НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

«На правах рукопису»

УДК _____

«До захисту допущено»

Завідувач кафедри
_____ Сергій СТІРЕНКО
(підпис) (ім'я, прізвище)
"____" _____ 2021 р.

Магістерська дисертація

зі спеціальності: *123. Комп'ютерна інженерія*
(код та назва напряму підготовки або спеціальності)

Спеціалізація: *123. Комп'ютерні системи та мережі*

на тему: Прогнозування часових рядів з використанням методів машинного навчання

Виконала: студентка VI курсу, групи ІО-01мп
(шифр групи)

Бровченко Анастасія Вікторівна
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник проф., д.т.н., проф. Стіренко С.Г.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант з нормоконтролю проф., д.т.н., проф. Кулаков Ю.О.
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент доц. каф. ІСТ, к.т.н., Писаренко А.В.
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студентка _____
(підпис)

Київ – 2021 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет (інститут) Інформатики та обчислювальної техніки
(повна назва)

Кафедра Обчислювальної техніки
(повна назва)

Рівень вищої освіти – другий (магістерський) за освітньо-професійною програмою

Спеціальність 123. Комп'ютерна інженерія
(код і назва)

Спеціалізація 123. Комп'ютерні системи та мережі
(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Сергій СТІРЕНКО
(підпис) (ініціали, прізвище)
« » 2021 р.

ЗАВДАННЯ
на магістерську дисертацію студентці

Бровченко Анастасії Вікторівні
(прізвище, ім'я, по батькові)

1. Тема дисертації Прогнозування часових рядів з використанням методів машинного навчання

Науковий керівник дисертації Стіренко Сергій Григорович, проф., д.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від « 25» жовтня 2021 р. № 3587-с

2. Строк подання студентом дисертації 30 листопада 2021р

3. Об'єкт дослідження методи прогнозування часових рядів.

4. Предмет дослідження розробка методу прогнозування часових рядів, що базується на використанні алгоритмів машинного навчання, для підвищення точності прогнозів.

5. Перелік завдань, які потрібно розробити: дослідити існуючі методи прогнозування часових рядів, розробити гібридний метод для прогнозування часових рядів, виконати тестування розробленого методу та проаналізувати отримані результати

6. Консультанти розділів дисертації:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	проф., д.т.н. Кулаков Ю.О.		

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів дисертації	Примітка
1	Вивчення літератури	21.09.2021	
2	Складання і узгодження технічного завдання	31.09.2021	
3	Написання вступної частини та огляд рішень	18.10.2021	
4	Моделювання розробленого способу	15.11.2021	
5	Оформлення документації ДП	20.11.2021	
6	Попередній захист та проходження нормативного контролю	10.12.2021	
7	Подання ДП рецензенту		

Студентка _____ Анастасія БРОВЧЕНКО
(підпис) (ініціали, прізвище)

Науковий керівник дисертації _____ Сергій СТІРЕНКО
(підпис) (ініціали, прізвище)

РЕФЕРАТ

на магістерську дисертацію

виконану на тему: Прогнозування часових рядів з використанням методів машинного навчання

студенткою: Бровченко Анастасією Вікторівною

Робота складається із вступу та чотирьох розділів. Загальний обсяг роботи: 70 аркушів основного тексту, 17 ілюстрацій, 30 таблиць. При підготовці використовувалася література з 26 різних джерел.

Актуальність. Потреба в прогнозуванні часових рядів виникає практично в усіх сферах нашого життя від економічних показників до погодних процесів, від соціальних до медичних вимірів. Компанії обробляють величезні об'єми даних і очікується, що їх обсяг буде зростати в геометричній прогресії.

Традиційні методи машинного навчання часто не дають потрібної точності прогнозів, тому альтернативою є поєднання кількох методів прогнозування в один, з метою отримати в кінцевій моделі переваги від усіх поєднаних методів. Ідея поєднання кількох моделей не є новою. Щороку з'являється багато наукових праць, які досліджують різні способи поєднання та комбінації моделей, проте остаточних інструкцій які моделі обрати та як їх поєднати при вирішенні конкретної задачі немає. Тому дослідження цієї теми є актуальною задачею.

Мета і завдання дослідження. Метою магістерської роботи є покращення точності прогнозування часового ряду з використанням методів машинного навчання.

Для досягнення мети дослідження поставлено і вирішено такі завдання:

- розглянути існуючі методи прогнозування часових рядів;
- розглянути методи поєднання різних алгоритмів для прогнозування часових рядів;
- розробити гібридний метод прогнозування часового ряду на основі поєднання різних методів;
- за допомогою розробленого методу отримати прогнози для заданого часового ряду об'ємів продажів товару;

- дати інтерпретацію результатам проведених експериментів, порівняти розроблений алгоритм з стандартними методами щодо точності та стабільності прогнозу.

Об’єкт дослідження – методи прогнозування часових рядів.

Предмет дослідження – розробка методу прогнозування часових рядів, що базується на використанні алгоритмів машинного навчання, для підвищення точності прогнозів.

Методи досліджень. Для досягнення поставлених в магістерській роботі задач, використано методи попередньої обробки даних для їх подальшого використання в алгоритмах машинного навчання, методи машинного навчання та методи поєднання алгоритмів машинного навчання.

Наукова новизна одержаних результатів. В результаті виконання магістерської роботи було запропоновано гібридний метод для прогнозування часових рядів, який полягає у стековому поєднанні чотирьох алгоритмів – RandomForest, CatBoost, LGBM, RidgeRegressor – з метарегресором RidgeRegressor, розроблено програмний продукт для прогнозування часових рядів з використанням запропонованого методу та проведено його тестування на даних продажів товару в магазині. Запропонований метод направлений на підвищення точності прогнозування часових рядів завдяки поєднанню гетерогенних моделей на першому рівні стеку. Проведене дослідження показало, що модель дозволяє отримувати більш точні прогнозні оцінки об’ємів продажів, ніж стандартні методи.

Особистий внесок здобувача. Магістерське дослідження є самостійно виконаною роботою, в якій відображено особистий авторський підхід та особисто отримані теоретичні та прикладні результати, що відносяться до вирішення задачі прогнозування часових рядів.

Практична цінність. Отримані результати можуть використовуватися у майбутніх дослідженнях за напрямками:

- вдосконалення методів прогнозування часових рядів;
- аналіз ефективності поєднання запропонованих ШНМ;
- розробка узагальненого алгоритму для вибору методу прогнозування часового ряду в залежності від типу ряду та поставленої задачі.

Ключові слова

Часові ряди, прогнозування часових рядів, машинне навчання, штучні нейронні мережі.

ABSTRACT

for a master's thesis

performed on the topic: Time Series Forecasting Using Machine Learning Methods

student: Brovchenko Anastasiia

The work consists of an introduction and four chapters. Total amount of work: 70 sheets of the main text, 17 illustrations, 30 tables. Literature from 26 different sources was used in the preparation.

Topicality. The need to predict time series arises in almost all areas of our lives from economics to weather processes, from social to medical dimensions. Companies process huge amounts of data and it is expected that their volume will grow exponentially.

Traditional machine learning methods often do not provide the required accuracy of predictions, so the alternative is to combine several forecasting methods into one, in order to obtain the benefits of all combined methods in the final model. The idea of combining several models is not new. There are many scientific papers each year that explore different ways to combine models, but there are no clear instructions on which models to choose and how to combine them to solve a particular problem. Therefore, the study of this topic is an urgent task.

The purpose and objectives of the study. The aim of the master's thesis is to improve the accuracy of time series forecasting using machine learning methods.

To achieve the goal of the study the following tasks were set and solved:

- to consider existing methods of time series forecasting;
- to consider methods of combining different algorithms for time series forecasting;
- to develop a hybrid method of the time series forecasting based on a combination of different methods;
- to obtain forecasts for a given time series of sales of goods using the developed method;
- to interpret the results of experiments, to compare the developed algorithm with standard methods for accuracy and stability of the forecast.

The object of research is time series forecasting methods.

The subject of research is the development of a time series forecasting method, based on the usage of machine learning algorithms, to improve the accuracy of forecasts.

Research methods. In the master's thesis were applied data preprocessing methods, machine learning methods and methods of combining machine learning algorithms to achieve the set objectives.

Scientific novelty of the obtained results. As a result of the master's thesis, was proposed a hybrid method for time series prediction, which consists of a stack combination of four algorithms - RandomForest, CatBoost, LGBM, RidgeRegressor – with metaregressor RidgeRegressor; a software for time series forecasting using the proposed method was developed and tested on the data of sales of goods in the store. The proposed method is aimed at improving the accuracy of time series prediction by combining heterogeneous models at the first level of the stack. The study showed that the proposed model allows to obtain more accurate estimates of sales than standard methods.

Personal contribution of the applicant. The master's research is an independently performed work, which reflects the personal author's approach and personally obtained theoretical and applied results related to solving the problem of time series forecasting.

Practical value. The obtained results can be used in future research in the following areas:

- improvement of time series forecasting methods
- analysis of the effectiveness of the combination of the proposed ANN;
- development of a generalized algorithm for choosing the method of forecasting the time series depending on the type of series and the task.

Keywords

Time series, time series forecasting, machine learning, artificial neural networks.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AR – Autoregression, Авторегресія.

MA – Moving Average, Метод ковзного середнього.

SES – Simple Exponential Smoothing, просте експоненційне згладжування.

ARMA – Autoregressive Moving Average, авторегресія ковзного середнього.

ARIMA – Autoregressive Integrated Moving Average, авторегресія інтегрованого ковзного середнього або модель Бокса-Дженкінса.

HWES – Holt Winter's Exponential Smoothing, метод експоненційного згладжування Хольта-Вінтерса.

MLP – Multi-Layer Perceptron, багатошаровий персептрон.

RNN – Recurrent Neural Network, рекурентна нейронна мережа.

GRU – Gated Recurrent Units.

BNN – Bayesian Neural Network, байєсівська нейронна мережа.

LSTM – Long Short-Term Memory.

RF – Random Forest, випадковий ліс.

ЦА – цільова аудиторія.

ВСТУП

Прогнозування часових рядів було і залишається важливою областю дослідження. На сьогоднішній день виникає потреба в прогнозуванні практично в усіх сферах нашого життя:

- Ціни на продукти
- Доходи бізнесу
- Продажі, попит на певні товари
- Попит на поїздки в транспорті
- Курси валют, акцій
- Прогнозування погодних умов
- Прогнозування поширення COVID-19

Найважливіше завдання прогнозування явищ і процесів – виявлення закономірностей і встановлення основних тенденцій розвитку. Моделювання та прогнозування дозволяють управляти явищами, процесами та передбачити їх подальший розвиток.

Однією з головних цілей цієї дослідницької роботи є пошук надійного механізму прогнозування тенденцій продажів, який реалізується за допомогою методів машинного навчання для досягнення найкращого можливого доходу. Прогнозування продажів – це важливий етап розробки стратегії компанії, який визначить подальші позиції на ринку. Наприклад, у ході прогнозування стане ясно, що компанії вигідніше виходити на нові ринки, а не працювати з іншою цільовою аудиторією на освоєних. Звичайно, така інформація істотно впливає на розвиток фірми.

Сучасний бізнес обробляє величезні об’єми даних. Очікується, що обсяг даних буде зростати в геометричній прогресії. Тому існує гостра потреба нових методів аналізу даних та інтелектуальних моделей прогнозування тенденцій продажів з максимально можливим рівнем точності та надійності. Прогнозування продажів дає уявлення про те, як компанія повинна керувати своєю робочою силою, грошовими потоками та ресурсами. Це важлива передумова для

планування та прийняття рішень. Це дозволяє компаніям ефективно планувати свої бізнес-стратегії.

Точні прогнози дозволяють організації покращити зростання ринку з більш високим рівнем отримання доходу. Інтелектуальні технології дуже ефективні в перетворенні величезного обсягу даних на корисну інформацію для прогнозування витрат і продажів, це основа правильного бюджетування. На організаційному рівні прогнози збуту є важливими вхідними матеріалами для багатьох заходів щодо прийняття рішень у різних функціональних сферах, таких як операції, маркетинг, продажі, виробництво та фінанси.

РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ДЛЯ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

В цьому розділі розглянуто існуючі методи для прогнозування часових рядів, як класичні, так і методи машинного навчання. Виконано аналіз переваг та недоліків кожного методу.

1.1. Визначення і характеристики часового ряду

Будь-які дані, записані з деяким фіксованим інтервалом часу, називаються часовим рядом. У часових рядах порядок точок даних має важливе значення. Цей фіксований інтервал може бути погодинним, щоденним, місячним або щорічним. Наприклад погодинне зчитування вологості, щоденна зміна цін на паливо, щомісячний рахунок за електроенергію, щорічний звіту про прибуток компанії тощо. У даних часових рядів час завжди буде незалежною змінною, і може бути одна або багато залежних змінних. Якщо існує лише одна змінна, що залежить від часу, то дані називають одновимірним часовим рядом. Якщо існує кілька залежних змінних, то багатовимірним часовим рядом.

Мета аналізу часових рядів - зрозуміти, як зміна часу впливає на залежні змінні, і відповідно передбачити значення для майбутніх часових інтервалів. Першим кроком при аналізі часового ряду для розробки прогностичної моделі є виявлення і розуміння закономірностей, що лежать в основі часового ряду.

Trend (Тенденція, тренд) - довгострокова поступова зміна ряду. Якщо тенденція зростаюча, залежна змінна буде збільшуватися з часом і навпаки.

Seasonality (Сезонність) - передбачувані, короткострокові зміни даних, які виникають через певні фіксовані проміжки часу і повторюються нескінченно. Сезонні зміни даних можуть статися через природні чи техногенні події. Наприклад, щороку продажі теплої тканини збільшуються безпосередньо перед зимовим сезоном.

Cyclicity (Циклічна складова) - Довгострокові коливання даних, на які можуть піти роки або десятиліття. Такі коливання відбуваються непередбачувано і часто є результатом зовнішніх економічних умов.

Noise / Irregularities (Помилка, шум) - випадкові коливання, що виникають внаслідок неконтрольованих обставин, таких як землетруси, війни, повені, пандемія тощо. Наприклад, через пандемію COVID-19 існує величезний попит на дезінфікуючі засоби та маски для рук.

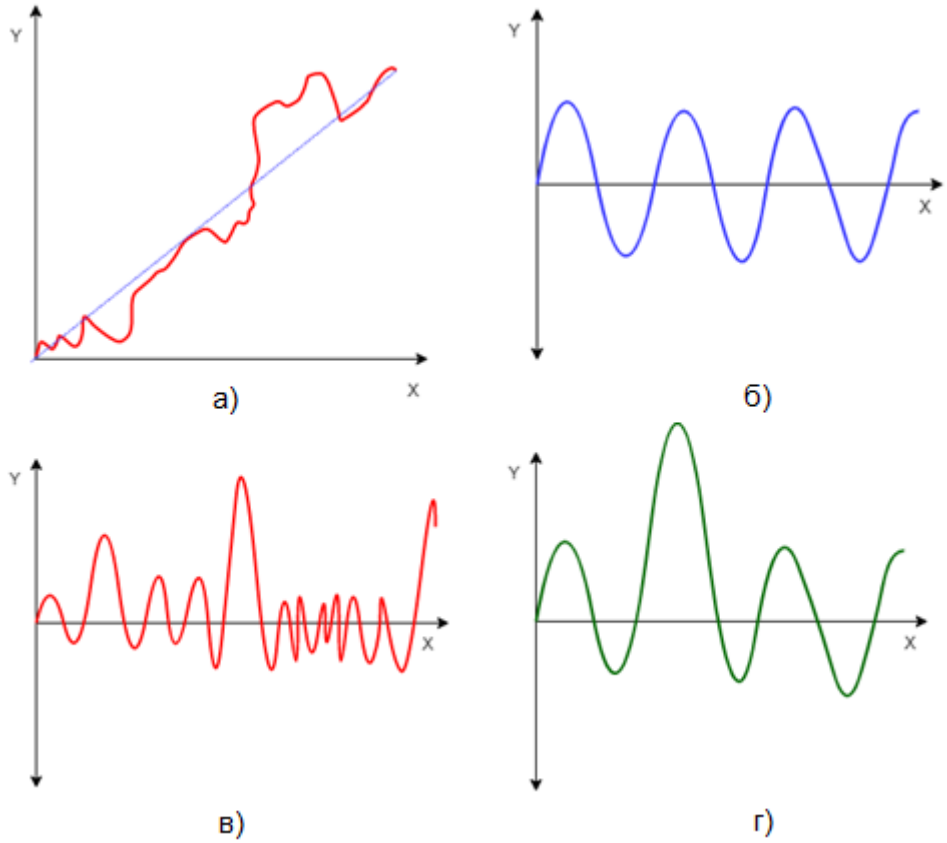

Рис 1.1 – Характеристики часового ряду:
а) тренд, б) циклічність, в) сезонність, г) шум

Дані часових рядів, які мають зазначені вище характеристики, називаються "нестаціонарними даними". Більшість реальних даних, з якими ми стикаємося є саме нестаціонарними [1].

1.2. Класичні методи прогнозування часових рядів

На практиці обирається модель до заданого часового ряду і її відповідні параметри оцінюються за допомогою відомих значень ряду. Перш ніж обрати модель, виконують аналіз часового ряду. Він містить методи, які намагаються зрозуміти природу поведінки ряду і часто корисні для майбутнього прогнозування та моделювання.

Під час прогнозування часових рядів попередні спостереження збираються та аналізуються для розробки відповідної математичної моделі, яка фіксує основні процеси генерування даних для ряду. Далі майбутні події прогнозуються за допомогою моделі. Цей підхід є особливо корисним, коли недостатньо знань про статистичну схему, за якою слідують послідовні спостереження, або коли бракує задовільної пояснювальної моделі.

Прогнозування часових рядів має важливе застосування в різних областях. Часто цінні стратегічні рішення та запобіжні заходи приймаються на основі результатів прогнозу. Таким чином, зробити хороший прогноз, тобто обрати і налаштувати відповідну модель до часового ряду, є досить важливим. За останні кілька десятиліть дослідники доклали багатьох зусиль для розробки та вдосконалення відповідних моделей прогнозування часових рядів.

1.2.1. Naïve model (Наївний прогноз)

В наївному прогнозі прогнозоване значення просто дорівнює значенню останнього спостереження. Цей найпростіший метод часто використовується як еталон для оцінки роботи складніших прогнозів. Іншими словами, якщо ваша складна модель менш точна, ніж наївний прогноз, то ви, швидше за все, робите щось не так.

Сезонний наївний метод схожий на наївний прогноз за винятком того, що прогнозоване значення дорівнює значенню останнього спостереження з того ж сезону. Наприклад, в місячному масштабі при використанні цього методу прогноз листопада буде дорівнює останньому спостереженню в листопаді. Доцільно використовувати для рядів з сезонністю.

1.2.2. Autoregression Model

У моделі лінійної регресії ми прогнозуємо змінну, що нас цікавить, використовуючи лінійну комбінацію попередніх значень цієї змінної. Термін авторегресія вказує, що це регресія змінної проти неї самої. Розраховується за формулою:

$$x_{t+1} = k_0 + k_1 x_t + k_2 x_{t-1} + k_3 x_{t-2} + \ldots + k_n x_{t-n} ,$$

де k_0 ... k_n - коефіцієнти, знайдені шляхом оптимізації моделі на тренувальних даних, а x_t ... x_{t-n} – вхідні значення [2].

1.2.3. Moving Average (MA, Метод ковзного середнього)

Модель ковзного середнього (MA) є поширеним підходом для моделювання одновимірних часових рядів. Прогнозоване значення дорівнює середньому значенню визначеної кількості періодів часу. Ми можемо не враховувати всі історичні дані, а використовувати лише останні.

$$x_{t+1} = \frac{(x_t + x_{t-1} + x_{t-2} + \cdots + x_{t-n})}{n}.$$

Також можна в цю модель можна ввести ваги для вхідних значень, щоб, наприклад, надати більшу вагу останнім значенням.

$$x_{t+1} = k_0 + \frac{(k_1 x_t + k_2 x_{t-1} + k_3 x_{t-2} + \cdots + k_n x_{t-n})}{n},$$

де k_0 ... k_n – коефіцієнти (ваги) , а x_t ... x_{t-n} – вхідні значення.

Цей метод доволі простий, а тому і поширений, для прогнозування стохастичних часових рядів, проте для нестохастичних рядів його використовувати недоцільно. Наприклад якщо ряд має зростаючий чи спадаючий тренд, усереднюючи його значення, для прогнозу ми отримаємо велику похибку. Також отримаємо велику похибку, якщо дані зашумлені [2].

1.2.4. Autoregressive Moving Average (ARMA)

Поєднує в собі дві моделі – авторегресії (AR) та ковзного середнього з коефіцієнтами (MA). При цьому зазвичай кількість членів авторегресії позначають p замість n, а кількість членів ковзного середнього – q. Коефіцієнти авторегресії позначають φ, а ковзного середнього – θ; c – стала. Тоді p називають числом спостережень відставання, включених у модель або порядком відставання. q називають розміром вікна ковзної середньої або порядком ковзної середньої.

Загальна формула виглядає наступним чином:

$$x_{t+1} = c + \sum_{i=0}^{p} \varphi_i x_{t-i} + \sum_{i=0}^{q} \theta_i x_{t-i},$$

Такий ряд позначають ARMA(p, q).

Є також інші варіації моделі ARMA, найпопулярніша: Autoregressive Integrated Moving Average (ARIMA або модель Бокса-Дженкінса)

Бокс і Дженкінс запропонували виділити клас нестаціонарних рядів, які можна звести до стаціонарного ряду типу ARMA взяттям послідовних різниць. Якщо ряд після взяття d послідовних різниць зводиться до стаціонарного, то для його прогнозування можна застосувати комбіновану модель авторегресії і ковзного середнього, що позначається як ARIMA(p, d, q) .

Першу різницю ряду знаходять наступним чином:

$$\Delta^1 = x_t - x_{t-1} .$$

Другу різницю:

$$\Delta^2 = \Delta^1_t - \Delta^1_{t-1} .$$

Загальна формула виглядає наступним чином:

$$x_{t+1} = c + \sum_{i=0}^{p} \varphi_i \Delta^d x_{t-i} + \sum_{i=0}^{q} \theta_i x_{t-i} .$$

Виконувати перетворення ряду можна не тільки взяттям послідовних різниць, а і логарифмуванням, розрахунком темпу приросту і т. д. Головне в результаті привести ряд до стаціонарного виду.

Модель ARIMA широко використовується для прогнозування часових рядів, проте вона має певні недоліки. З ARIMA важко моделювати нелінійні зв’язки між змінними [2, 3, 4].

1.2.5. Regression

Linear Regression

У моделі лінійної регресії ми прогнозуємо змінну, що нас цікавить, використовуючи лінійну комбінацію попередніх значень цієї змінної. Термін авторегресія вказує, що це регресія змінної проти неї самої. Розраховується за формулою:

$$x_{t+1} = k_0 + k_1 x_t + k_2 x_{t-1} + k_3 x_{t-2} + \ldots + k_n x_{t-n} ,$$

де k_0 ... k_n - коефіцієнти, знайдені шляхом оптимізації моделі на тренувальних даних, а x_t ... x_{t-n} – вхідні значення [5, 6].

Ridge Regression

Це ще один із типів регресії в машинному навчанні, який зазвичай використовується, коли існує висока кореляція між незалежними змінними. Це пояснюється тим, що у випадку мультиколінеарних даних оцінки за найменшими квадратами дають незміщені значення. Але, якщо колінеарність дуже висока, може бути деяке значення зміщення. Тому в рівняння гребневої регресії вводиться матриця зміщення (λ). Коли $\lambda = 0$, гребнева регресія дорівнює регресії за найменшими квадратами. Якщо $\lambda = \infty$, всі коефіцієнти зменшуються до нуля. Таким чином, ідеальне λ лежить десь між 0 і ∞.

Функція оптимізації:

$$\min || Xw — y ||^2 + \lambda || w ||^2 .$$

Це потужний метод регресії, в якому модель менш сприйнятлива до перенавчання. Регресія працює для рядів, в яких розподіл відмінний від нормального [5, 6].

Lasso Regression (Least absolute shrinkage and selection operator)

У регресії лассо, як і в гребневій, ми додаємо матрицю зміщення в функцію оптимізації для того, щоб зменшити колінеарність і дисперсію моделі. Але замість квадратного зміщення, використовується зміщення абсолютного значення:

$$\min || Xw — y ||^2 + \lambda || w || .$$

Цей тип регуляризації може призвести до розріджених моделей з кількома коефіцієнтами. Деякі коефіцієнти можуть стати нульовими і бути автоматично виключеними з моделі. Більші штрафи призводять до того, що значення коефіцієнта наближаються до нуля, що є ідеальним варіантом для створення простіших моделей. Зазвичай цей метод використовується, коли у нас є багато параметрів, оскільки їх фільтрація виконується автоматично [5, 7, 8].

Elastic Net Regression

Еластична регресія є комбінацією регресії Лассо та гребневої регресії. Вона враховує ефективність обох методів.

$$\min || Xw — y ||^2 + \lambda_1 || w || + \lambda_2 || w ||^2$$

Еластична регресія зазвичай добре працює, коли ми маємо великий набір даних [5].

Polynomial Regression

Поліноміальна регресія — це інша форма регресії, в якій максимальна степінь незалежної змінної більше 1. Ця лінійна модель не є прямою лінією, а має форму кривої.

Квадратична регресія, або регресія з поліномом другого порядку, визначається таким рівнянням:

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 ,$$

де $\theta_0 \dots \theta_2$ – вектори-коефіцієнти, знайдені шляхом оптимізації моделі на тренувальних даних, а x – вектор вхідних значень.

В цілому, регресія - поширений метод для моделювання і прогнозування лінійних часових рядів. Проте регресія дуже чутлива до «шумів» і має тенденцію до перенавчання [5].

1.2.6. Exponential Smoothing models

Simple Exponential Smoothing (SES, Просте експоненційне згладжування)

Експоненційне згладжування використовує середньозважене значення попередніх спостережень, де більша вага приділяється останнім спостереженням і зменшується зі старінням спостережень. Формула виглядає так:

$$x_{t+1} = k_t * x_t + k_{t-1} * x_{t-1} + k_{t-2} * x_{t-2} + \dots + k_n * x_n$$

, де t – час до останнього спостереження (t=0 для останнього спостереження, t=1 для передостаннього і т. д.), x_t – значення останнього спостереження в час t, n – кількість періодів, $k_t = \alpha * (1 - \alpha) * t$. Вибір правильного параметра згладжування (α) часто є ітераційним процесом і його можна спростити за допомогою відомих статистичних методів.

Метод простого експоненційного згладжування не враховує жодних тенденцій або сезонних складових, скоріше, він використовує лише зменшувальні ваги для прогнозування майбутніх результатів. Це робить метод придатним лише для часових рядів без тренду та сезонності [1, 4].

Holt's Model (Метод Хольта)

Цей метод також називають подвійним експоненціальним згладжуванням, і він включає тренд, додаючи другу експоненційну модель. Це призводить до того, що прогнози мають тенденцію до лінійного зростання або зниження залежно від тренду. Цю модель варто застосовувати до будь-якого несезонного набору даних.

Модель прогнозування представляє динаміку часового ряду як лінійну залежність з параметрами, що постійно змінюються:

$$A_t = \alpha x_t + (1-\alpha)(A_{t-1} + T_{t-1}) ,$$

де A_t – згладжена величина на поточний період, α – коефіцієнт згладжування, x_t – поточне значення ряду, A_{t-1} – згладжена величина за попередній період, T_{t-1} – значення тренду за попередній період.

$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1} ,$$

де A_t – згладжена величина на поточний період, β – коефіцієнт згладжування тренду, A_{t-1} – згладжена величина за попередній період, T_{t-1} – значення тренду за попередній період. Значення тренду першого періоду приймається за 0.

Прогноз на p періодів виглядає наступним чином:

$$x_{t+p} = A_t + pT_t .$$

Перевагою моделі Хольта є те, що вона враховує тренд ряду даних. Проте ця модель не враховує сезонність даних і не застосовується для сезонних рядів [1].

Holt Winter’s Exponential Smoothing (HWES, Метод Хольта – Вінтерса)

Метод Хольта - Вінтерса використовується для прогнозування часових рядів, коли в структурі даних є сформований тренд і сезонність. Цей метод містить три рівняння згладжування ряду, тенденції та сезонної складової. Перевага даного методу - це можливість зробити прогноз на тривалий період. Але для того щоб зробити прогноз, наприклад, на 1 рік, знадобляться дані мінімум за 2 повних року, а краще за 3 - 5 повних років.

Прогноз на p періодів виглядає наступним чином:

$$x_{t+p} = A_t + pT_t + S_{t-p+1+(h-1)*mod(p)} ,$$

де згладжений ряд A_t, рівняння тренду T_t, рівняння сезонності S_t розраховуються наступним чином:

$$A_t = \alpha(x_t - S_{t-p}) + (1-\alpha)(A_{t-1} + T_{t-1})$$
$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1}$$
$$S_t = \gamma(x_t - A_t) + (1-\gamma)S_{t-p}\,.$$

Тут α, β, γ є параметрами, які потрібно вибрати перед прогнозуванням. Усі вони змінюються від 0 до 1.

Є мультиплікативний варіант моделі, коли сезонність не додається, а множиться. Так, формула для прогнозування набуде вигляду:

$$x_{t+p} = A_t + pT_t * S_{t-p+1+(h-1)*mod(p)}\,,$$

де згладжений ряд A_t, рівняння тренду T_t, рівняння сезонності S_t розраховуються наступним чином:

$$A_t = \alpha(x_t/S_{t-p}) + (1-\alpha)(A_{t-1} + T_{t-1})$$
$$T_t = \beta(A_t - A_{t-1}) + (1-\beta)T_{t-1}$$
$$S_t = \gamma(x_t/A_t) + (1-\gamma)S_{t-p}\,.$$

Перевагою моделі Хольта – Вінтерса є те, що вона враховує тренд і сезонність часового ряду, але вона потребує багато вхідних даних. При прогнозуванні зашумленого ряду має велику похибку [1, 3].

1.3. Нейромережеві методи для прогнозування часових рядів

Нейромережеві методи — це методи, що ґрунтуються на штучних нейронних мережах. Штучна нейронна мережа являє собою шарувату структуру з пов'язаних нейронів. Структура цієї мережі прийшла в світ програмування завдяки біологічним процесам. Це дозволяє машині аналізувати, запам'ятовувати різну інформацію і також відтворювати її зі своєї пам’яті.

1.3.1. Multi-Layer Perceptron (MLP)

Багатошаровий персептрон (часто званий просто нейронною мережею) є, мабуть, найпопулярнішою архітектурою мережі, яка використовується сьогодні як для класифікації, так і для регресії. Як прикладний підхід машинного навчання, модель MLP передбачає потрійну структуру початкового рівня мережі, яка

приймає вхідні дані, прихований шар вузлів і вихідний шар, який використовується для прогнозування.

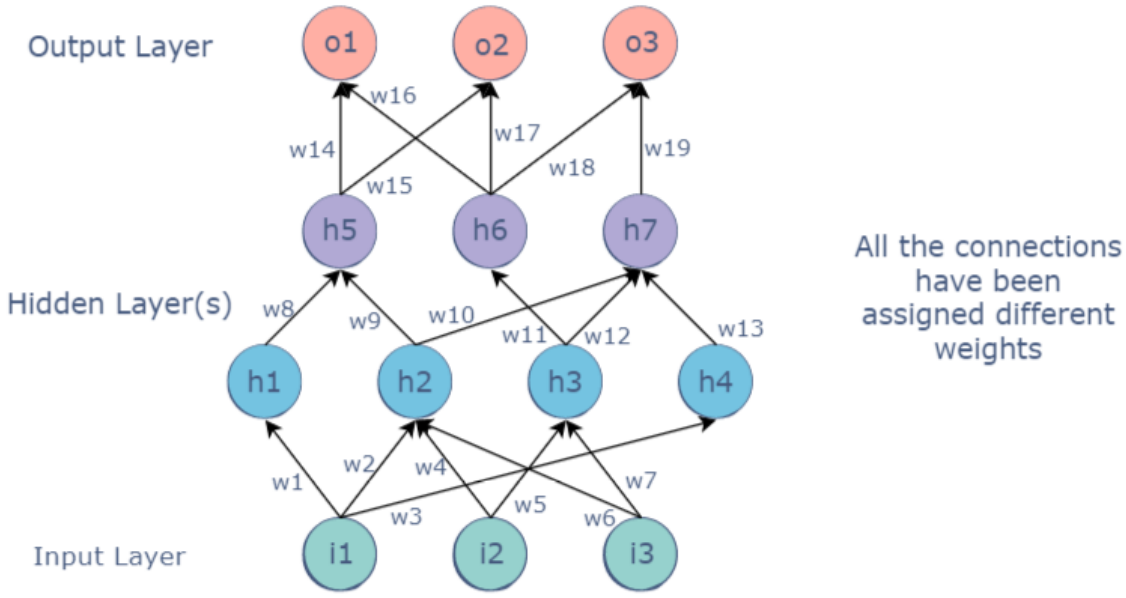

Рис 1.2 - Модель MLP

Математично багатошаровий персептрон описується наступним чином:

$$y = v_0 + \sum_{j=1}^{NH} v_j g(w_j^T x'),$$

де x' - вхідний вектор x, доповнений 1, тобто $x' = (1, x^T)^T$; w_j– вектор ваги для j-го прихованого вузла, $v_0, v_1 \ldots v_{NH}$ — ваги вихідного вузла, а y — вихід мережі, результат. Функція g – логістична функція: $g(u) = 1/(1 + exp(-u))$. Спорідненою моделлю в економетричній літературі є модель авторегресії плавного переходу, яка також базується на побудові лінійних функцій і переходах логістичних функцій.

MLP є сильно параметризованою моделлю, і, вибравши кількість прихованих вузлів NH, ми можемо контролювати складність моделі. Головною перевагою цієї моделі є можливість апроксимувати будь-яку функцію (в нашому випадку часовий ряд) як завгодно близько. Проте для задачі прогнозування при обмежених та зашумлених даних (як це частіше за все і буває) надмірна параметризація скоріше призведе до перенавчання моделі. Тому вибір архітектури

моделі (за допомогою вибору кількості прихованих вузлів та шарів) викликає великий інтерес у літературі про нейронні мережі [8, 9, 10]. Для уникнення перенавчання використовуються різні способи перевірки, наприклад *K-fold*.

Щоб отримати вагові коефіцієнти, визначається середньоквадратична помилка, а вагові коефіцієнти оптимізуються за допомогою градієнтних методів. Найвідомішим методом є алгоритм зворотного поширення.

1.3.2. Recurrent Neural Network (RNN)

RNN – це в основному нейронні мережі з пам’яттю, які можна використовувати для передбачення часових рядів. Повторювані нейронні мережі можуть запам’ятовувати раніше записаний стан вхідних даних, щоб прийняти рішення про майбутній часовий крок. Головним недоліком цієї мережі є те, що вона запам’ятовує невелику кількість попередніх часових кроків і не здатна робити прогнози на довготривалі періоди. Для вирішення цього завдання було розроблено LSTM мережі [11].

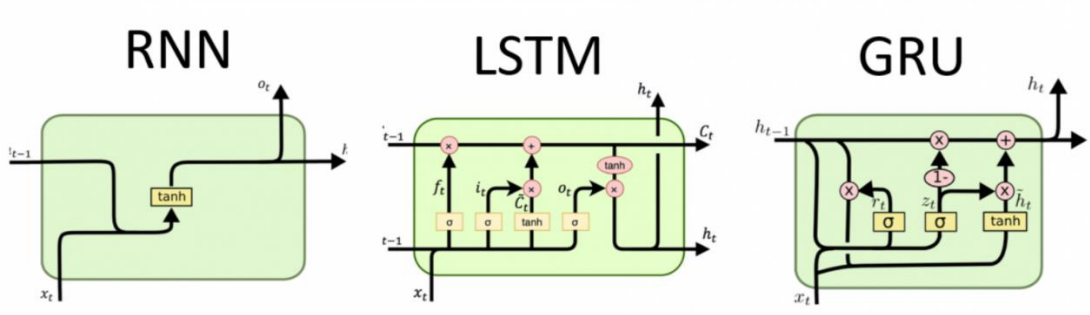

Рис 1.3 – Моделі RNN, LSTM, GRU

Long Short-Term Memory (LSTM)

LSTM – це різновид згорткової нейронної мережі. На відміну від звичайних нейронних мереж з прямими зв’язками, LSTM мережі мають зворотні зв’язки. Ці мережі добре підходять для моделювання та прогнозування часових рядів.

У LSTM мережі існують 3 типи шарів:

1. Шар "забування" (Forget gate) - на вихід подається число від 0 до 1 де 1 позначає необхідність повного запам'ятовування, а 0 повністю стирає з пам'яті.

2. Шар пам'яті (Memory gate) вибирає, які дані необхідно зберегти. Насамперед за допомогою сигмоїдного шару вибираються значення, які потім запам'ятовуються.
3. Вихідний шар (Output gate) вибирає інформацію з кожної «комірки», в якій зроблено запам'ятовування.

LSTM моделі стійкі до «шумів», мають високу точність прогнозів на зашумлених рядах даних. Також мають високу точність прогнозах на довготривалі періоди [11, 12].

Gated Recurrent Units (GRU, Вентильні рекурентні вузли)

GRU – це ще один тип закритої рекурентної мережі, представлений у 2014 році. GRU схожий на LSTM із шаром забування, але має менше параметрів, ніж LSTM, оскільки йому не вистачає вихідного шару.

Щоб вирішити проблему зникаючого градієнта стандартного методу RNN, GRU використовує так звані вентилі оновлення та скидання. По суті, це два вектори, які вирішують, яку інформацію слід передати на вихід. Особливість у них полягає в тому, що їх можна навчити зберігати інформацію з довгий період часу, не змінюючи її часом і не видаляючи інформацію, яка не має відношення до передбачення.

Було показано, що GRU демонструють кращу продуктивність для деяких менших і менш частих наборів даних [11].

1.3.3. Bayesian Neural Network (BNN)

Байєсівська нейронна мережа (BNN) — це нейронна мережа, розроблена на основі байєсівської імовірнісної формулювання. BNN пов’язані з концепцією класичної статистики оцінки байєсівських параметрів, а також пов’язані з концепцією регуляризації. BNN користуються широким застосуванням таких областях, як економіка, фінанси, інженерія. Ідея BNN полягає в тому, щоб розглядати параметри або ваги мережі не як точкові значення, а як випадкові величини, що підкорюються певному розподілу.

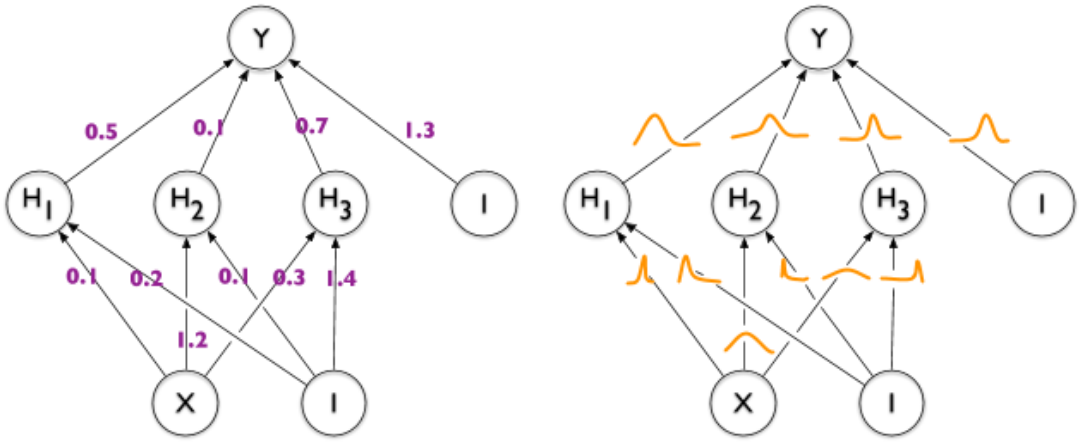

Рис 1.4- Модель BNN

Байєсівські нейронні мережі корисні для вирішення проблем у областях, де даних мало, як спосіб запобігти перенавчанню. Вони можуть отримувати кращі результати для багатьох задач, але їх надзвичайно важко масштабувати до великих проблем. Ще одна перевага байєсівських нейронних мереж перед стандартними нейронними мережами полягає в тому, що можна визначити вхідні дані, де модель не впевнена у своєму передбаченні. Проте вони мають і значні недоліки: BNN значно складніші, ніж стандартні нейронні мережі, що ускладнює їх використання і їх складніше навчити, ніж стандартні нейронні мережі. Значна частина сучасних досліджень, пов'язаних з байєсівськими нейронними мережами, пов'язана з пошуком методів, які полегшать їх навчання [13].

1.3.4. Random Forest (RF)

Випадковий ліс – ансамблевий метод машинного навчання для класифікації, регресії та інших завдань, які використовують за допомогою побудови чистих дерев прийняття рішень під час тренування моделей та виробляють модулі для прогнозів (регресії) побудованих дерев. Ансамблевий метод означає, що одночасно використовуються декілька алгоритмів навчання. Передбачене значення знаходиться як середнє значення незалежних передбачень кожного дерева ансамблю. Ансамблеві методи використовують для покращення результатів прогнозування.

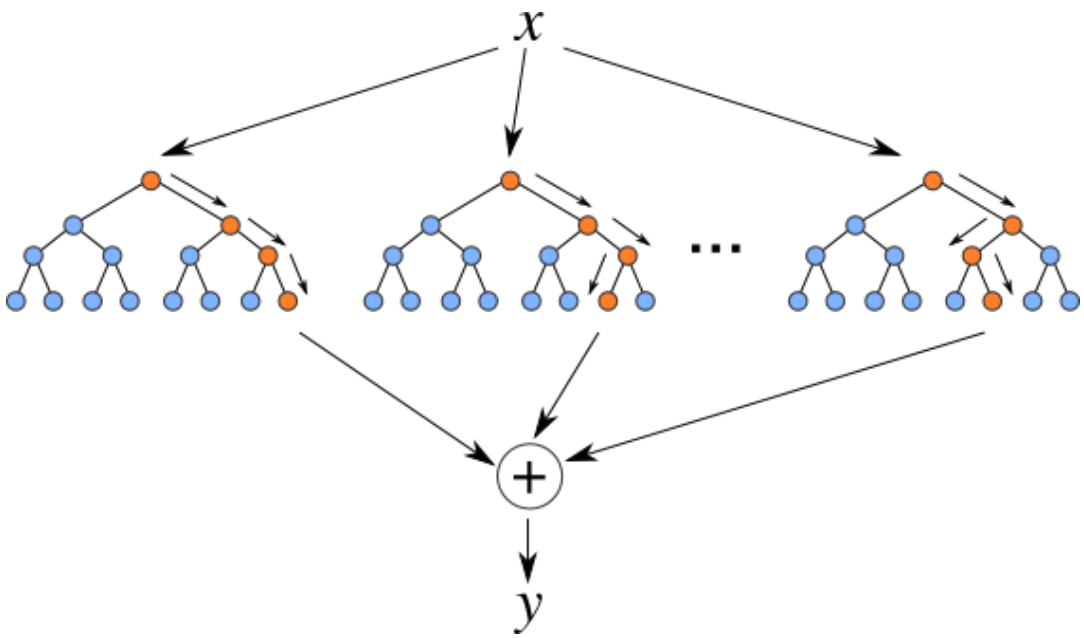

Рис 1.5 - Модель Random Forest

Модель регресії випадкового лісу є потужною та точною. Зазвичай вона відмінно справляється з багатьма проблемами, включаючи функції з нелінійними зв'язками. До недоліків, однак, можна віднести наступне: немає інтерпретації, може легко статися перенавчання, ми повинні вибрати кількість дерев для включення в модель [14, 15].

1.3.5. Gradient Boosting

Gradient Boosting алгоритм використовується для створення моделі ансамблю шляхом поєднання простих прогнозних моделей в єдину композиційну модель. Цей алгоритм можна використовувати для навчання моделей як для задачі регресії, так і для задачі класифікації. Ось чому «boosting» (підвищення) відоме як адитивна модель, оскільки прості моделі додаються по одній, при цьому існуючі дерева в моделі залишаються незмінними. Оскільки ми об'єднуємо все більше і більше простих моделей, повна кінцева модель стає сильнішою. Термін «градієнт» походить від того факту, що алгоритм використовує градієнтний спуск для мінімізації втрат.

Першою реалізацією цього алгоритму, яка мала великий успіх у застосуванні, був Adaptive Boosting або скорочено AdaBoost. Слабкі учні в AdaBoost — це дерева рішень з єдиним розділенням, які через їх малу розгалудженість називають пнями рішень. AdaBoost працює шляхом зважування

спостережень, надаючи більші ваги прикладам, які важко класифікувати, і менші тим, які вже добре оброблені. Послідовно додаються нові слабкі учні, які зосереджують своє навчання на складніших моделях. Це означає, що вибірки, які важко класифікувати, отримують все більші ваги, поки алгоритм не визначить модель, яка правильно класифікує ці вибірки. Прогнози робляться більшістю голосів прогнозів слабких учнів, зважених відповідно до їхньої індивідуальної точності.

Gradient Boosting алгоритм забезпечує високу точність прогнозування. Попередня обробка даних для цього алгоритму не потрібна - часто чудово працює з категоріальними та числовими значеннями як є, обробляє відсутні дані. Проте цей алгоритм потребує багато дерев (>1000), для зберігання яких треба великий об’єм пам’яті а також багато часо для обчислення.

Наразі існує багато реалізацій цього алгоритму, як, наприклад, XGBoost (extreme gradient boosting), LightGBM, CatBoost та інші [14, 15, 16].

1.4. Способи поєднання алгоритмів

Ми приходимо до об'єднання алгоритмів, коли необхідно поєднати переваги кожного алгоритму в одній моделі. Низький зсув і низька дисперсія, хоча вони найчастіше змінюються в протилежних напрямках, є двома найважливішими ознаками, які ми очікуємо від моделі. Справді, щоб мати можливість вирішити поставлену проблему прогнозування продажів, ми хочемо, щоб наша модель мала достатню кількість ступенів свободи, щоб вирішити основну складність даних, з якими ми працюємо, але ми також хочемо, щоб вона не мала занадто великих ступенів свободи, задля уникнення високої дисперсії і щоб бути більш надійною.

Ансамблеве моделювання – особливо популярний метод, який застосовується в data science змаганнях. На цих змаганнях спонсор публікує навчальний і тестовий датасет і ставить глобальний виклик, щоб створити найкращі прогнози тестового набору, для яких встановлений певний критерій продуктивності. Команди-переможці майже завжди використовують ансамблеві моделі замість однієї точно налаштованої моделі.

Існує три основні підходи до композиції алгоритмів:

- Беггінг (bagging)
- Бустинг (boosting)
- Стекінг (stacking)

Bagging

Назва методу bagging розшифровується як “bootstrap aggregating” (завантажене агрегування»). Суть методу полягає у паралельному навчанні однорідних слабких учнів незалежно один від одного і об’єднанні їх за допомогою методу усереднення. Набір даних розбивається на рівні вибірки. Кожна вибірка використовується для навчання окремої моделі. Потім ці моделі комбінуються шляхом усереднення (для регресії) або голосування (для класифікації). Метод часто використовується в статистичній класифікації та регресії. Не зважаючи на те, що беггінг зазвичай застосовується до моделей на основі дерев рішень, його можна застосовувати до будь-яких типів моделей. Алгоритм дозволяє зменшити дисперсію і уникнути перенавчання [10, 17].

Boosting

Суть методу boosting полягає у навчанні однорідних слабких учнів послідовно (базова модель залежить від попередніх) і поєднання їх за детермінованою стратегією. Кожна послідовно під’єднана модель зосереджується на обробці найскладніших спостережень, які можна обробити на даний момент. Тому в кінці ми отримуємо сильну модель з меншим зміщенням. Моделі можна поєднувати двома способами: адаптивно і градієнтно. Бустинг використовується як для задач класифікації, так і для регресії. Алгоритм застосовується з метою зменшення зміщення [10, 17].

Stacking

Stacked Generalization або скорочено stacking — це ансамблевий алгоритм машинного навчання, в якому з допомогою базових алгоритмів отримують прогнози (метаознаки) і використовують їх як ознаки деякого узагальнюючого алгоритму (метаалгоритма). Є різні способи реалізації стекування: використання різних алгоритмів навчання, різні налаштування гіперпараметрів у моделях,

використання різних підмножин навчальних наборів або ж використання різних навчальних наборів. Існують також реалізації багаторівневого стекування, коли на другому рівні ми замість одної мета-моделі навчаємо кілька і на третьому рівні навчаємо останню мета-модель. На відміну від двох попередніх методів, у стекуванні зазвичай поєднуються різні алгоритми навчання [8, 17].

Можна узагальнити, що беггінг в основному зосереджений на отриманні моделі ансамблю з меншою дисперсією, ніж її компоненти, тоді як бустинг та стекування в основному намагатимуться створити сильні моделі, менш упереджені, ніж їхні компоненти (навіть якщо дисперсію також можна зменшити).

ВИСНОВКИ ДО РОЗДІЛУ 1

Перший розділ присвячений дослідженню методів прогнозування часових рядів. Реальні часові ряди зазвичай мають певні тренд та сезонність і є досить зашумленими, через що їх моделювання і прогнозування є складним процесом. Розглянувши багато різних класичних та нейромережевих методів прогнозування часових рядів ми бачимо, що жоден з них не є досконалим та універсальним. Також не існує чітких інструкцій щодо вибору методу для певного типу часового ряду.

Класичні методи прогнозування часових рядів широко використовуються, тому що такі моделі легше інтерпретувати та пояснити отримані прогнози. Проте їх точності не завжди достатньо для вирішення певних задач.

Нейромережеві методи зазвичай дозволяють досягти кращої точності прогнозування, ніж класичні, проте отримані значення важко пояснити, наприклад, замовникові моделі чи прогнозу, важко інтерпретувати як впливають використані фактори на прогнозоване значення. Нейронні мережі важче навчати, адже вони схильні до перенавчання і навчання займає більше часу, тому що їх структура є складнішою.

Поєднання різних моделей для покращення точності прогнозування не є новою ідеєю, існує багато алгоритмів їх поєднання. Проте ця сфера не повністю досліджена. Немає універсальних рекомендацій щодо того які моделі, скільки моделей та яким способом їх слід об’єднувати для вирішення задачі прогнозування часових рядів. До того ж постійно з’являються нові моделі, роботу яких слід досліджувати. Тому було прийнято рішення розробити власний гібридний метод, який би дозволив отримати кращу точність прогнозів, ніж стандартні методи, розглянуті у цьому розділі, та дослідити його ефективність для прогнозування часових рядів.

РОЗДІЛ 2. РОЗРОБКА ГІБРИДНОГО МЕТОДУ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

Для підвищення точності прогнозування необхідно розробити гібридний метод з урахуванням особливостей часового ряду, що аналізується. При розробці методу слід вирішити наступні питання:

- Визначитись з моделями, що будуть входити в розроблюваний метод;
- Обрати архітектуру, за якою ці моделі будуть поєднані;
- Обрати програмний засіб для реалізації розробленого методу;
- Визначити оптимальні параметри моделі;
- Визначити засоби запобігання перенавчанню моделі;
- Визначити методи оцінки моделі.

2.1. Вибір архітектури та методу навчання гібридної моделі

Реальні часові ряди, як правило, мають чітко виражену сезонність та тренд, а також велику кількість шумів в даних. Для прогнозування прогнозування таких рядів слід використовувати гібридні алгоритми, які поєднують у собі кілька різнорідних моделей. Поєднання саме різнорідних моделей дасть збільшення різноманітнітності і взаємодоповнюваності базових предикторів шляхом поєднання набору різних регресорів для покращення ефективності регресії.

Стекування – це техніка, яка використовується для об’єднання кількох гетерогенних моделей шляхом навчання додаткової моделі – метарегресора. Концепція стекування полягає в тому, що деякі моделі можуть краще узагальнювати дані, тоді як інші можуть виявити сезонні коливання при гіршому узагальненні. Метарегресор вчиться на основі цього різноманіття передбачень і намагається об’єднати моделі, щоб покращити прогнозовану точність базових моделей. Девід Уолперт [18] стверджує, що стек моделей виводить зміщення в моделі на певний набір даних (метаознак), щоб пізніше ці зміщення можна було виправити за допомогою метарегресора.

Архітектуру розробленої гібридної моделі наведено на рисунку 2.1.

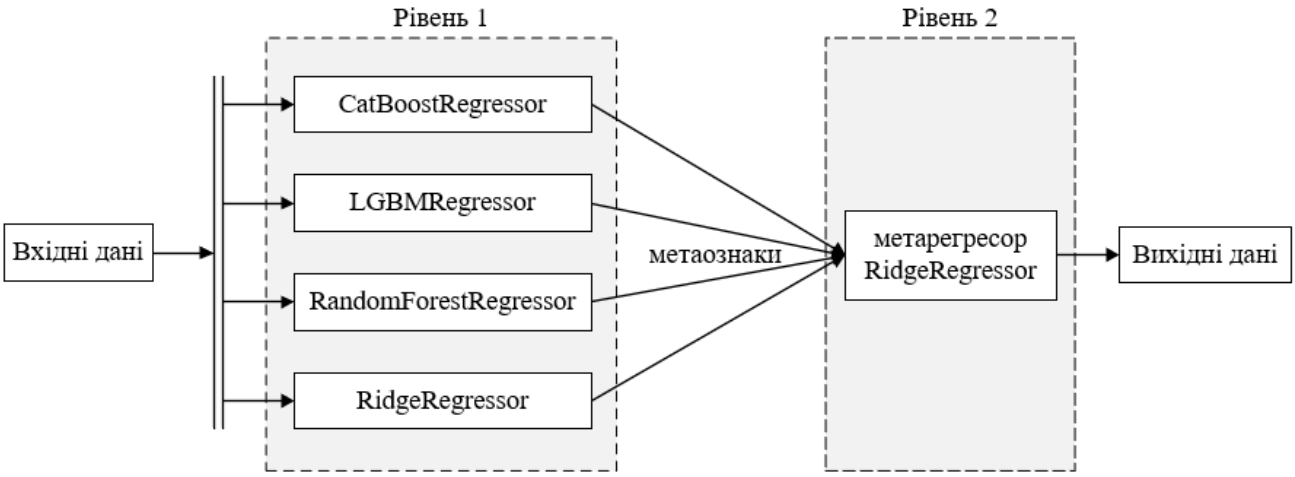

Рис 2.1 – Обрана архітектура моделі

Поєднуючи різнорідні моделі, ми хочемо отримати фінальну модель, яка матиме переваги всіх об'єднаних моделей. RidgeRegressor – лінійна модель, яка добре узагальнює дані, але не здатна прогнозувати коливання продажів. CatBoost, RandomForest та LGBMR навпаки ж при певних налаштуваннях мають тенденцію повторювати коливання і шуми в даних і таким чином спотворювати прогнози. При поєднанні цих моделей випадкові шуми в їх прогнозах будуть компенсуватись і вирівнюватись, однак сезонні коливання навпаки мають набути більш вираженого характеру, чого нам і треба досягти.

Поєднаємо перелічені моделі методом стекування і використаємо метарегресор RidgeRegressor, оскільки саме регресія Ріджа зазвичай використовується, коли існує висока кореляція між вхідними змінними. Задача метарегресора – віднайти оптимальні ваги, з якими слід враховувати прогноз кожної моделі, щоб компенсувати всі їх слабкі сторони.

Передбачення всіх моделей (метаознаки), виконані на етапі 1, використовуються як вхідні дані для навчання метарегресора.

Алгоритм навчання такої моделі буде наступним:

1. Підібрати найкращі параметри для кожної з моделей 1 рівня – CatBoostRegressor, RidgeRegressor, RandomForestRegressor, LGBMRegressor.
2. Навчити всі моделі першого рівня і створити 4 набори прогнозів (по одному для кожної моделі 1 рівня).

3. Використати 4 набори, які були отримані на 2 кроці як вхідні дані для метарегресора.
4. Навчити метарегресор на вхідних даних та отримати кінцеві прогнози.

2.2. Вибір програмного засобу для реалізації гібридної моделі

Python є мовою високого рівня, яка використовуються data science спеціалістами для різних проектів/додатків із науки про дані. Python надає велику функціональність для роботи з математикою, статистикою та науковими функціями. Більшу частину роботи з дослідження даних можна виконати за допомогою п’яти бібліотек: NumPy, Pandas, Scipy, Scikit-learn і Seaborn. Фреймворки глибокого навчання, доступні з API Python, на додаток до наукових пакетів зробили Python неймовірно продуктивним і універсальним.

NumPy є стандартом для виконання обчислень масивів сьогодні. Ця бібліотека написана на мові C, тому дозволяє швидко виконувати різні математичні операції. NumPy підтримує широкий спектр апаратних і обчислювальних платформ і добре працює з графічними процесорами, використання яких дозволить пришвидшити навчання розробленої гібридної моделі. SciPy розширює NumPy, надаючи додаткові інструменти для обчислень масивів [19, 20].

Pandas — це швидкий, потужний, гнучкий і простий у використанні інструмент для аналізу та маніпулювання даними з відкритим вихідним кодом, створений на основі мови програмування Python. Pandas має інструменти для читання та запису даних у різних форматах: CSV та текстові файли, Microsoft Excel, бази даних SQL та швидкий формат HDF5. Ця бібліотека дозволяє виконувати перетворення даних, такі як операції розділення-додавання-об'єднання над наборами даних. Крім цього має функціонал для роботи з часовими рядами: генерація діапазону дат, перетворення частоти, статистика рухомого вікна, зсув і відставання дати [21].

Scikit-learn побудована на NumPy, SciPy і Matplotlib і реалізує прості та ефективні інструменти для прогнозного аналізу даних [22]. Зокрема це інструменти:

- для попередньої обробки даних – стандартизація, нормалізація, кодування категорійних змінних та інші;
- вже реалізовані певні моделі для регресії та класифікації;
- для підбору параметрів моделей;
- метрики для оцінки моделей;

Seaborn — це бібліотека для створення статичних візуалізацій на мові Python. Вона побудована на основі бібліотеки matplotlib і тісно інтегрується із структурами даних Pandas [23, 24]. Seaborn пропонує високорівневий інтерфейс: побудова більшості простих графіків реалізується в один рядок коду. Також вона забезпечує більш привабливу візуалізацію, ніж Matplotlib.

В бібліотеці MLXtend є реалізовані функції, які дозволять спростити реалізацію методу. Функція StackingCVRegressor реалізує стекування з перехресною перевіркою для задачі регресії. Як вхідні дані подаються початкові моделі першого рівня та метамодель. Для кожної вхідної моделі можна задати перелік параметрів. Також можна одразу виконати підбір параметрів усіх моделей, використовуючи метод GridSearchCV з бібліотеки sklearn.model_selection [25].

2.3. Запобігання перенавчанню гібридної моделі

Перенавчання є особливо великою проблемою в стекінгових моделях, оскільки поєднується багато предикторів, кожен з яких намагається передбачити однакові дані. Перенавчання часто спричинене колінеарністю між предикторами.

Для запобігання перенавчанню ми використаємо перехресну перевірку k-fold. Набір даних буде розбито на k частин, і в k послідовних циклах k-1 частина буде використана для навчання регресорів першого рівня. У кожному циклі регресори першого рівня після навчання застосовуються до 1 підмножини (out-of-fold, OOF), яка не була використана для навчання моделей на кожній ітерації. Сума квадратів помилок між прогнозами OOF і справжніми цільовими

значеннями дасть похибку перехресної перевірки моделі і буде хорошою мірою узагальнення. Після цього отримані прогнози об'єднуються в стек і надаються як вхідні дані регресору другого рівня. Після навчання регресори першого рівня роблять передбачення для всього набору даних для отримання оптимальних прогнозів.

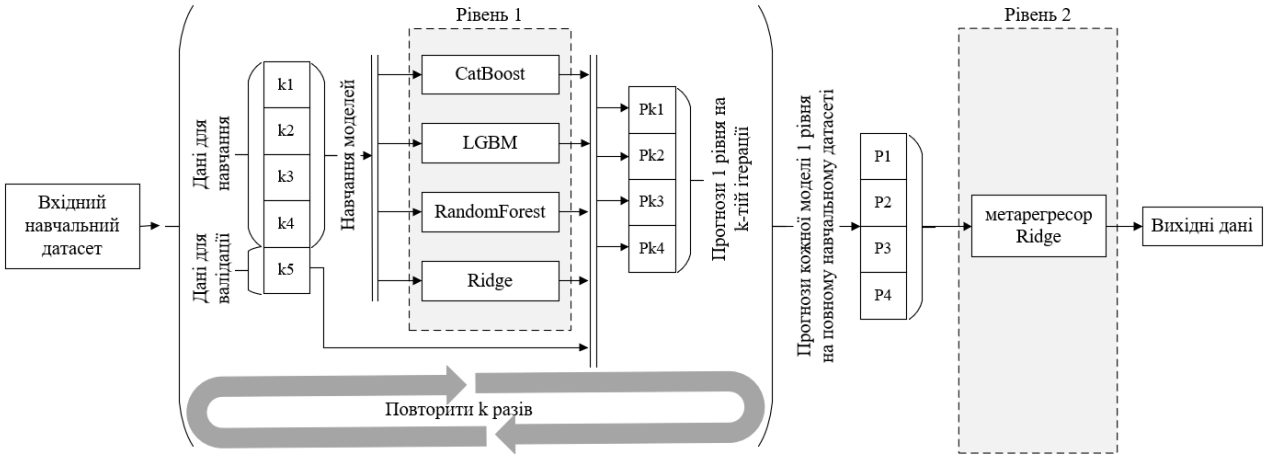

Рис 2.2 – Модель з перехресною перевіркою

Відкоригований алгоритм навчання моделі виглядатиме так:

1. Підібрати найкращі параметри для кожної з моделей 1 рівня – CatBoostRegressor, RidgeRegressor, RandomForestRegressor, LGBMRegressor – окремо за допомогою автоматичного налаштування.
2. Розділити навчальний датасет на k=5 рівних частин.
3. Використати 4 частини датасету для навчання моделей 1 рівня.
4. На п'ятій частині датасету отримати прогнози для кожної з моделей 1 рівня.
5. Повторити пункти 2-4 k=5 разів, щоб отримати прогнози для кожної моделі 1 рівня на кожній частині розділеного датасету.
6. 5 отриманих передбаченнь на неповних наборах для кожної моделі об'єднати в стек. Як результат цього пункту ми отримаємо прогнози на повних вхідних даних для кожної з 4 моделей 1 рівня.
7. Використати 4 набори, які були отримані на 6 кроці як вхідні дані для метарегресора RidgeRegressor.
8. Навчити метарегресор на вхідних даних та отримати кінцеві прогнози.

2.4. Оптимізація параметрів гібридної моделей

Гіперпараметри – це параметри моделі , які не вивчаються безпосередньо на етапі навчання моделі. Сенс оптимізації гіперпараметрів полягає у встановленні таких значень гіперпараметрів і зміни їх у часі, за яких досягається максимально можлива ефективність і точність гібридної моделі. Для оптимізації гіперпараметрів використаних моделей на усіх рівнях використаємо стандартний метод GridSearchCV з бібліотеки sklearn.model_selection. Цей метод перебирає всі можливі комбінації параметрів із заданих списків для кожного параметру та знаходить серед них найкращу комбінацію.

Будемо підбирати наступні гіперпараметри:

- RidgeRegressor
    - ridge__alpha – сила регуляризації. Регуляризація зменшує дисперсію оцінок. Більші значення визначають сильнішу регуляризацію.
- RandomForestRegressor
    - randomforestregressor__n_estimators – кількість дерев в лісі.
    - randomforestregressor__max_depth – обмеження максимальної глибини дерева.
    - randomforestregressor__min_samples_leaf – мінімальна кількість зразків, яка повинна бути у вузлі листка. Точка розщеплення на будь-якій глибині буде розглядатися, лише якщо вона залишає принаймні min_samples_leaf навчальні вибірки в кожній з лівої та правої гілок. Для регресії це має ефект згладжування моделі.
    - randomforestregressor__min_samples_split – мінімальна кількість вибірок, необхідних для розділення вузла.
- LGBMRegressor
    - lgbmregressor__n_estimators – кількість ітерацій.
    - lgbmregressor__learning_rate – швидкість навчання.
    - lgbmregressor__max_depth – обмеження максимальної глибини дерева.

    - lgbmregressor__min_child_samples – мінімальна кількість даних в одному листку. Впливає на навчання так само, як min_samples_leaf в RandomForestRegressor.
    - lgbmregressor__min_child_weight – мінімальна сума ваг екземплярів, потрібна для листа (дитини).
    - lgbmregressor__num_leaves – кількість листків в одному дереві.
- CatBoostRegressor – залишаємо стандартні параметри, оскільки метод GridSearchCV не підтримує підбір параметрів для даної моделі.
- Мета-регресор RidgeRegressor
    - meta_regressor__alpha – сила регуляризації. Регуляризація зменшує дисперсію оцінок. Більші значення визначають сильнішу регуляризацію.

2.5. Визначення ефективності гібридної моделі

При застосуванні гібридної моделі до реального часового ряду спочатку необроблені дані поділяються на дві частини, а саме навчальний і тестовий набори.

Спостереження в навчальному наборі використовуються для побудови потрібної моделі. Невелика частина навчального набору зберігається з метою перевірки і називається набором валідації (Validation Set). Іноді попередня обробка проводиться шляхом нормалізації даних або прийняття логарифмічних чи інших перетворень. Після побудови моделі вона використовується для генерування прогнозів. Спостереження тестового набору зберігаються для перевірки наскільки точно працювала модель при прогнозуванні цих значень. При необхідності на прогнозовані значення застосовується обернене перетворення для їх повернення в початковий вид. Для того, щоб оцінити точність прогнозування моделі або оцінити та порівняти різні моделі, враховується їх відносна ефективність на тестовому наборі даних. З цієї причини пропонуються різні заходи щодо ефективності оцінки точності прогнозу та порівняння різних

моделей. Вони також відомі як показники ефективності. Кожен з цих заходів є функцією фактичних та прогнозованих значень часового ряду.

Значення «втрат», як правило, повідомляються алгоритмами глибинного навчання. Технічна втрата - це певна кара за поганий прогноз. Більш конкретно, значення втрат буде нульовим, якщо прогноз моделі ідеальний. Отже, метою є мінімізація значень втрат за рахунок отримання набору ваг і відхилень, що мінімізує втрати. Крім втрат, які використовуються алгоритмами глибинного навчання, часто використовуються такі метрики як середнє квадратичне відхилення (RMSE), середня абсолютна похибка (MAE) та середня абсолютна похибка у % (MAPE) для оцінки показників прогнозування.

RMSE

RMSE переважно вимірює середньоквадратичну помилку наших прогнозів. RMSE – це просто квадратний корінь із MSE. Квадратний корінь введений, щоб масштаб помилок був таким самим, як масштаб цілей. Для кожної точки обчислюється квадратна різниця між прогнозами та метою, усереднюються ці значення і з результату обчислюється квадратний корінь:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2},$$

де N - загальна кількість спостережень, y_i - фактичне значення; тоді як $\hat{y}_i$ - це прогнозоване значення.

Чим вище це значення, тим гірша модель. RMSE ніколи не буває негативним, оскільки ми зводимо в квадрат окремі помилки прогнозування, перш ніж підсумовувати їх, але для ідеальної моделі це буде нуль.

Перевага: Корисно, якщо ми маємо несподівані значення, про які ми повинні піклуватися. Дуже високе чи низьке значення ряду, на яке ми повинні звернути увагу.

MAE

У MAE помилка розраховується як середнє абсолютних різниць між цільовими значеннями та прогнозами. MAE - це лінійна оцінка, яка означає, що

всі індивідуальні відмінності зважені однаково в середньому. Наприклад, різниця між 10 і 0 буде вдвічі більша від різниці між 5 і 0. Однак те ж саме не вірно для RMSE. Математично MAE розраховується за такою формулою:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|\,,$$

де N - загальна кількість спостережень, y_i - фактичне значення; тоді як $\hat{y}_i$ - це прогнозоване значення.

MAE не такий чутливий до шумів, як середньоквадратична помилка.

MAPE

Для MAPE вага її вибірки обернено пропорційна її цільовим значенням. Але, яку ми платимо за фіксовану абсолютну помилку, залежить від цільового значення. І коли ціль збільшується, ми платимо менше.

$$MAPE = \frac{100\%}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right|,$$

де N - загальна кількість спостережень, y_i - фактичне значення; тоді як $\hat{y}_i$ - це прогнозоване значення.

ВИСНОВКИ ДО РОЗДІЛУ 2

У рамках даного розділу було розроблено модель для прогнозування часового ряду даних продажів (сам часовий ряд описано в розділі 3), обрано метрики для оцінки моделі та обґрунтовано вибір мови програмування. Крім цього, в даному розділі було вирішено наступні завдання:

1. Було обрано моделі, що будуть входити в розроблюваний метод: CatBoost, Ridge, RandomForest, LGBM. Поєднання різнорідних алгоритмів повинно забезпечити кращу точність прогнозування моделі.
2. Обрано архітектуру, за якою ці моделі будуть поєднані – stacking. Слабкі сторони моделей першого рівня буде компенсовано метарегресором RidgeRegressor на другому рівні розробленої гібридної моделі. Моделі першого рівня будуть прогнозувати однакові дані і кореляція між отриманими метаознаками буде високою, а саме регресія Ріджа використовується коли існує висока кореляція між вхідними змінними.
3. Обрано програмний засіб для реалізації розробленого методу. Мова Python надає всі необхідні бібліотеки для реалізації фінальної моделі – Pandas, NumPy, SciPy, Scikit-Learn, MLXtend, Matplotlib, Seaborn. Надано короткі відомості про бібліотеки, що будуть використовуватись.
4. Визначено оптимальні параметри усіх моделей що входять до розробленого гібридного методу. Це дозволить запобігти перенавчанню гібридної моделі та зменшити дисперсію при прогнозуванні.
5. Для запобігання перенавчанню було вирішено використовувати 5-кратну перехресну перевірку моделей першого рівня.
6. Для оцінки ефективності моделі обрано метрики MAE, MAPE та RMSE.

РОЗДІЛ 3. ОПИС ДАТАСЕТУ ПРОДАЖІВ В МАГАЗИНІ МЕРЕЖІ FAVORITA В СТОЛИЦІ ЕКВАДОРУ ТА ТЕСТУВАННЯ ГІБРИДНОЇ МОДЕЛІ

У розділі описано обраний датасет. Виконано дослідження точності прогнозування описаної в розділі 2 гібридної моделі на обраному датасеті. Порівняно ефективність гібридної моделі з стандартними моделями, описаними в розділі 1, за обраними метриками.

3.1. Опис та аналіз вхідних даних продажів товару в магазині мережі Favorita в Еквадорі

Як набір даних було обрано продажі одного товару в мережі магазинів Favorita в Еквадорі за 2013-2017 роки. Початковий набір містить дані продажів 4036 товарів в 54 магазинах Favorita і складається з кількох файлів з даними [26].

Файл train.csv містить датовану інформацію про магазин і товар, інформацію про те, чи рекламувався цей товар, а також продажі в одиницях товару.

Табл 3.1 – Поля в файлі train.csv

Назва поля	**Тип даних**	**Коментар**
Id	Int64	Номер рядка даних.
Date	Object	Дата.
Store_nbr	Int64	Номер магазину. Містить 54 номери, що позначають різні магазини.
Item_nbr	Int64	Номер товару. Містить 4036 номерів, що позначають різні товари.
Unit_sales	Float64	Продажі можуть бути цілим числом (наприклад, мішок чіпсів) або дробовим (наприклад, 1,5 кг сиру). Від'ємні значення означають повернення цього конкретного товару.

Назва поля	Тип даних	Коментар
Onpromotion	Object	У стовпці onpromotion вказується, чи був цей товар в акції у цей день у цьому магазині. Приблизно 16% значень в цьому стовбці - NaN.

Файл stores.csv містить дані про магазини: місто, штат, тип магазину і кластер. Кластер — це група подібних магазинів.

Табл 3.2 – Поля в файлі stores.csv

Назва поля	Тип даних	Коментар
Store_nbr	Int64	Номер магазину.
City	Object	Місто, в якому розташований магазин. Всього 22 міста в наборі даних.
State	Object	Назва штату. Всього 16 штатів в наборі даних.
Type	Object	Категорія магазину, що позначається літерами з діапазону A-E.
Cluster	Int64	Номер групи подібних магазинів. Всього 17 кластерів.

Файл items.csv містить дані про товари: категорію товару, клас і чи швидко товар псується.

Табл 3.3 – Поля в файлі items.csv

Назва поля	Тип даних	Коментар
Item_nbr	Int64	Номер товару. Містить 4036 номерів, що позначають різні товари.
Family	Object	Категорія товару, наприклад: бакалія, м’ясо, напої, особистий догляд, книги, білизна та інші.
Class	Int64	Числове позначення товару.

Назва поля	Тип даних	Коментар
Perishable	Int64	Позначає чи товар швидко псується. Якщо швидко псується, значення = 1, інакше = 0.

Файл transactions.csv містить дані про кількість транзакцій продажу для кожної дати по магазинах.

Табл 3.4 – Поля в файлі transactions.csv

Назва поля	Тип даних	Коментар
Date	Object	Дата.
Store_nbr	Int64	Номер магазину.
Transactions	Int64	Кількість транзакцій. Числове значення лежить в діапазоні 5 - 8359.

Файл oil.csv містить дані про добову ціна нафти. Еквадор – залежна від нафти країна, і його економічний стан дуже вразливий до зміни цін на нафту.

Табл 3.5 – Поля в файлі oil.csv

Назва поля	Тип даних	Коментар
Date	Object	Дата.
Dcoilwtico	Float64	Добова ціна нафти. Містить 43 значення NaN.

Файл holidays_events.csv містить дані про свята.

Табл 3.6 – Поля в файлі holidays_events.csv

Назва поля	Тип даних	Коментар
Date	Object	Дата.
Type	Object	Тип подій: свято з вихідним днем; свято без вихідного дня; вихідний день, що був перенесений на честь свята, яке випало на вихідний; надзвичайні події.

Назва поля	**Тип даних**	**Коментар**
Locale	Object	Позначає чи це свято національного / регіонального / локального масштабу.
Locale_name	Object	Назва регіону, де проходить свято.
Description	Object	Назва свята
Transferred	Bool	Позначає чи був вихідних на честь свята перенесений на інший день.

Для прогнозування з цього датасету ми взяли дані продажів одного товару в одному магазині. Із файлу train.csv використали саме дані продажів. Із файлу items.csv використали інформацію про товар. Із файлу oil.csv використали дані про добову ціну нафти, оскільки цей економічний показник є важливою ознакою купівельної спроможності громадян Еквадору. Із файлу holidays_events.csv ми взяли дані про свята, святкування яких не було перенесено на інші дні. Ми взяли саме не перенесені свята, оскільки дні, коли за календарем є свято, але його святкування переноситься на інший день більше по характеристикам схожі на звичайні будні дні. Якщо у цей день є певне свято, то в колонці holiday стоїть значення 1. Зведений датасет представлено на рисунку 3.1.

	id	date	store_nbr	item_nbr	unit_sales	onpromotion	family	class	perishable	dcoilwtico	holiday
0	32292	2013-01-02	44	103520	12.0	NaN	GROCERY I	1028	0	93.14	NaN
1	72723	2013-01-03	44	103520	9.0	NaN	GROCERY I	1028	0	92.97	NaN
2	112569	2013-01-04	44	103520	11.0	NaN	GROCERY I	1028	0	93.12	NaN
3	154674	2013-01-05	44	103520	16.0	NaN	GROCERY I	1028	0	NaN	1.0
4	196658	2013-01-06	44	103520	11.0	NaN	GROCERY I	1028	0	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...
1578	125052918	2017-08-11	44	103520	7.0	False	GROCERY I	1028	0	48.81	1.0
1579	125158259	2017-08-12	44	103520	9.0	False	GROCERY I	1028	0	NaN	NaN
1580	125263904	2017-08-13	44	103520	5.0	False	GROCERY I	1028	0	NaN	NaN
1581	125368254	2017-08-14	44	103520	1.0	False	GROCERY I	1028	0	47.59	NaN
1582	125471186	2017-08-15	44	103520	4.0	False	GROCERY I	1028	0	47.57	NaN

1583 rows × 11 columns

Рис 3.1 – Початковий зведений датасет

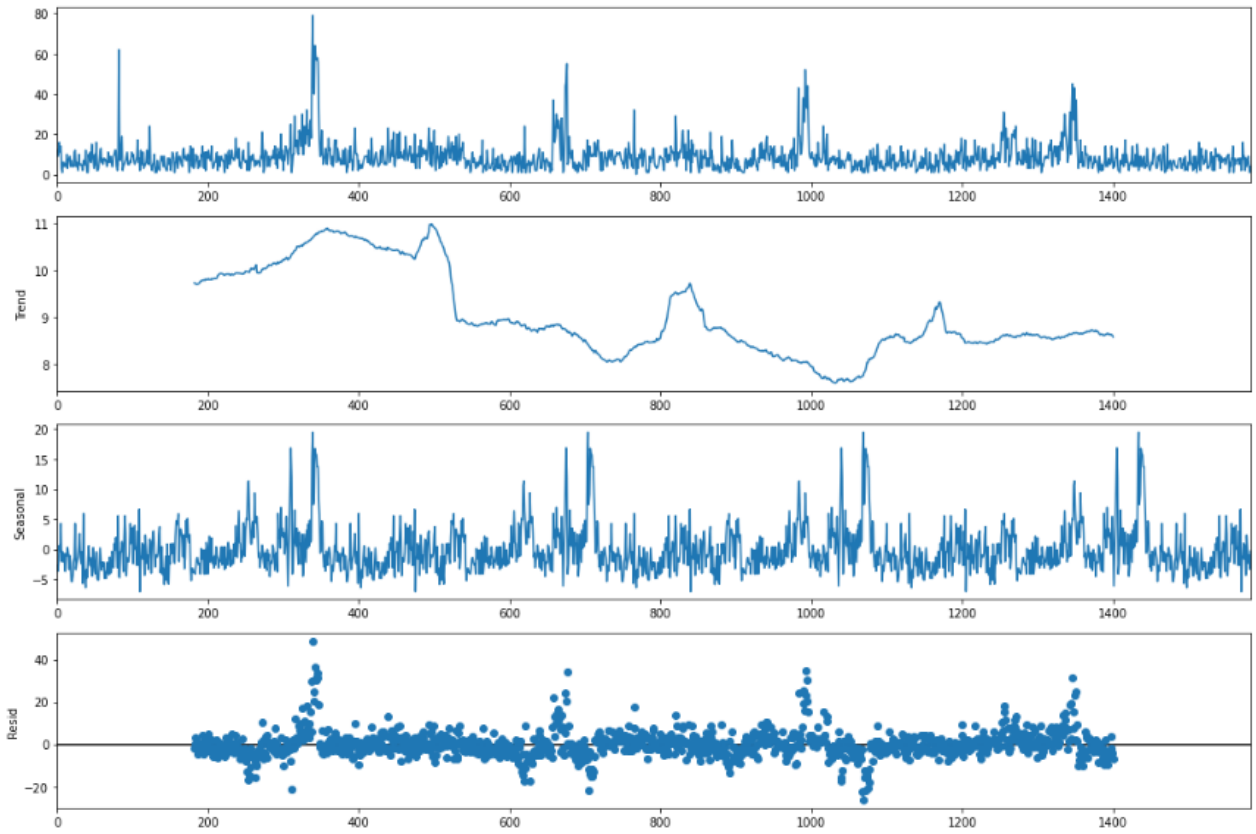

Рис 3.2 – Декомпозиція часового ряду даних

Як видно з графіків на рис. 3.2, часовий ряд має чітку виражену сезонність і спадаючий тренд. Схоже на те, що попит на даний товар підвищується перед новорічними святами. Але, так як тренд спадаючий, то товар стає менш популярним з роками, можливо навіть через купівельну спроможність громадян Еквадору, оскільки ціна за барель нафти спадає.

Можемо також розглянути автокореляцію з лагами від 1 до 50 на рис. 3.3.

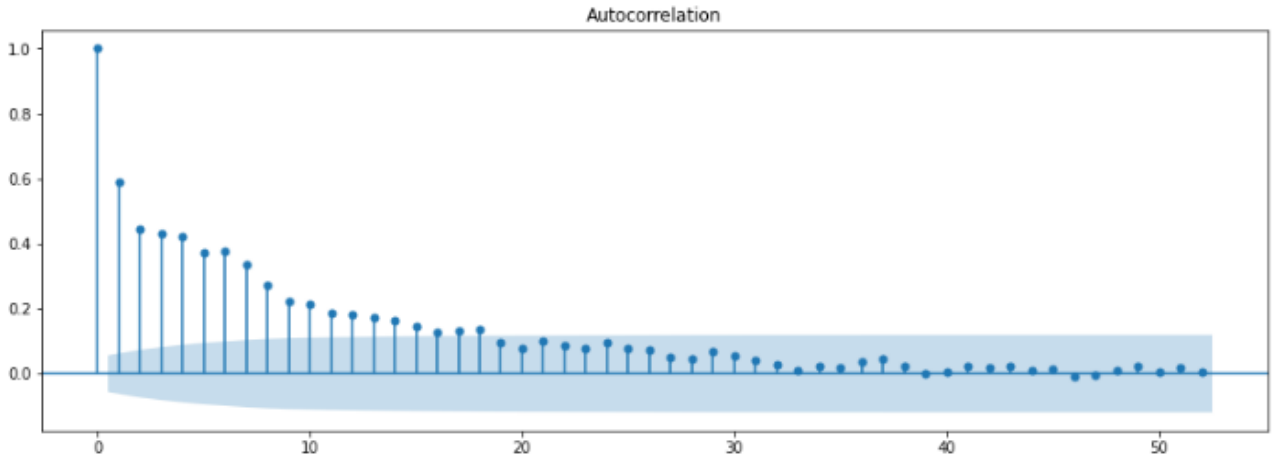

Рис 3.3 – Графік автокореляції продажів

На графіку ми бачимо синій коридор значущості відмінності кореляції від 0. Ця значущість рахується за допомогою критерія Стьюдента. Якщо кореляція не виходить за рамки цього коридору, то вона незначно відрізняється від 0. З графіка видно, що продажі продукту сильно корелюють з продажами цього ж продукту до 10 днів назад. Такий вид корелограми означає, що у вихідного ряду продажів є чітко виражений тренд.

3.2. Попередня обробка вхідних даних

Так як ми обрали дані одного товару в одному магазині, то номер магазину, клас товару, «сім'я» товару і номер товару ніякої важливої інформації для прогнозування не несуть. Також колонка id для прогнозування нам не потрібна.

	id	store_nbr	item_nbr	unit_sales	class	perishable	dcoilwtico	holiday
count	1.583000e+03	1583.0	1583.0	1583.000000	1583.0	1583.0	1087.000000	130.0
mean	5.242130e+07	44.0	103520.0	8.725837	1028.0	0.0	68.449154	1.0
std	3.738769e+07	0.0	0.0	7.250398	0.0	0.0	25.925238	0.0
min	3.229200e+04	44.0	103520.0	-1.000000	1028.0	0.0	26.190000	1.0
25%	1.892710e+07	44.0	103520.0	5.000000	1028.0	0.0	46.650000	1.0
50%	4.592896e+07	44.0	103520.0	7.000000	1028.0	0.0	53.300000	1.0
75%	8.467504e+07	44.0	103520.0	11.000000	1028.0	0.0	96.255000	1.0
max	1.254712e+08	44.0	103520.0	79.000000	1028.0	0.0	110.620000	1.0

Рис 3.4 – Детальна інформація про початковий датасет

Поглянувши на детальну інформацію на рисунку 3.4, бачимо, в колонках holiday та dcoilwtico є пропущені дані. В колонці holiday зробимо заміну пропущених даних на 0, оскільки в ці дні просто не було ніяких свят.

В колонці dcoilwtico є багато пропущених значень (рис. 3.5), але це важливий показник і просто виключити його з датасету не можна. Бачимо, що дані про добову ціну нафти пропущені не підряд, а лише за певні вибіркові дні. Тому доповнимо цю колонку середніми значеннями за попередній та наступний день.

	id	date	store_nbr	item_nbr	unit_sales	onpromotion	family	class	perishable	dcoilwtico	holiday
3	154674	2013-01-05	44	103520	16.0	NaN	GROCERY I	1028	0	NaN	1.0
4	196658	2013-01-06	44	103520	11.0	NaN	GROCERY I	1028	0	NaN	NaN
10	433911	2013-01-12	44	103520	4.0	NaN	GROCERY I	1028	0	NaN	1.0
11	475970	2013-01-13	44	103520	5.0	NaN	GROCERY I	1028	0	NaN	NaN
16	717276	2013-01-19	44	103520	8.0	NaN	GROCERY I	1028	0	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...
1566	123791209	2017-07-30	44	103520	7.0	False	GROCERY I	1028	0	NaN	NaN
1572	124431986	2017-08-05	44	103520	16.0	False	GROCERY I	1028	0	NaN	NaN
1573	124542678	2017-08-06	44	103520	8.0	False	GROCERY I	1028	0	NaN	NaN
1579	125158259	2017-08-12	44	103520	9.0	False	GROCERY I	1028	0	NaN	NaN
1580	125263904	2017-08-13	44	103520	5.0	False	GROCERY I	1028	0	NaN	NaN

496 rows × 11 columns

Рис 3.5 – Пропущені значення dcoilwtico

В колонці з продажами бачимо мінімальне значення -1. Це означає, що в певний день не було продано жодної одиниці товару, а натомість було здійснено повернення. Так як ціль нашого прогнозування – прогнозування продажів – замінимо це значення на 0.

Змінну date у такому вигляді, яка вона зараз є, ми не можемо використати як вхідний параметр моделі, оскільки вхідні параметри повинні бути цілим чи дробовм числом. Тому з цієї змінної ми виділимо ряд інших змінних:

- year_item_purchased – рік;
- quarter_item_purchased – квартал;
- month_item_purchased – місяць;
- week_item_purchased – тиждень;
- day_item_purchased – день.

	onpromotion	dcoilwtico	holiday	day_item_purchased	month_item_purchased	quarter_item_purchased	week_item_purchased	year_item_purchased
count	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000
mean	0.042728	74.233878	0.086278	15.599836	5.953164	2.329499	24.101890	2014.324569
std	0.202326	26.855668	0.280889	8.836793	3.274745	1.071027	14.291818	1.074830
min	0.000000	26.190000	0.000000	1.000000	1.000000	1.000000	1.000000	2013.000000
25%	0.000000	47.170000	0.000000	8.000000	3.000000	1.000000	12.000000	2013.000000
50%	0.000000	90.880000	0.000000	16.000000	6.000000	2.000000	23.000000	2014.000000
75%	0.000000	97.940000	0.000000	23.000000	8.000000	3.000000	35.000000	2015.000000
max	1.000000	110.620000	1.000000	31.000000	12.000000	4.000000	53.000000	2016.000000

Рис 3.6 – Опис вхідних даних після відбору параметрів та заповнення пропусків в даних

Всі вхідні дані мають різну розмірність і якщо ми будемо навчати нашу модель на таких даних, високої точності не отримаємо. Тому ми виконуємо масштабування наших вхідних і вихідних даних. Формула, за якою виконується масштабування:

$$z = \frac{x - x_{min}}{x_{max} - x_{min}}.$$

Опис масштабованих вхідних і вихідних даних наведено на рисунках 3.7-3.8.

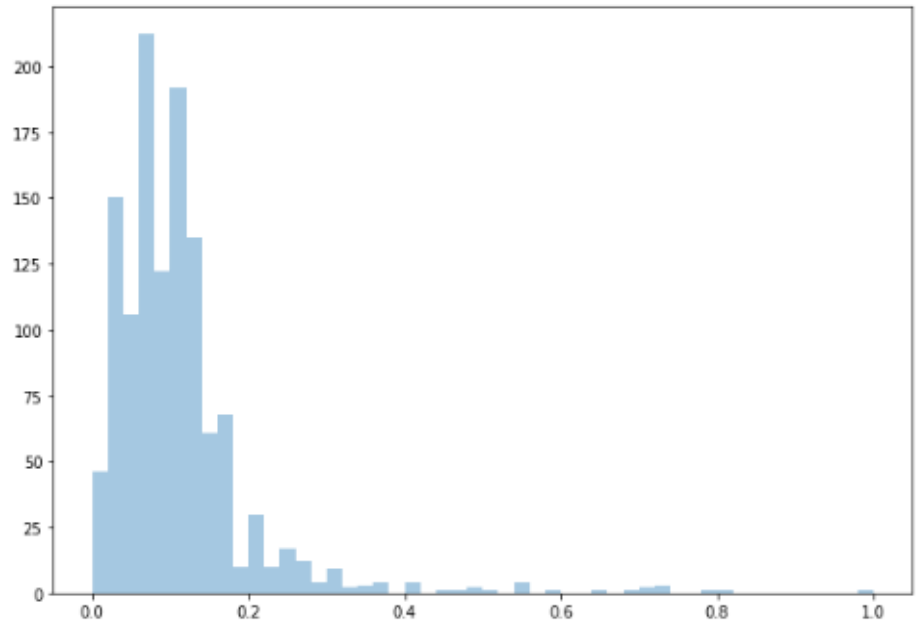

Рис 3.7 – Розподіл продажів після масштабування

	onpromotion	dcoilwtico	holiday	day_item_purchased	month_item_purchased	quarter_item_purchased	week_item_purchased	year_item_purchased
count	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000	1217.000000
mean	0.042728	0.569038	0.086278	0.486661	0.450288	0.443166	0.444267	0.441523
std	0.202326	0.318082	0.280889	0.294560	0.297704	0.357009	0.274843	0.358277
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.248490	0.000000	0.233333	0.181818	0.000000	0.211538	0.000000
50%	0.000000	0.766197	0.000000	0.500000	0.454545	0.333333	0.423077	0.333333
75%	0.000000	0.849816	0.000000	0.733333	0.636364	0.666667	0.653846	0.666667
max	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Рис 3.8 – Опис вхідних даних після масштабування

3.3. Тестування гібридної моделі

Параметри, з якими якими тестувалась розробрена гібридна модель наведені в таблиці 3.7. Ці параметри були підібрані для кожної моделі 1 рівня окремо методом підбору для зменшення втрат кожної моделі.

Табл 3.7 – Фінальні параметри гібридної моделі

Параметр	**Значення**
ridge__alpha	0.05
randomforestregressor__n_estimators	120
randomforestregressor__max_depth	150
randomforestregressor__min_samples_leaf	1
randomforestregressor__min_samples_split	2
lgbmregressor__n_estimators	100
lgbmregressor__learning_rate	0.1
lgbmregressor__max_depth	-1
lgbmregressor__min_child_samples	20
lgbmregressor__min_child_weight	0.001
lgbmregressor__num_leaves	31
meta_regressor__alpha	0.2
random_state	1

Для порівняння результатів було обрано наступні моделі:

1. Наївний прогноз – найпростіший проноз. Якщо розроблена складна модель дає похибку більшу, за наївний прогноз, то це погана модель.
2. RandomForestRegressor
3. CatBoostRegressor
4. LGBMRegressor
5. XGBRegressor
6. Ridge
7. LSTM
8. SARIMA

При тестуванні моделей ми виконували прогноз продажів на рік вперед, тобто на 365 днів. На всіх інших 1217 точках даних моделі навчались.

Результати тестування наведені в таблиці 3.8. З таблиці 3.8 видно, що нам вдалось покращити прогнози моделі порівняно з еталонними нейромережевими методами. Розроблена модель точніше прогнозує дані обраного часового ряду.

Табл 3.8 – Результати прогнозування продажів на рік

Метод	RMSE	MAE	MAPE
Naive	9.00	6.53	65.79
RF	5.98	4.13	67.01
CB	5.64	3.97	69.70
LGBM	5.85	4.11	70.67
XGBM	6.38	4.43	70.94
RIDGE	6.25	4.01	62.92
LSTM	5.41	3.84	68.51
ARIMA	6.26	4.37	87.01
Гібридна модель	4.98	3.58	61.65

При порівнянні на графіку (рис 3.9) реальних даних з прогнозованими на тестовому датасеті, побачимо, що розроблена модель добре узагальнює дані.

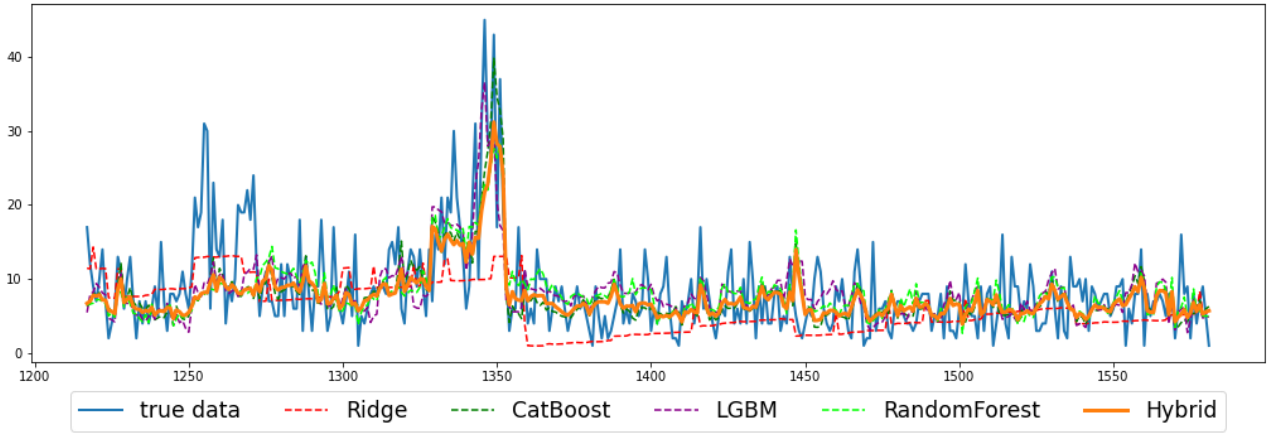

Рис 3.9 – Результат прогнозування розробленої моделі на тестових даних

Якщо ми подивимось на залишки прогнозування (рис 3.10), побачимо що в першій половині року залишки в основному додатні, це означає що в даний період часу ми отримаємо занижений прогноз продажів. В нормі цей графік має бути стаціонарним, з однаковою дисперсією на всьому періоді. Така поведінка в першій

половині року може означати, що ми включили не всі важливі фактори в нашу модель і модель ще може бути покращеною.

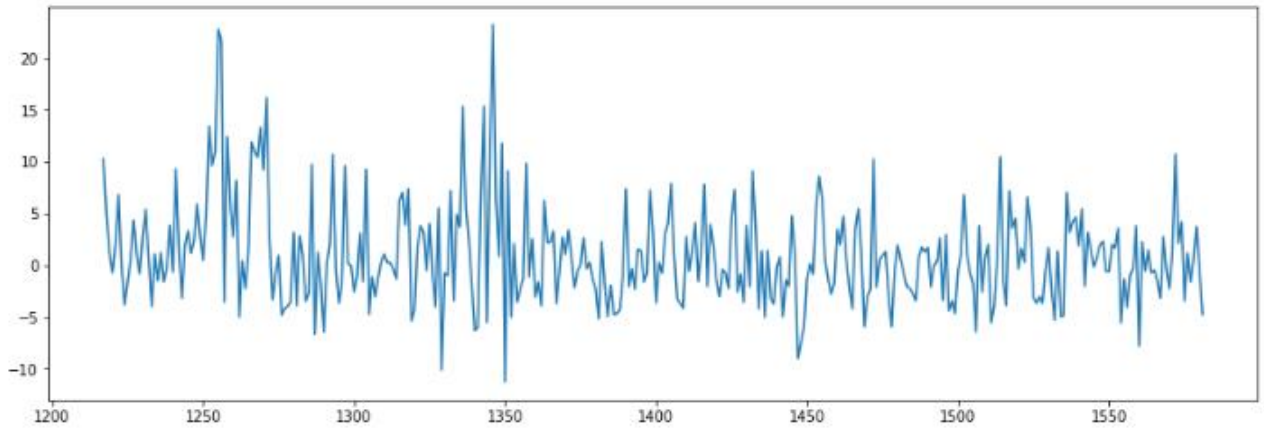

Рис 3.10 – Графік залишків прогнозування

Відхилення прогнозів від фактичних значень в об’ємах продажів за рік складає -11%. Компанії, які надають послуги прогнозування для бізнесів, обіцяють точність від 10% до 8%, тому щоб мати змогу конкурувати з такими компаніями, слід ще покращити точність моделі. Це можна зробити додавши більше факторів до нашої моделі, наприклад про продажі конкурентів, про рекламу даного товару в різних засобах масової інформації.

Недоліком розробленого методу є час навчання моделі. Тестування моделі виявило, що час навчання нашої моделі перевищує час навчання стандартних методів машинного навчання для цієї задачі.

ВИСНОВКИ ДО РОЗДІЛУ 3

У рамках даного розділу було проаналізовано дані продажів товару в магазині мережі Favorita в місті Кіто, столиці Еквадору, виконано попередню обробку вхідних даних, проведено тестування запропонованої моделі та порівняно її ефективність з стандартними нейромережевими методами.

При підготовці вхідного датасету було проведено відбір даних, заповнено пропущені значення, виконано масштабування даних.

У ході тестування було виконано прогноз продажів товару на 1 рік (365 днів) вперед. Тестування показало, що розроблена модель виконує прогнози краще, за стандартні методи. Проте, виходячи з проведеного аналізу залишків, модель можна покращити додавши більше важливих факторів, наприклад дані про рекламу даного продукту в засобах масової інформації.

Недоліком гібридної моделі є те, що її тренування займає більше часу, порівняно з стандартними методами. Це пояснюється тим, що і архітектура розробленої моделі складніша.

Отримані прогнози доцільно використовувати для планування маркетингової активності компанії та розробки стратегії просування продукту. Розробка подібних моделей дає можливість компаніям отримувати короткострокові прогнози об’ємів продажів, які необхідні для планування управлінської діяльності.

РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЕКТУ CRM-СИСТЕМИ ДЛЯ ПРОДАЖІВ

В розділі виконується розробка стартап-проекту CRM-системи для ведення продажів.

4.1. Опис ідеї проекту

Суть проекту: створення CRM-системи для ведення продажів

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	**Напрямки застосування**	**Вигоди до користувача**
CRM-система для ведення бізнесу.	1. Ведення обліку заяв та угод	Інформація про всі продажі і угоди зведена в одному місці згрупована по етапах, які налаштовуються користувачем.
	2. Автоматизація бізнес-процесів	Можливість автоматизації задач = менше витрати на робітників, робітники зайняті важливими нерутинними задачами. Програма сама може відправити листа клієнту, змінити статус угоди, поставити задачу.
	3. Генерація звітів.	Відображення графіків, діаграм за будь-якими бізнес-показниками.
	4. Аналітика, прогнозування попиту на товар	Виявлення тренду, продажів, прогнозування попиту на товари/послуги на основі історичних даних. Маючи такий прогноз і правильно ним скориставшись, компанія може підвищити свої прибутки.
	5. База клієнтів	Збереження бази клієнтів в одному місці.

Ми провели аналіз потенційних техніко-економічних переваг ідеї (чим відрізняється від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів:

- визначили перелік техніко-економічних властивостей та характеристик ідеї;
- визначили попереднє коло конкурентів, що вже існують на ринку, та провели збір інформації щодо значень техніко-економічних показників для ідеї власного проекту та проектів-конкурентів відповідно до визначеного переліку;
- провели порівняльний аналіз показників: визначили показники, що мають а) гірші значення (W, слабкі); б) аналогічні (N, нейтральні) значення; в) кращі значення (S, сильні) (табл. 4.2).

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні характеристики ідеї	(потенційні) товари/ концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Sales force	Бітрікс 24			
1	Ціна	$120	$300	$160	-	-	+
2	Масштабованість	Так	Так	Так	-	+	-
3	Функція пронозування	Так	Так	Так	-	+	-
4	Точність прогнозування	5%	10%	9%	-	-	+

Визначений перелік слабких, сильних та нейтральних характеристик та властивостей ідеї потенційного товару є підґрунтям для формування його конкурентоспроможності. Сильними сторонами розробленого проекту є його ціна дешевша від конкурентів та точність виконання прогнозів.

4.2. Технологічний аудит ідеї проекту

Проведено технологічний аудит ідеї проекту. В таблиці 4.3 наведено технологічну здійсненність ідеї проекту.

Обрана технологія реалізації ідеї проекту є вдосконалення архітектури моделі для прогнозування даних, запобігання перенавчанню, покращення пайплайну. Всі ці технології реалізуються програмними засобами.

Таблиця 4.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Покращити якість прогнозування продажів в CRM за допомогою розробленого методу, створити власну CRM.	- покращення пайплайн для обробки вхідних даних - покращення архітектури моделі для прогнозування - запобігання перенавчанню	Наявні	Доступні
Обрана технологія реалізації ідеї проекту: вдосконалення архітектури моделі для прогнозування даних, запобігання перенавчанню, покращення пайплайну.				

4.3. Аналіз ринкових можливостей запуску стартап-проекту

Визначимо ринкові можливості, які можна використати під час ринкового впровадження проекту, та ринкові загрози, які можуть перешкодити реалізації проекту. Це дозволить спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (табл. 4.4).

Таблиця 4.4 – Попередня характеристика потенційного ринку стартап-проекту

№	**Показники стану ринку (найменування)**	**Характеристика**
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	4000грн*500од = 2000000грн (в місяць)
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Відсуні
5	Специфічні вимоги до стандартизації та сертифікації	Читабельний шрифт Зручний інтерфейс Приємна кольорова гама Дизайнерське рішення для інтерфейсу
6	Середня норма рентабельності в галузі (або по ринку), %	25%

Банківський відсоток на вкладення складає 7-10% річних. Середня норма рентабельності складає 25%. Отже проект є вигідним у фінансовому плані. На ринку присутні багато продуктів, проте їх якість відрізняється. І загальні рішення для певної галузі бізнесу можуть бути незручними. Тому створення нішевого продукту, який буде кращим за рішення конкурентів є хорошою бізнес-ідеєю. Ринок є привабливим для входу за попередніми оцінками.

Ми визначимо потенційні групи клієнтів, їх характеристики, та сформували орієнтовний перелік вимог до товару для кожної групи (табл. 4.5).

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№	**Потреба, що формує ринок**	**Цільова аудиторія**	**Відмінності у поведінці різних потенційних цільових груп клієнтів**	**Вимоги споживачів до товару**
1	Автоматизована система для спрощення ведення бізнес-процесів, адміністрування ресурсами і робочою силою, аналітики великого бізнесу.	Власники великих, середніх бізнесів, ТОП-менеджери компаній	Частіше працюватти з CRM буде не сам власник чи ТОП-менеджер, а велика кількість найманих робітників.	Безперебійна та коректна робота продукту. Інтуїтивно зрозумілий інтерфейс. Кастомізація продукту під бізнес замовника. Навчання нових співробітників.
2	Підвищення продажів, пришвидшення росту компанії, розподіл робочих задач	Власники малих чи сімейних бізнесів	Малий штат співробітників. Власник сам виконує багато функцій: від виготовлення товару, доставки, до розпоряджання бюджетом.	Інтуїтивно зрозумілий інтерфейс.

Проведемо аналіз ринкового середовища: складемо таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (табл. 4.6-4.7). Фактори в таблиці подані в порядку зменшення значущості.

Таблиця 4.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Загострення конкурентної боротьби	Загострення конкурентної боротьби в цільовому сегменті	Запропонувати більш досконалий продукт
2	Загострення функціональної конкуренції	Можливість морального старіння даного товару внаслідок загострення функціональної конкуренції	Запропонувати більш досконалий продукт
3	Недостатня рекламна компанія	Недостатнє розповсюдження реклами та недостатнє зацікавлення цільової аудиторії	Вдосконалити власну рекламну компанію
4	Неякісний продукт	Неякісно створений продукт внаслідок відсутності досвіду управління компаніями у засновників	Консультуватися з інвесторами щодо прийняття важливих рішень

Таблиця 4.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Імідж	Завоювання позитивного іміджу	Матеріальні витрати на вдосконалення власного іміджу, на комунікацію з цільовою аудиторією
2	Інформаційна прозорість	Через доступність до інформаційних технологій, потрібно дивитись на інших, і всі помилки одразу видно	Вчитися на помилках конкурентів, вдосконалювати свій продукт
3	Консультація з потенційними клієнтами	Можна дізнатися який функціонал користувачі хочуть бачити	Створення продукту пристосованого до користувача

Проведемо аналіз пропозиції: визначимо загальні риси конкуренції на ринку (табл 4.8).

Таблиця 4.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Тип конкуренції – чиста конкуренція.	На ринку присутні багато різних CRM.	Створення спеціалізованої CRM для певної галузі.
Національний рівень конкурентної боротьби.	Національний рівень	Додавання нових мов для опрацювання
За галузевою ознакою – внутрішньогалузева.	Конкуренція ведеться між компаніями однієї галузі	Оптимізація отримання прибутку, розширення

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
		ринків збуту, стратегія низьких витрат
Товарно-видова конкуренція за видами товарів.	Конкуренція між товарами одного виду.	Додавання нового функціоналу
За характером конкурентних переваг – нецінова	Конкуренція проводиться за рахунок покращення якості продукту	Покращення якості продукту
За інтенсивністю – не марочна	Роль торгової марки незначна	Заохочення клієнтів якістю товару

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі (табл. 4.9).

На ринку існує багато конкурентів, які пропонують загальні рішення для бізнесів. Також на вітчизняному ринку присутні закордонні аналоги, які також є загальними рішеннями. Однак спеціалізованих рішень під певні галузі є дуже мало. Тож чудовим бізнес-рішенням буде розробити вузькоспеціалізовану CRM для певної галузі підприємництва. Таким чином ми отримуємо перевагу над головними конкурентами, пропонуючи потенційному покупцю кращий, довершеніший товар для вирішення його проблеми. Окрім широкого функціоналу, система має бути простою для освоєння новими користувачами. Пропонуючи вузькоспеціалізоване рішення ми розробимо той функціонал, який безпосередньо буде необхідний для ведення бізнесу у цій галуз.

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	**Потенційні конкуренти**	**Постачальники**	**Клієнти**	**Товари-замінники**
Складові аналізу	Бітрікс 24 Salesforce Sales Creatio	Бар'єри входження в ринок: виробничі витрати на початковому етапі розробки системи	-	Фактори сили споживачів: Багатий функціонал, зрозумілий інтерфейс для пришвидшення навчання співробітників	Фактори загроз: ціна, функціонал
Висновки:	Помірна інтенсивність Конкурентної боротьби з боку прямих конкурентів	Строки виходу компанії на ринок обумовлені часом необхідним на розробку CRM системи	-	Доступна ціна, функціонал. Швидкість освоєння системи співробітниками.	Обмеження для роботи: функціонал розроблюваної CRM системи має перевищувати функціонал конкурентів

На основі проведеного аналізу конкуренції, а також із урахуванням характеристик ідеї проекту (табл. 4.2), вимог споживачів до товару (табл. 4.5) та факторів маркетингового середовища (табл. 4.6-4.7) визначимо та обґрунтуємо перелік факторів конкурентоспроможності (табл. 4.10).

Таблиця 4.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування
1	Точність прогнозування	На ринку представлені загальні рішення, які часто не відповідають вимогам підприємств в певних галузях.
2	Простота в освоєнні та використанні	Для бізнесів, особливо невеликих, важливо, щоб запропоноване рішення було простим, оскільки їм необхідно використовувати свій час на вирішення важливих бізнес питань.
3	Надійність	Система має працювати стабільно, без збоїв, оскільки вони можуть бути фатальними для користувачів.
4	Масштабованість	У міру зростання компанії зростає і база клієнтів, виникає необхідність у фільтрах та сортуванні, розбиттю на групи за якоюсь ознакою.

За визначеними факторами конкурентоспроможності (табл. 4.10) проведемо аналіз сильних та слабких сторін стартап-проекту (табл. 4.11).

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (табл. 4.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (табл. 4.11).

Перелік ринкових загроз та ринкових можливостей складений на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими

результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення.

Таблиця 4.11 – Порівняльний аналіз сильних та слабких сторін

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з стартапом						
			-3	-2	-1	0	+1	+2	+3
1	Точність прогнозування	20							+
2	Простота в освоєнні та використанні	15					+		
3	Надійність	20							+
4	Масштабованість	20							+

Таблиця 4.12 – SWOT- аналіз стартап-проекту

Сильні сторони: • Цілодобова підтримка • Інструкція по експлуатації • Дозволяє оптимізувати робочі ресурси • Автоматизація рутинних процесів • Високі аналітичні можливості • Функція прогнозування	**Слабкі сторони:** • Компанія не має репутації на ринку на початку впровадження продукту
Можливості: • Вихід на міжнародний ринок • Покращення методів прогнозування	**Загрози:** • Не є універсальним продуктом, застосовується лише в одній галузі

На основі SWOT-аналізу розробимо альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти

конкурентів, що можуть бути виведені на ринок (див. табл. 4.9, аналіз потенційних конкурентів).

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів (табл. 4.13).

Таблиця 4.13 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Соціальні мережі «Facebook», «Telegram», «Instagram»	5%	3-6 місяців
2	Власний сайт	5%	3-6 місяців
3	Власник бренду – обличчя бренду	5%	необмежений
4	Виступи з просуванням продукції на різних бізнес-конференціях	15%	необмежений
5	Моніторинг компаній в галузі та безпосередня комунікація з CEO чи ТОП-менеджером щодо впровадження продукту	20%	6-12 місяців

Для просування нашого продукту ми обираємо всі описані альтернативи ринкової поведінки.

Першим кроком будуть виступи на бізнес-конференціях та комунікація безпосередньо з власниками бізнесів та ТОП-менеджерами, щоб розповісти про продукт цільовій аудиторії. Бізнес-конференції це вдале рішення, оскільки там збирається велика частка підприємців та ТОП-менеджерів компанії, які є нашою цільовою аудиторією. З тими, хто не відвідує подібні конференції, ми будемо комунікувати напряму, пропонуючи впровадження нашого продукту. Після того, як цільова аудиторія дізнається про наш продукт, вона почна шукати інформацію про нього в інтернеті, відгуки, кейси впровадження. Така інформація буде представлена на сторінках бренду в соціальних мережах та на сайті компанії.

Для побудови власного бренду в сучасному діджиталізованому світі необхідно мати сторінки продукту в соціальних мережах. В соціальних мережах можна ділитись новинами про продукт. Наприклад результати впровадження CRM для певного бізнесу: ріст ефективності виконання запитів, ріст доходів. Також присутність в соціальних мережах і контакти з аудиторією допомагають компанії краще зрозуміти своїх користувачів і модифікувати продукти за побажаннями користувачів. Чим більше брендований контент команії поширюється мережею, тим більша впізнаваність бренду. А це означає, що якщо у певного підприємця з’явиться задача, яку вирішує наша CRM, він найперше обере її, а не конкурента.

Впровадження всіх цих альтернатив дасть нам змогу швидко просунутись на ринку та побудувати знання бренду.

4.4. Розробка ринкової стратегії проекту

Розробка ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (табл. 4.14).

Таблиця 4.14 – Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсив-ність конкурен-ції в сегменті	Простота входу у сегмент
1	Власники великих бізнесів, ТОП-менеджери	Висока	70% -середній попит	Висока	Важка, оскільки існує багато аналогів
2	Власники середніх бізнесів	Висока	70% -середній попит	Висока	Важка, оскільки існує багато аналогів
3	Власники малих бізнесів	Висока	70% -середній попит	Висока	Важка, оскільки існує багато аналогів
Які цільові групи обрано: власники бізнесів					

Для роботи в обраних сегментах ринку сформулювали базову стратегію розвитку (табл. 4.15).

Наступним кроком є вибір стратегії конкурентної поведінки (табл. 4.16).

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до продукту (див. табл. 4.5), а також в залежності від обраної базової стратегії розвитку (табл. 4.15) та стратегії конкурентної поведінки (табл. 4.16) розробили стратегію позиціонування (табл. 4.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати проект.

Таблиця 4.15 – Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Розробка CRM для малого бізнесу	Стратегія концентрованого маркетингу	Зручність та легкість у користування, не потребує важких налаштування	Стратегія спеціалізації

Таблиця 4.16 – Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопро-хідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Проект не є першопро-хідцем	Компанія буде забирати існуючих споживачів у конкурентів і шукати нових	Основні характеристики товару будуть схожими	Стратегія позиціювання

При визначенні стратегії позиціонування були обрані вимоги до товару такі, як простота використання та налаштування продукту, широкий функціонал та своєчасна підтримка продукту. Обрано базову стратегію розвитку – знизити ціни на продукцію та створити якісний товар; асоціації було обрано на базі вимог цільової аудиторії, які формують комплексну позицію проекту.

Таблиця 4.17 – Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспро-можні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
1	Ціна, якість	Знизити ціни на продукцію та створити якісний товар	Відповідна ціна, довіра до бренду	Якість, надійність, підтримка

4.5. Розробка маркетингової кампанії проекту

Першим кроком є формування маркетингової концепції товару, який отримає споживач (табл. 4.18).

Таблиця 4.18 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Оптимізація витрат часу	Автоматизація бізнес процесів	Потрібно розширити обсяг опрацьованих даних
2	Адаптивність	Версії для різних пристроїв	Зручність використання на улюбленому пристрої
3	Аналітика	Аналіз виконання KPI	Потрібно покращити аналітику порівняно з конкурентами
4	Підтримка продукту	Своєчасна підтримка клієнтів	Відповідна ціна, довіра до бренду
5	Прогнозування	Створення прогнозів попиту на товар	Висока точність прогнозів

Надалі розробимо трирівневу маркетингову модель товару (табл. 4.19).

Таблиця 4.19 – Опис трьох рівнів моделі товару

<table>
<tr><th>Рівні товару</th><th colspan="3">Сутність та складові</th></tr>
<tr><td>I. Товар за задумом</td><td colspan="3">Розробка CRM для малого бізнесу з алгоритмом, який здатен прогнозувати попит на товар.</td></tr>
<tr><td rowspan="7">II. Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>1. Зручний інтерфейс
2. Адаптованість
3. Точність прогнозування
4. Вартість обслуговування
5. Собівартість товару</td><td>М
М
Нм
М
М</td><td>Тх/Ор
Тх/Ор
Тх
Е/Тх
Тх</td></tr>
<tr><td colspan="3">Якість: Відповідає рекомендаціям по вдосконаленню сайтів комітетів Верховної Ради України</td></tr>
<tr><td colspan="3">Пакування: Електронна документація, що містить таку інформацію:
загальна назва продукту, власна назва;
найменування та адреса виробника і місце виготовлення;
товарний знак;
інструкція щодо користування.
необхідні технічні вимоги.</td></tr>
<tr><td colspan="3">Марка: MyBusiness</td></tr>
<tr><td rowspan="2">III. Товар із підкріпленням</td><td colspan="3">До продажу: адаптивний сервіс</td></tr>
<tr><td colspan="3">Після продажу: технічна підтримка</td></tr>
<tr><td colspan="4">За рахунок чого потенційний товар буде захищено від копіювання: патенту та авторського права</td></tr>
</table>

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар, яке передбачає аналіз ціни на товари-

аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 4.20). Аналіз проводиться експертним методом.

Таблиця 4.20 – Визначення меж встановлення ціни

№	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	0-70$	150-300$	>1200$	100-150$

Далі ми визначили оптимальну систему збуту, в межах якого приймається рішення (табл. 4.21):

Таблиця 4.21 – Формування системи збуту

№	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Клієнти підписуються на продукт безпосередньо у компанії-розробника	1. Встановлення контактів зі споживачами і їх підтримка; 2. Дослідницька робота зі збору маркетингової інформації	0-Канал нульового рівня (виробник безпосередньо продає товар клієнту)	Через сайт виробника

Останньою складової маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 4.22).

Таблиця 4.22 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціону-вання	Завдання рекламного повідом-лення	Концепція рекламного звернення
1	Клієнти дізнаються про товар з інтернету та на конферен-ціях	Інтернет, бізнес-конференції	Кращий функціонал порівняно з аналогами. Довіра до бренда.	1. Поширення знань про продукт 2. Проінфор-мувати про переваги продукту	1. Перелік основних правдивих даних про продукт 2. Науково-професійний стиль

Проаналізувавши специфіку поведінки цільових клієнтів, було обрано концепцію рекламного звернення:

1. Перелік основних правдивих даних про продукт,

2. Науково- професійний стиль.

Основними способами поширення реклами будуть виступи на конференціях та безпосередня комунікація з потенційними клієнтами. Як додаткові засоби комунікації було обрано соціальні мережі та власний сайт компанії. Завданням рекламного повідомлення є зацікавлення та поширення знань про продукт новим клієнтам.

ВИСНОВКИ ДО РОЗДІЛУ 4

Четвертий розділ присвячений розробці стартап-проекту для розроблюваного способу прогнозування часових рядів. Для дослідження в рамках розробки стартап-проекту було виконано наступні завдання:

1. Наведено загальний опис ідеї проекту для виходу на ринок, описаний функціонал проекту, визначено основних конкурентів та співставлено функціонал конкурентів для визначення переваг розроблюваного продукту над обраними конкурентами.
2. Проведено технічний аудит проекту і визначено можливості реалізації програмного продукту
3. Проаналізовано ринкові можливості для запуску проекту, порівняно перспективи запуску в порівнянні з конкурентами, встановлено план та ринкову стратегію розвитку продукту, визначено цільову аудиторію та способи просування проекту.
4. Розроблено маркетингову програму для просування продукту на ринку, описано способи та основні канали збуту, визначно пріорітетні цільові групи та маркетингові повідомлення для розширення клієнтської бази

Попит на продукцію є, що підтверджується позитивною динамікою ринку та потребами споживачів. Конкуренція в цій області на ринку України є, що обумовлює певні труднощі входу на ринок. Цільовою аудиторією є власники бізнесів та ТОП-менеджери. Перевагою проекту для ЦА є те, що запропонований продукт буде будувати кращі прогнози для бізнесу, що дозволить клієнтам підвищити свої доходи.

Так як комунікувати безпосредньо з власниками бізнесів і ТОП-менеджментом важко, було вирішено, що доцільним шляхом розповсюдження є представлення продукту на бізнес-конференціях, а також створення сторінок бренду в соціальних мережах і сайту компанії.

ЗАГАЛЬНІ ВИСНОВКИ

Магістерська робота присвячена вирішенню проблеми прогнозування часових рядів продажів товарів.

Було проведено теоретичний огляд та аналіз існуючих методів прогнозування часових рядів, описано їх основні недоліки і переваги. Проведене теоретичне дослідження показало, що стандартні методи прогнозування часових рядів часто не дають потрібної точності, тому, для отримання більш точних прогнозів, використовуються гібридні методи, суть яких полягає у поєднанні стандартних методів у більш складні архітектури. Проте немає універсальних рекомендацій щодо того які стандартні методи, скільки та яким способом їх слід об'єднувати для отримання необхідної точності прогнозів. До того ж постійно з'являються нові методи, роботу яких слід досліджувати.

Було запропоновано власний варіант гібридної моделі для прогнозування часових рядів, а також аргументовано вибір архітектури моделі. Запропонована модель має архітектуру стека, яка складається з 4 моделей – CatBoost, Ridge, RandomForest, LGBM – на першому рівні та материгресора Ridge на другому. Дана архітектура направлена на підвищення точності прогнозування моделі завдяки поєднанню гетерогенних моделей на першому рівні стеку. Поєднання обраних моделей збільшить різноманітність і взаємодоповнюваність базових предикторів шляхом поєднання набору різних регресорів для покращення ефективності регресії. Для розробленої моделі методом підбору було визначено оптимальні гіперпараметри.

Виконано тестування запропонованої гібридної моделі на часовому ряді продажів товару в магазині мережі Favorita в місті Кіто, столиці Еквадору. Тестування показало, що розроблена модель виконує прогнози краще, за стандартні методи. Виходячи з проведеного аналізу залишків, модель можна покращити додавши більше важливих факторів в датасет. Тренування моделі займає більше часу, порівняно з стандартними методами, оскільки і архітектура розробленої моделі складніша.

Запропонована гібридна модель дозволяє отримувати досить точні прогнозні оцінки для прийняття управлінських рішень. Отримані прогнози доцільно використовувати для планування маркетингової активності компанії та розробки стратегії просування продукту.

Завершальним етапом магістерської роботи була розробка стартап-проекту CRM-системи для продажів, яка матиме функціонал для прогнозування часових рядів з використанням розробленого методу. В підсумку, було отримано стартап-проект для запуску програмного продукту на ринок, отримано навички створення стартап-проектів, побудови маркетингової стратегії та аналізу обраного ринку.

ЛІТЕРАТУРА

1. Christopher Chatfield. The Analysis of Time Series: Theory and Practice / Christopher Chatfield. – New York: Springer-Science+Business Media, B.V., 2013. – 263 с.
2. Peter J. Brockwell. Time Series: Theory and Methods / Peter J. Brockwell, Richard A. Davis. – New York: Springer-Science+Business Media, 2009. – 580 с.
3. Дауб І. С. Исследование статистических методов прогнозирования временных рядов с трендом и сезонностью / Ігор Сергійович Дауб. // Научно-образовательный журнал для студентов и преподавателей «StudNet». – 2021. – №5.
4. Rob J Hyndman. Forecasting: principles and practice / Rob J Hyndman, George Athanasopoulos. – Онлайн видавництво OTexts.com, 2018. – 380 с.
5. Scikit-Learn User Guide – Linear Models [Електронний ресурс]– 2021. – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression.
6. Antoni Wibowo. Food Price Prediction Using Time Series Linear Ridge Regression with The Best Damping Factor / Antoni Wibowo, Inten Yasmina. // «Advances in Science», Technology and Engineering Systems Journal. – 2021. – Vol.6 №2. – С. 694–698.
7. LASSO Regression and Its Application in Forecasting Macro Economic Indicators: A Study on Vietnam’s Exports / Pham Hoang Uyen, Vo Thi Le Uyen, Le Thanh Hoa, Trinh Quoc Trung // Prediction and Causality in Econometrics and Related Topics. ECONVN 2021. Studies in Computational Intelligence / Pham Hoang Uyen, Vo Thi Le Uyen, Le Thanh Hoa, Trinh Quoc Trung. – Cham: Springer, 2021. – С. 575–585.
8. Павлишенко Б. Machine-Learning Models for Sales Time Series Forecasting / Богдан Павлишенко. // Рецензований онлайн журнал з відкритим доступом "Data", видавництво MDPI. – 2019. – №4.
9. Multilayer Perceptron: Architecture Optimization and Training / Hassan Ramchoun, Mohammed Amine Janati Idrissi, Youssef Ghanou, Mohamed Ettaouil

// «International Journal of Interactive Multimedia and Artificial Intelligence» / Hassan Ramchoun, Mohammed Amine Janati Idrissi, Youssef Ghanou, Mohamed Ettaouil., 2016. – Vol. 4, N°1 – С. 26–30.

10. Masini, Ricardo P. Machine Learning Advances for Time Series Forecasting [Електронний ресурс] / Masini, Ricardo P., Medeiros, Marcelo C., Mendes, Eduardo F. // eprint arXiv:2012.12802. – 2020. – Режим доступу до ресурсу: https://arxiv.org/pdf/2012.12802.pdf.
11. Recurrent Neural Network Architectures / Bianchi F.M., Maiorino E., Kampffmeyer M.C. [та ін.] // Recurrent Neural Networks for Short-Term Load Forecasting. SpringerBriefs in Computer Science. / Bianchi F.M., Maiorino E., Kampffmeyer M.C. [та ін.]. – Cham: Springer, 2017. – 10 листопада – С. 23–29.
12. Валихметова Ю. И. Применение методов машинного обучения в области прогнозирования объема продаж с учетом динамически изменяющихся признаков / Ю. И. Валихметова, Э. И. Идрисова. // Научно-образовательный журнал для студентов и преподавателей "StudNet". – 2020. – №10.
13. Z.M.Niu. Nuclear mass predictions based on Bayesian neural network approach with pairing and shell effects / Z.M.Niu, H.Z.Liang. // Physics Letters B, Elsevier B.V.. – 2018. – №778. – С. 48–53.
14. Combination of Feature Selection and CatBoost for Prediction: The First Application to the Estimation of Aboveground Biomass / Mi Luo, Yifu Wang, Yunhong Xie та ін.]. // Рецензований онлайн журнал у відкритому доступі з лісового господарства та лісової екології "Forests", видавництво MDPI. – 2021. – №12.
15. Rising Odegua. Applied Machine Learning for Supermarket Sales Prediction / Rising Odegua. // Research gate. – 2020. – №1.
16. Sales Forecasting Based on CatBoost / Jingyi Ding, Ziqing Chen, Li Xiaolong, Baoxin Lai. // 2020 2nd International Conference on Information Technology and Computer Application (ITCA). – 2020.
17. Matheus Henrique Dal MolinRibeiro. Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series /

Matheus Henrique Dal MolinRibeiro, Leandrodos Santos Coelho. // Applied Soft Computing, Elsevier B.V.. – 2020. – №88.

18. D.H. Wolpert. Stacked generalization / D.H. Wolpert. // Neural Networks. – 1992. – №5. – С. 241–259.
19. SciPy [Електронний ресурс] – Режим доступу до ресурсу: https://scipy.org/.
20. NumPy [Електронний ресурс] – Режим доступу до ресурсу: https://numpy.org/.
21. Pandas [Електронний ресурс] – Режим доступу до ресурсу: https://pandas.pydata.org/.
22. Scikit-Learn [Електронний ресурс] – Режим доступу до ресурсу: https://scikit-learn.org/stable/.
23. Seaborn [Електронний ресурс] – Режим доступу до ресурсу: https://seaborn.pydata.org/.
24. Matplotlib [Електронний ресурс] – Режим доступу до ресурсу: https://matplotlib.org/.
25. MLXtend [Електронний ресурс] – Режим доступу до ресурсу: http://rasbt.github.io/mlxtend/.
26. Corporación Favorita Grocery Sales Forecasting [Електронний ресурс] – режим доступу: https://www.kaggle.com/c/favorita-grocery-sales-forecasting/data