, Information and Computer Network (BDICN). – 2022. – С. 39–42.
17. Y. Chen, H. Wu, T. Wang, Y. Wang. A Comparison of Methods of Facial Expression Recognition. Proceedings of the 1st WRC Symposium on Advanced Robotics and Automation 2018 Beijing, 2018. P. 261-262
18. Yao, Yifan & Wang, Lina. (2021). Combination of window-sliding and prediction range method based on LSTM model for predicting cryptocurrency. 10.48550/arXiv.2102.05448.
19. Yu F., Chen H., Wang X. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition : conference materials, Seattle, 13-19 June 2020. Seattle, 2020. 263P.

ДОДАТКИ

Код бібліотеки.

```

Model1.mbcconfig*  NuGet: MLNetCOMWrapper  Class1.cs*  MLNetCOMWrapper  packages.config
MLNetCOMWrapper  SalesOptimizationLibrary.SalesOptimizer  _databaseConnectionString

1  using System;
2  using System.Collections.Generic;
3  using System.Data;
4  using System.IO;
5  using CsvHelper;
6  using CsvHelper.Configuration;
7  using Microsoft.ML;
8  using Microsoft.ML.AutoML;
9  using Microsoft.ML.Data;
10 using System.Runtime.InteropServices;
11
12 namespace SalesOptimizationLibrary
13 {
14     /// <summary>
15     /// Цей клас реалізує обгортку для бібліотеки ML.NET, що дозволяє тренувати моделі, прогнозувати дані та зберігати результати.
16     /// </summary>
17     [ConVisible(true)]
18     public class SalesOptimizer
19     {
20         private readonly string _databaseConnectionString;
21         private readonly MLContext _mlContext;
22
23         /// <summary>
24         /// Конструктор класу. Ініціалізує підключення до бази даних та створює екземпляр MLContext для роботи з ML.NET.
25         /// </summary>
26         /// <param name="databaseConnectionString">Рядок підключення до бази даних.</param>
27         public SalesOptimizer(string databaseConnectionString)
28         {
29             _databaseConnectionString = databaseConnectionString;
30             _mlContext = new MLContext();
31         }
32
33         /// <summary>
34         /// Ця функція відповідає за тренування моделі на основі історичних даних і прогнозування продажів із записом результатів у CSV.
35         /// </summary>
36         /// <param name="historicalDataCsvPath">Шлях до файлу CSV з історичними даними.</param>
37         /// <param name="forecastResultsCsvPath">Шлях до файлу CSV для збереження результатів прогнозування.</param>
38         public void TrainAndPredict(string historicalDataCsvPath, string forecastResultsCsvPath)
39         {
40             // Завантаження історичних даних
41             var trainingData = LoadCsvData<HistoricalData>(historicalDataCsvPath);
42
43             // Тренування моделі
44             var experiment = _mlContext.Auto().CreateRegressionExperiment(maxExperimentTimeInSeconds: 60);
45
46             var data = _mlContext.Data.LoadFromEnumerable(trainingData);
47             var split = _mlContext.Data.TrainTestSplit(data, testFraction: 0.2);
48
49             var result = experiment.Execute(split.TrainSet, labelColumnName: "SalesVolume");
50
51             // Оцінка моделі
52             var metrics = _mlContext.Reggression.Evaluate(result.BestRun.Model.Transform(split.TestSet));
53             SaveModelPerformance(result.BestRun.TrainerName, metrics);
54
55             // Прогноз майбутніх продажів
56             var forecastResults = PredictSales(result.BestRun.Model, trainingData);
57
58             // Збереження результатів прогнозування у CSV
59             SaveCsvData(forecastResults, forecastResultsCsvPath);
60         }
61
62         /// <summary>
63         /// Функція зчитує дані з CSV-файлу і перетворює їх у список об'єктів заданого типу.
64         /// </summary>
65         /// <typeparam name="T">Тип даних, які зберігаються у CSV.</typeparam>
66         /// <param name="filePath">Шлях до CSV-файлу.</param>
67         /// <returns>Список об'єктів типу T.</returns>
68         private List<T> LoadCsvData<T>(string filePath) where T : class
69         {
70             var reader = new StreamReader(filePath);
71             var csv = new CsvReader(reader, new CsvConfiguration(System.Globalization.CultureInfo.InvariantCulture));
72             return new List<T>(csv.GetRecords<T>());
73         }
74
75         /// <summary>
76         /// Функція записує дані у форматі CSV у заданий файл.
77         /// </summary>
78         /// <typeparam name="T">Тип даних, які потрібно записати.</typeparam>
79         /// <param name="data">Дані для запису у CSV.</param>
80         /// <param name="filePath">Шлях до CSV-файлу.</param>
81         private void SaveCsvData<T>(IEnumerable<T> data, string filePath) where T : class
82         {
83             var writer = new StreamWriter(filePath);
84             var csv = new CsvWriter(writer, new CsvConfiguration(System.Globalization.CultureInfo.InvariantCulture));
85             csv.WriteRecords(data);
86         }
87
88         /// <summary>
89         /// Ця функція прогнозує продажі на основі тренованої моделі та історичних даних.
90         /// </summary>
91         /// <param name="model">Тренована модель.</param>
92         /// <param name="historicalData">Історичні дані для прогнозування.</param>
93         /// <returns>Список результатів прогнозування.</returns>
94         private IEnumerable<ForecastResult> PredictSales(ITransformer model, IEnumerable<HistoricalData> historicalData)
95         {
96             var forecastResults = new List<ForecastResult>();
97
98             foreach (var record in historicalData)
99             {
100                 var predictionEngine = _mlContext.Model.CreatePredictionEngine<HistoricalData, SalesPrediction>(model);
101                 var prediction = predictionEngine.Predict(record);
102
103                 forecastResults.Add(new ForecastResult
104                 {
105                     ProductID = record.ProductID,
106                     ForecastDate = DateTime.Now.AddDays(1),
107                     PredictedSales = prediction.PredictedSales,
108                     ConfidenceLevel = 95.00%, // Приклад довільного рівня
109                     ModelUsed = model.ToString(),
110                     GeneratedOn = DateTime.Now
111                 });
112             }
113
114             return forecastResults;
115         }
116     }
117 }

```

```

116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180

```

```

/// <summary>
/// Функція зберігає метрики ефективності моделі в базі даних для подальшого аналізу.
/// </summary>
/// <param name="modelName">Назва моделі.</param>
/// <param name="metrics">Метрики ефективності моделі.</param>
private void SaveModelPerformance(string modelName, RegressionMetrics metrics)
{
    var connection = new System.Data.SqlClient.SqlConnection(_databaseConnectionString);
    connection.Open();

    var command = connection.CreateCommand();
    command.CommandText = @"INSERT INTO ModelPerformance (modelName, TrainingDate, MAE, MSE, RMSE, MAPE)
VALUES (@modelName, @TrainingDate, @MAE, @MSE, @RMSE, @MAPE)";

    command.Parameters.AddWithValue("@modelName", modelName);
    command.Parameters.AddWithValue("@TrainingDate", DateTime.Now);
    command.Parameters.AddWithValue("@MAE", metrics.MeanAbsoluteError);
    command.Parameters.AddWithValue("@MSE", metrics.MeanSquaredError);
    command.Parameters.AddWithValue("@RMSE", metrics.RootMeanSquaredError);
    //command.Parameters.AddWithValue("@MAPE", metrics.MeanAbsolutePercentageError);

    command.ExecuteNonQuery();
}

/// <summary>
/// Цей клас представляє структуру історичних даних, які використовуються для тренування моделі.
/// </summary>
[Serializable]
public class HistoricalData
{
    public int RecordID { get; set; }
    public int ProductID { get; set; }
    public DateTime Date { get; set; }
    public decimal SalesVolume { get; set; }
    public decimal Price { get; set; }
    public string Promotion { get; set; }
    public int StockLevel { get; set; }
    public string Weather { get; set; }
    public string DayOfWeek { get; set; }
    public bool Holiday { get; set; }
}

/// <summary>
/// Цей клас містить структуру для збереження результатів прогнозування.
/// </summary>
[Serializable]
public class ForecastResult
{
    public int ProductID { get; set; }
    public DateTime ForecastDate { get; set; }
    public decimal PredictedSales { get; set; }
    public decimal ConfidenceLevel { get; set; }
    public string ModelUsed { get; set; }
    public DateTime GeneratedOn { get; set; }
}

/// <summary>
/// Цей клас використовується для представлення прогнозованих результатів моделі.
/// </summary>
public class SalesPrediction
{
    [ColumnName("Score")]
    public float PredictedSales { get; set; }
}

```