

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ  
СІКОРСЬКОГО»  
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ  
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису  
УДК 519.688

До захисту допущено  
В. О. завідувача кафедри ММСА

О.Л.Тимошук  
« \_\_\_ » \_\_\_\_\_ 2020 р.

## Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз  
на тему: «Використання темпоральних згорткових нейронних мереж в  
системах прогнозування продажів»

Виконав:  
студент II курсу, групи КА-92мп  
Степаненко Євгеній Юрійович

\_\_\_\_\_

Керівник:  
доцент кафедри ММСА,  
к.т.н, доц. Жиров О.Л.

\_\_\_\_\_

Рецензент:  
доцент кафедри системного проектування  
КПІ ім. І. Сікорського  
к.т.н., доц. Кисельов Г.Д.

\_\_\_\_\_

Засвідчую, що у цій магістерській  
дисертації немає запозичень з праць інших

Digitized by Electronic Library of Kyiv Polytechnic Institute

Digitized by COBE

Київ

Студент \_\_\_\_\_

Київ  
2020

**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Інститут прикладного системного аналізу**  
**Кафедра математичних методів системного аналізу**

Рівень вищої освіти – другий (магістерський)  
Спеціальність (спеціалізація) – 124 «Системний аналіз» («Системний аналіз фінансового ринку»)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
\_\_\_\_\_ О.Л. Тимощук  
«\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на магістерську дисертацію студенту**  
**Степаненку Євгенію Юрійовичу**

1. **Тема дисертації** «Використання темпоральних згорткових нейронних мереж в системах прогнозування продажів», науковий керівник дисертації Жиров Олександр Леонідович, кандидат технічних наук, професор, затверджені наказом по університету від «02» листопада 2020 р. № 3182-с
2. **Термін подання студентом дисертації** 14 грудня 2020
3. **Об'єкт дослідження:** Статистичні дані про обсяг продажів магазинів роздрібною торгівлі
4. **Предмет дослідження:** Темпоральні згорткові мережі в задачах прогнозування продажів
5. **Перелік завдань, які потрібно розробити:** Проаналізувати предметну область, та на її основі запропонувати методи прогнозування. Використовуючи обраний метод, розробити програмний модуль для прогнозування. Зробити висновок.
6. **Орієнтовний перелік графічного (ілюстративного) матеріалу:**
  - 1) Таблиці
  - 2) Блок схеми архітектури нейронних мереж
  - 3) Графіки
7. **Дата видачі завдання:** 1 вересня 2019 року

## Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів роботи	Примітка
1	Формулювання тематики (напрямку) дослідження.	07.09.2019 – 30.09.2019	
2	Аналіз актуальності задач стосовно тематики дослідження	01.10.2019 – 30.10.2019	
3	Аналіз відомих результатів стосовно тематики дослідження	01.11.2019 – 30.11.2019	
4	Формулювання задач дослідження	01.12.2019 – 30.12.2019	
5	Уточнення (формулювання) теми дисертації	25.02.2020–28.02.2020	
6	Вибір методів виконання досліджень	01.03.2020 – 30.03.2020	
7	Виконання теоретичного дослідження	01.04.2020 – 30.05.2020	
8	Збір статичних даних, попередній аналіз даних, підготовка до моделювання	01.06.2020 – 30.08.2020	
9	Розробка (вибір) програмного продукту для виконання обчислювальних експериментів	01.09.2020 – 11.10.2020	
10	Виконання обчислювальних експериментів, аналіз та оформлення результатів	11.10.2020 – 10.11.2020	
11	Оформлення пояснювальної записки у цілому	11.11.2020 – 21.12.2020	
12	Захист дисертації	21.12.2020 – 23.12.2020	

Студент

Степаненко Є.Ю.

Науковий керівник дисертації

Жиров О.Л.

## РЕФЕРАТ

Магістерська дисертація: 74 с., 22 табл., 14 рис. 22 джерел

Об'єкт дослідження – дані продажів магазинів роздрібної торгівлі

Предмет дослідження – системи прогнозування продажів з застосуванням нейронних мереж

Мета роботи – дослідити точність згорткових мереж в системах прогнозування продажів.

Метод дослідження – побудова прогнозних моделей.

Актуальність роботи - створення програмного модуля, що дозволить робити прогнози продажів магазинів роздрібної торгівлі.

Програмний продукт реалізовано за допомогою мови програмування Python, на базі фреймворку Tensorflow. Обробка даних виконується за допомогою бібліотеки pandas

Отримані результати – розроблено систему прогнозування продажів для магазинів роздрібної торгівлі та інтернет магазинів. Проведено порівняльний аналіз мереж LSTM та TCN

ЗГОРТКОВІ ЕЙРОННІ МЕРЕЖІ, РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ, ЧАСОВІ РЯДИ, МОДЕЛІ, ПРОДАЖІ, ПРОГНОЗУВАННЯ, PYTHON.

## ABSTRACT

Theme: “Using temporal convolutional neural networks for sales forecasting systems”

Thesis:74 pages, 22 tables, 14 picture, 22 cited sources.

Object of the study – retail and e-commerce sales data.

Subject of research – sales forecasting systems.

The purpose of the work is to develop a system for analysis and forecasting of e-commerce and retail sales using neural networks.

The method of research – development of forecasting models.

Actuality is to work is to create software module that will allow you to make sales forecasting for pricing decisions.

Application is implemented using the Python programming language, based on the Tensorflow framework. Data preprocessing is executed using pandas library

Obtained results - the information-analytical system for sales modeling and forecasting for e-commerce. LSTM and TCN architectures comparison analysis.

CONVOLUTIONAL NEURAL NETWORK, RECURRENT NEURAL NETWORK, TIME SERIES, MODELS, SALES, FORECASTING, PYTHON

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	8
ВСТУП.....	9
1 ОГЛЯД ІСНУЮЧОГО СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ .....	10
1.1 Актуальність задачі створення системи аналізу продажів товарів.....	10
1.2 Машинне навчання. ....	14
1.2.1 Типи та класифікація машинного навчання. ....	16
1.2.2 Переваги та недоліки. ....	17
1.3. Нейронні мережі. ....	19
1.4. Формалізація та постановка задачі. ....	20
Висновки до розділу.....	22
РОЗДІЛ 2 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ДО ЗАДАЧІ ПРОГНОЗУВАННЯ .....	23
2.1 Прогнозування продажів як задача аналізу часових рядів.....	23
2.1.1 Моделі з авторегресією та ковзним середнім.....	23
2.2 Нейромережеві підходи до задач прогнозування.....	25
2.3 Темпоральні згорткові мережі для задач прогнозування .....	33
2.3.1 Архітектура згорткових темпоральних мереж.....	36
2.3.2 Розширені згортки .....	38
Висновки до розділу.....	44
РОЗДІЛ 3 АРХІТЕКТУРА ТА АНАЛІЗ РЕЗУЛЬТАТІВ РОБОТИ .....	45
3.1 Вибір платформи та мови реалізації.....	45
3.2 Структура та попередня обробка даних .....	45
3.3 Дослідження ефективності роботи запропонованих методів. ....	47
Висновки до розділу.....	53
РОЗДІЛ 4 СТАРТАП АНАЛІЗ ПРОЕКТУ .....	54
4.1 Інформаційна карта проекту .....	54
4.2 Технологічний аудит ідеї проекту .....	56
4.3 Аналіз ринкових можливостей запуску стартап-проекту .....	59
4.4 Розроблення ринкової стратегії проекту .....	66
4.5 Розроблення маркетингової програми стартап-проекту .....	69

4.6 Висновки до розділу.....	70
ВИСНОВКИ .....	71
ПЕРЕЛІК ПОСИЛАНЬ .....	72

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

TCN – temporal convolutional network

RNN – recurrent neural network

НН – нейронна мережа

СППР – система підтримки прийняття рішень



## ВСТУП

Будь-який вид прогнозування - це складне завдання, особливо там, де майбутнє залежить від багатьох обставин. Продажі в електронній комерції є дуже мінливими і непередбачуваними за своєю природою. Агенти роздрібною торгівлі часто ризикують при ціноутворенні, бажаючи отримати високі рівні продажів. Існує багато факторів, які впливають на продаж, такі як попит та пропозиція, тенденції на ринку, глобальна економіка, корпоративні результати, історична ціна, суспільні настрої, маркетингові компанії, популярність (хороша чи погана репутація компанії), все це може призвести до збільшення або зменшення кількості покупців тощо. Хоча можна проаналізувати безліч факторів, все-таки важко досягти кращих показників прогнозування продажів та передбачити оптимальну ціну.

Прогнозування продажів на конкретний товар на один період вперед - саме по собі дуже складне завдання. У цьому дослідженні продажі товарів прогноуються на наступний тиждень. На кожен тиждень проводяться порівняння з фактичними цінами для підтвердження моделі. У цьому дослідженні дано відповіді на два питання нижче:

1. Як ми можемо прогнозувати на тиждень вперед, використовуючи лише історичні дані товарів?
2. Як можна перевірити результати для розробленої моделі?

У цьому дослідженні згортова темпоральна нейронна мережа використовується як технологія машинного навчання для аналізу та прогнозування майбутніх продажів товарів на основі історичних даних.

# 1 ОГЛЯД ІСНУЮЧОГО СТАНУ ПРОБЛЕМНОЇ ОБЛАСТІ

## 1.1 Актуальність задачі створення системи аналізу продажів товарів

Поява та масове розповсюдження мережі Інтернет створили новий напрям у сфері надання послуг та спричинили його стрімкий розвиток - продажів товарів та послуг через мережу Інтернет. Це явище відоме в західних країнах як «e-commerce» (або «електронна комерція»), такий термін також використовується в сучасній літературі.

Електронна комерція (від англ. e-commerce) — це сфера цифрової економіки, що включає всі фінансові та торгові транзакції, які проводяться через Інтернет, за участю комп'ютерних мереж. Тобто бізнес-процеси, що пов'язані з проведенням транзакцій.[1]

Особливе місце в електронній комерції займає функціонування інтернет магазинів та маркетплейсів – агрегаторів, що забезпечують порівняння цінових пропозицій у різних магазинах та надають інформацію про них.

Інтернет магазин (від англ. online shop или e-shop) — сайт з продажів товарів, що працює через мережу Інтернет. Це надає можливість налагоджувати та розвивати бізнес процеси онлайн, використовуючи будь який агент користувача (мобільний додаток або браузер). Найпопулярнішою тенденцією на 2020 рік є мобільний додатки, за допомогою яких можна за лічені хвилини сформувані замовлення товарів або послуг, вибравши один з кількох способів доставки та оплати замовлення. Продаж товарів при цьому здійснюється дистанційно і це накладає певні обмеження на товари.

Не дивлячись на наявність онлайн-консультантів, електронні покупки не передбачають можливостей контактної оцінки товару та не надають можливість об'єктивно оцінити його зовнішній вигляд. [2]

Інтернет магазини збільшують відсоток транзакцій, по відношенню до загальних продажів. Так у [3, 4] показано, що за період з 1999 року по 2009 рік кількість продажів збільшилося майже в 6 разів. Це зумовлює зацікавленість інвесторів до даного бізнесу й викликає суттєвий приріст гравців на ринку, відповідно до конкуренції. Особливо актуальним явище електронної комерції є на даний момент, коли оффлайн-взаємодія покупців та продавців регулюється карантинними обмеженнями.

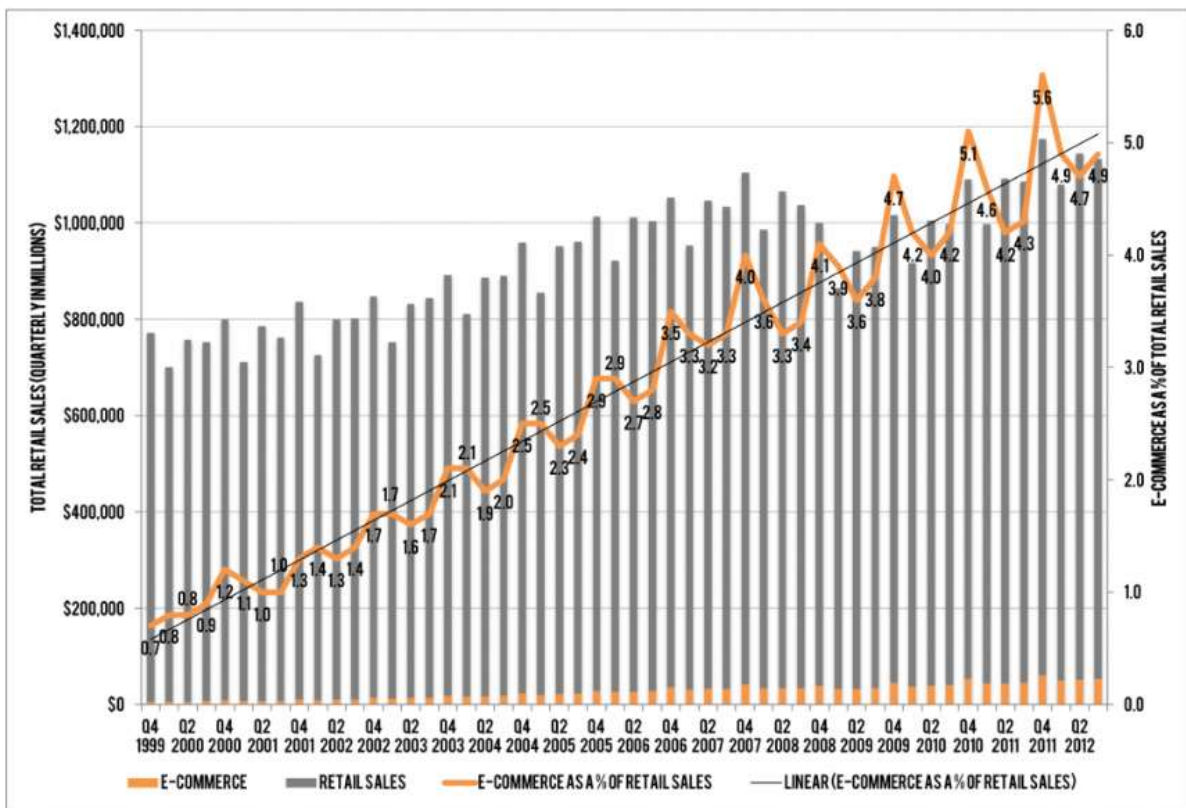


Рисунок 1.1 - Рівень продажів електронної комерції й відсоток продажів в електронній комерції, від загальних продажів

Оскільки продажі в мережі інтернет були доволі вдалим, активна робота інтернет-магазинів призвела до появи нового виду магазинів, такого як маркетплейси. Американська компанія Амазон застосувала їх як підхід першою

у світі, і зараз є основним технологічним гігантом. Застосування передових технічних практик дозволило їх розповсюдити свою сферу впливу на інші напрями мережевого бізнесу, такі як хостинг та надання послуг дата-центрів. Прикладом українських маркетплейсів можна назвати такі сервіси як Hotline та Ekatalog.

Онлайн маркетплейс (або онлайн e-commerce маркетплейс) – це вид сайту електронної комерції, з великою кількістю товарів та послуг, що керуються та корегуються продавцями, які виступають третьою стороною. При цьому, всі транзакції здійснюються самим маркетплейсом. Інтернет маркетплейси є основним типом багатоканальної електронної комерції. Вони характеризуються як «простий і зручний портал» для оптимізації виробничого процесу. [5]

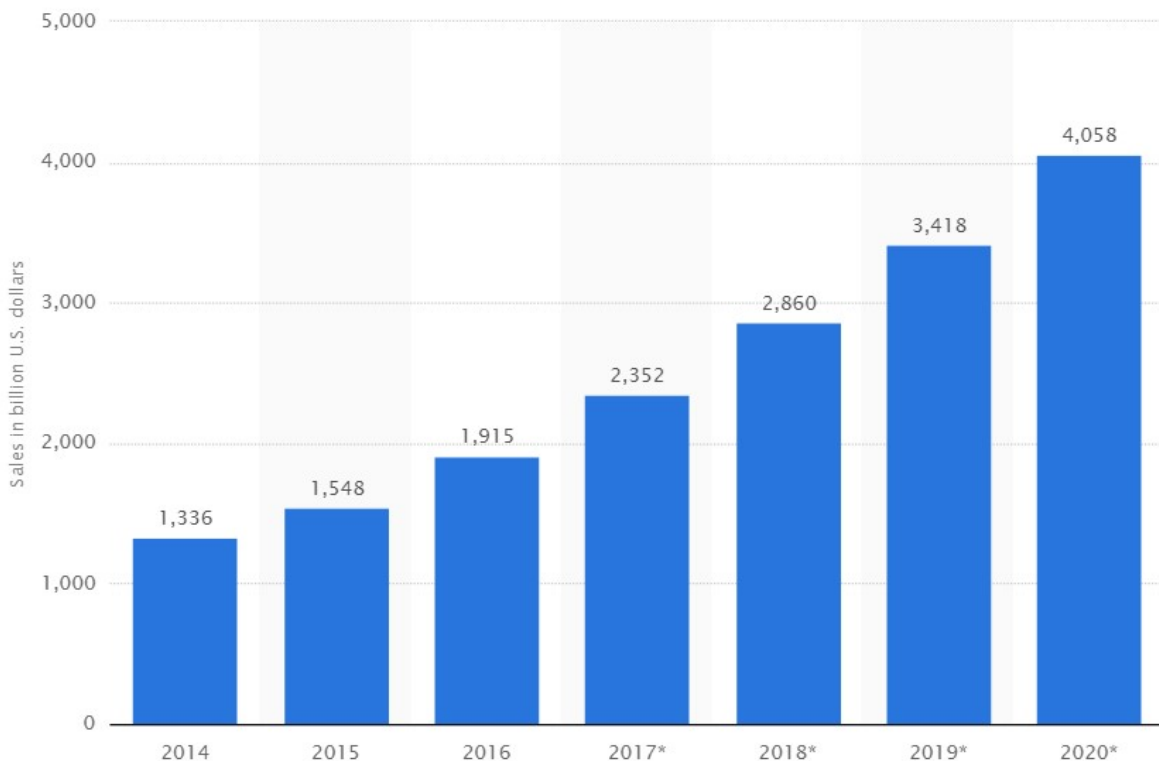


Рисунок 1.2 - Прогнозована кількість продажів в електронній комерції ринку США

В маркетплейсах процеси замовлення товару опрацьовуються оператором торгового майданчика, а потім передаються учасникам роздрібної та оптової

торгівлі. Окрім опосередкованої агрегації, маркетплейси можуть додатково надавати тікі послуги, як:

- аукціони;
- каталоги;
- замовлення;
- реклама.

Веб-ресурс, організований як маркетплейс, надає продавцям можливість реєструвати та продавати окремі елементи (товарні позиції та набори товарів) на умовах «пост-продажної» сплати.

Розвиток інформаційних технологій в усьому світі відбувається дуже швидко та призводить до того, що Інтернет використовується майже половиною людства, але не тільки для задач повсякденного життя, а й для розвитку свого бізнесу. Саме такий стрімкий розвиток призвів до формування важливого та необхідного на даний момент напрямку економіки – електронній комерції.

Електронна комерція – це сфера цифрової економіки, яка включає в себе всі фінансові і торговельні трансакції, які здійснюються за допомогою комп'ютерних мереж, і бізнес процеси, пов'язані з проведенням таких трансакцій [4].

До основних видів електронної комерції зараз відносять:

- електронний обмін інформацією (Electronic Data Interchange) – сюди відносять обмін діловою(комерційною)інформацією між бізнес-партнерами;
- електронний рух капіталу (Electronic Funds Transfer) – використовується для переведення коштів і для безготівкових розрахунків;

- електронну торгівлю (E-Trade) – здійснення купівлі-продажу товару у всесвітній мережі Інтернет, що охоплює всі етапи торгово-закупівельної діяльності, починаючи з його вибору і закінчуючи доставкою до пункту призначення;
- електронні гроші (E-Cash) – електронний аналог паперових грошей, за допомогою яких можна здійснювати грошові розрахунки;
- електронний маркетинг (англ.- E-Marketing) – комплекс маркетингових заходів, які направлені на просування та збільшення продажів товару за допомогою інтернет-реклами, веб-сайтів, соціальних мереж, тощо;
- електронний банкінг (англ. - E-Banking) – традиційне банківське обслуговування, яке здійснюється дистанційно за допомогою мережі Інтернет [4].

## 1.2 Машинне навчання.

Машинним навчанням називають набір засобів та методів, які застосовують для обробки та аналізу певним чином агрегованої інформації з метою виявлення закономірностей та складних функціональних залежностей, для вирішення специфічного класу задач. Прогнозування майбутнього значення або поведінки цих спостережень чи закономірностей потребує ітеративного навчання алгоритму на основі даних, що відрізняється від типових підходів до створення систем підтримки прийняття рішень. Ціль машинного навчання – програмування комп'ютерної системи на використання зразкових даних у якості минулого досвіду. Модель використовує інформацію, отриману з шаблонів цих даних для прогнозування майбутнього. До проблем машинного навчання відносять не лише задачі баз даних, а в тому числі і задачі штучного інтелекту (AI). [6]

Підходи машинного навчання дозволяють вирішувати різні проблеми, в тому числі:

- розпізнавання зображень;
- розпізнавання голосу чи мови;
- розпізнавання обличчя;
- аутентифікації біометричних показників;
- ідентифікація медичних діагнозів;
- задачі сільського господарства;
- економіки;
- проектування комп'ютерних мереж, робототехніки тощо.

Машинне навчання відкриває новий напрямок інформатики, який часто використовує статистичні прийоми, щоб дати комп'ютерам можливість «вчитися» з даними, не будучи явно запрограмованими. [7]

Схема класифікація машинного навчання представлена нижче на рисунку 1.1:

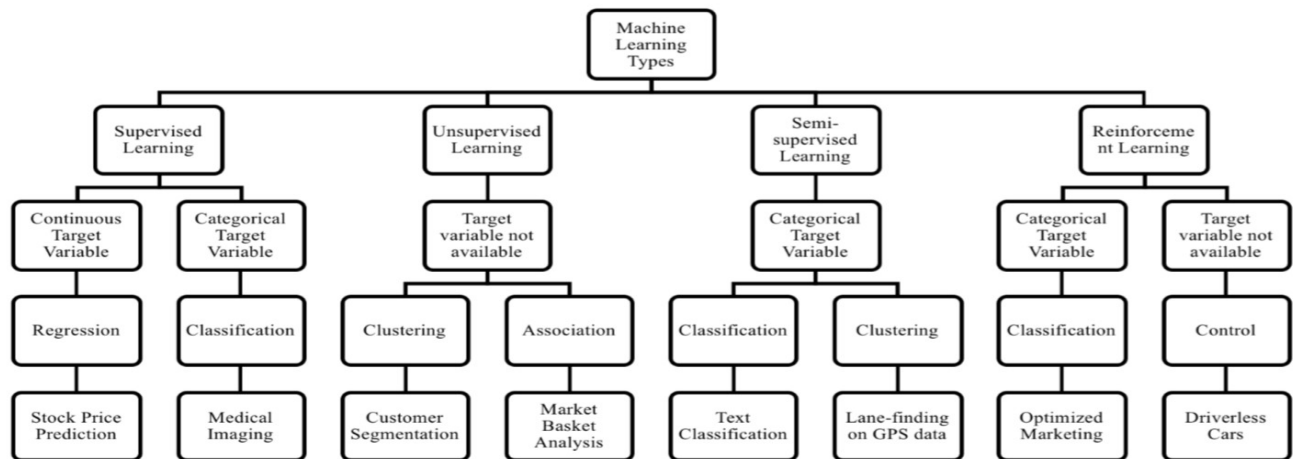


Рисунок. 1.3 - Види машинного навчання

### 1.2.1 Типи та класифікація машинного навчання.

У найзагальнішому випадку розрізняють два типу машинного навчання: навчання по прецедентах, або індуктивне навчання, і дедуктивне навчання. Оскільки останнє прийнято відносити до області експертних систем, то терміни «машинне навчання» і «навчання по прецедентах» можна вважати синонімами. Цей метод навчання зараз, як прийнято говорити, в тренді, а ось експертні системи переживають кризу. Бази знань, що лежать в їх основі, важко узгоджувати з реляційною моделлю даних, тому промислові СУБД неможливо ефективно використовувати для наповнення баз знань експертних систем.

Навчання по прецедентах, в свою чергу, поділяють на три основних типи: контрольоване навчання, або навчання з учителем (supervised learning), неконтрольоване навчання (unsupervised learning), або навчання без учителя, і навчання з підкріпленням (reinforcement learning).

Крім названих, розробляються і інші методи навчання: активне, багатозадачне, різноманітне, трансферне і т.д. Особливо успішно розвивається в останні роки «глибоке навчання», при використанні якого можуть успішно поєднуватися алгоритми навчання з вчителем і без вчителя. [8, 9]

Контрольоване машинне навчання - контрольоване навчання, що використовує теговані дані для прогнозування майбутніх даних міток або подій з певними визначеними функціями. Якщо дані мітки є безперервними, це називається проблемою регресії, а якщо дані мітки категоріальні, це проблема класифікації. Її називають бінарною класифікацією, якщо задача полягає в ідентифікації належності до класу.

Навчання без учителя - неконтрольоване навчання використовує інформацію, яка не є ні класифікованою, ні позначеною. Такі дані називають



неструктурованими. Це дозволяє групувати різні типи даних. Задачі, що вирішує навчання без учителя відносять до кластеризації або асоціації.

Напівконтрольоване машинне навчання - коли дані - це поєднання розмічених та не розмічених спостережень, переважно велика кількість неуточнених даних і менше наявних міток, тоді проблема підпадає під категорію *semi-supervised learning*. Відповідно, завдяки комбінації різних тоді система здатна значно підвищити точність навчання.

Навчання з підкріпленням полягає в ідеї взаємодії з навколишнім середовищем і працює шляхом спроб і помилок, а також дій, які приносять найбільшу винагороду.

### 1.2.2 Переваги та недоліки.

Машинне навчання має деякі переваги та недоліки. Вони описані нижче.

Переваги машинного навчання:

1. Приклади машинного навчання широко застосовуються в таких галузях, як фінансова діяльність, банківська справа, охорона здоров'я, оренда, освіта, економіка, сільське господарство тощо.
2. Машинне навчання застосовується передовими технологічними компаніями, що викладають вихідні коди розроблених ними програмних модулів у відкритий доступ, завдяки чому вони підтримуються та вдосконалюються розробниками з усього світу.
3. У динамічних чи невизначених умовах машинне навчання може обробляти багатовимірні та різноманітні дані.

4. Методологія машинного навчання дозволяє суттєво скоротити час, затрачений на вирішення задачі, на відміну від підходів розробки програмних модулів, що застосовують лише алгоритмічний підхід
5. Машинне навчання здатне забезпечити автоматизацію (часткову і для деяких випадків повну) бізнес- та ділових процесів.
6. У машинному навчанні доступні інструменти для постійного поліпшення якості у великих та складних технологічних середовищах.

#### Недоліки машинного навчання:

- 1 Препроцесинг (попередня обробка) та процес тегування (розмітки) даних – основні проблеми, оскільки дані потрібно обробляти на основі різних алгоритмів, перш ніж подавати їх на вхід до обраних моделей.
- 2 Інтерпретація результатів залишається важливою проблемою машинного навчання. Причинно-наслідкові зв'язки у прийнятті рішень алгоритмами машинного навчання часто важко пояснювати, що впливає на ступінь довіри до таких алгоритмів, та робить процес корекції алгоритмів важчим.
- 3 Часто на результат дослідження впливає те, які дії чи алгоритм було використано. Підбір необхідного алгоритму потребує високого рівня навичок спеціаліста.
- 4 Машинне навчання потребує великої кількості даних, а у випадку використання глибоких нейронних мереж – великих обсягів обчислювальних ресурсів

### 1.3. Нейронні мережі.

Комп'ютерні програми, які працюють подібно до функціонування нервової системи людини, називаються штучними нейронними мережами. Існують різні типи штучних нейронних мереж, і вони використовуються на основі математичних операцій та набору параметрів, необхідних для визначення вихідних даних. За топологією, нейронні мережі поділяють на такі: [9]:

- 1 Нейронна мережа прямого поширення.
- 2 Радіально-базисна нейронна мережа.
- 3 Самоорганізаційна карта Кохонена.
- 4 Довга короткочасна пам'ять з рекурентною нейронною мережею (RNN).
- 5 Згорткова нейронна мережа.
- 6 Модулярна нейронна мережа.

Нейромережі прямого поширення дозволяють сигналам рухатися лише в один бік: від вводу до виходу. Для таких мереж характерна відсутність зворотніх зв'язків (циклів).

Рекурентна нейронна мережа дозволяє сигналам рухатися в обох напрямках, вставляючи в мережу петлі. В ній існують зворотні зв'язки (цикли), а мережі зворотного зв'язку є потужними, динамічними і можуть бути надзвичайно складними. Їх "стан" постійно змінюється, поки вони не досягнуть рівноваги.

Істотними недоліками рекурентних мереж велика витрата часу на навчання, а також неможливість проводити навчання паралельно в рамках однієї послідовності.

#### 1.4. Формалізація та постановка задачі.

Метою роботи є побудова системи, для аналізу діяльності фірми з продажів багато асортиментних товарів, в умовах високої конкуренції, з умовами динамічного ціноутворення та аналізу асортименту товарів. Один з найважливіших етапів даної роботи – це прогнозування продажів товарів. В процесах ціноутворення важливий вплив мають системи підтримки прийняття рішень, що за допомогою методів машинного навчання здатні надавати прогноз продажів відповідно до встановленого цінового портфелю.

В рамках цього завдання необхідно:

- Побудувати систему агрегації даних.
- Зробити попередню обробку зібраних даних.
- Проаналізувати отримані числові та категорійні показники.
- Провести процедуру навчання алгоритму для задачі прогнозування даних.
- Перевірити точність прогнозу.
- Виявити товари, що мають аномальні показники та відфільтрувати набори даних.
- Створити звіт про якість отриманого прогнозу та надійність обраної моделі для СППР.

Для досягнення даних цілей необхідно вирішити наступні завдання:

- Проаналізувати існуючі підходи в задачах прогнозування.
- Виділити окремі методи, які доцільно використовувати в системах прогнозування продажів.

- Розробити алгоритм для товарних позицій.
- Адаптувати алгоритм для різних категорій товарів.
- Створити програмний модуль, на базі обраного підходу та архітектури моделі.
- Проаналізувати результати роботи програми.

Система прогнозування повинна мати високу швидкість, оскільки обчислювальна складність є вагомою для динамічного ціноутворення. Набори даних можуть містити декілька мільйонів записів, а період ціноутворення не може перевищувати декількох годин.

Основними проблемами прогнозування продажів є:

- Різні ступені агрегації даних (одна та декілька точок продажу, що агрегуються в одну групу; необхідність агрегації даних офлайн та онлайн продажів).
- Розрідженість продажів, коли кількість товарних позицій перевищує кількість продажів для певної категорії.
- Непідконтрольні події, що сприяють появі аномальних трендів продажів (як аномально високих, так і аномально низьких).

Крім того, якщо конкретні вимоги, для яких розробляється система, не зазначені, варто враховувати відсутність або наявність детального опису товару та його основних характеристик, універсальність категорійної агрегації та цінових індексів в рамках категорії).

Щоб уникнути значного обмеження, потрібно проводити навчання на вибірці, що може не містити повного асортименту. Вибірка має містити агреговані дані з різних джерел у єдиному форматі, а розроблений програмний модуль забезпечити можливість обробки даних і необхідний формат.

Вихідними даними для системи є дані, в текстовому та числовому вигляді, що описують характеристики товару та обставини за яких здійснювалися

продажі. Результатом роботи системи є прогноз кількості продажів для певного товару на зазначений період. Доцільними діями є нормалізація та підготовка, для подальшої обробки спостережень продажів товарів.

### Висновки до розділу

Підводячи підсумок проведеного дослідження щодо розділу 1, можна зробити висновок, про те, що проблеми, які виникають у сфері e-commerce, піддаються вирішенню, через застосування конкретних методів та застосування особливих процесів. Ми розглянули основну задачу, на якій ґрунтується асортиментний аналіз та автоматичне ціноутворення – прогнозування продажів. Були наведені основні вимоги щодо системи підтримки прийняття рішень для ціноутворення з прогнозуванням продажів товарів.

## РОЗДІЛ 2 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ДО ЗАДАЧІ ПРОГНОЗУВАННЯ

### 2.1 Прогнозування продажів як задача аналізу часових рядів

#### 2.1.1 Моделі з авторегресією та ковзним середнім.

На етапі визначення моделі слід визначити, чи є ряд стаціонарним, тобто змінюються чи значення часового ряду в околиці деякого фіксованого рівня. Для цього корисно подивитися на графік часового ряду і графік вибіркової автокореляційної функції. Тимчасові ряди є нестаціонарними, якщо в значеннях ряду з часом проявляється зростання або спад, а графік функції вибіркової автокореляції демонструє відсутність швидкого зникнення значущих коефіцієнтів.

Часто нестаціонарні ряди можна перетворити в стаціонарні шляхом узяття різниці. Інакше кажучи, вихідний ряд замінюється рядом різниць. У цьому випадку модель ARMA визначається для ряду різниць. По суті, в цьому випадку аналітик моделює поведінку змін значень замість поведінки їх рівня.

У деяких випадках для того, щоб отримати стаціонарні дані, необхідно знайти різниці різниць. Процедура взяття різниці виконується двічі, і в підсумку дані виходять стаціонарними.

В принципі, взяття різниць може проводитися до тих пір, поки графік даних не покаже, що ряд змінюється в деякому фіксованому рівні, а вибіркві автокореляції досить швидко обриваються. Число повторень взяття різниці, необхідних для стабілізації стаціонарного поведінки даних, позначається буквою  $d$ .

Моделі для нестаціонарних часових рядів називаються авторегресійну інтегральними моделями із середнім і позначає моделі ARIMA ( $p, d, q$ ). Тут

параметр  $p$  відповідає порядку авторегресійної частини моделі, параметр  $d$  визначає різницевий порядок, параметр  $q$  вказує порядок компоненти змінного середнього. Якщо вихідний ряд є стаціонарним, то  $d = 0$  і модель ARIMA спрощується до моделі ARMA. Тому починаючи з цього моменту запису  $(p, d, q)$  будуть позначення моделей як стаціонарних ( $d = 0$ ), так і нестаціонарних ( $d > 0$ ) часових рядів.

Незважаючи на те, що в моделях ARIMA фігурує різниці, з підбраною моделі завжди може бути зроблений прогноз безпосередньо для вихідних часових рядів. На етапі визначення моделі слід з'ясувати, чи є ряд стаціонарним, тобто змінюються чи значення часового ряду в околиці деякого фіксованого рівня. Для цього корисно подивитися на графік часового ряду і графік вибіркової автокореляційної функції. Тимчасові ряди є нестаціонарними, якщо в значеннях ряду з часом проявляється зростання або спад, а графік функції вибіркової автокореляції демонструє відсутність швидкого зникнення значущих коефіцієнтів.

Після того як буде отримано стаціонарний ряд, аналітику необхідно визначити загальні характеристики моделі, яку передбачається використовувати.

На цьому кроці виконується порівняння коефіцієнтів автокореляції та приватної автокореляції, розрахованих для даних, з теоретичними значеннями для різних моделей ARIMA. Щоб полегшити вибір відповідних моделей, на рис. А1-А3 були приведені теоретичні значення коефіцієнтів кореляції для найбільш поширених типів моделей.

Кожна модель ARIMA має власний характерний набір коефіцієнтів автокореляції та приватної автокореляції, і аналітик повинен вміти зіставляти вибірккові значення з теоретичними.

При визначенні відповідної моделі ARIMA на основі структури вибіркових функцій автокореляції та приватної автокореляції може мати місце деяка



неоднозначність. Тому початковий вибір моделі повинен розглядатися як пробний. Аналіз адекватності обраної моделі поділяється на 2 і 3 кроки. У разі її неадекватності можна спробувати застосувати іншу модель.

Якщо вибіркові автокорреляції експоненціально прагнуть до нуля, а приватні автокорреляції швидко відсікаються, то в моделі повинні бути присутніми авторегресійні складові. Якщо вибіркові автокорреляції швидко відсікаються, а приватні автокорреляції плавно прямують до нуля, то в моделі повинні бути присутніми складові змінного середнього. Якщо ж обидва графіка, як вибіркової автокорреляції, так і приватної автокорреляції, плавно прямують до нуля, то це вказує на необхідність включити в модель складові обох типів. Порядок складових MA і AR можна визначити, порівнявши кількість значущих складових в вибіркових автокорреляції і приватних автокорреляції. Для того щоб зробити висновок про значущість коефіцієнтів кореляції обох типів, їх значення зазвичай порівнюють з величиною, де  $n$  - кількість спостережень в досліджуваному часовому ряду. Таке обмеження виправдовується в тих випадках, коли  $n$  досить велике. [10]

## 2.2 Нейромережеві підходи до задач прогнозування

Основним підходом в аналізі послідовностей (sequential data) є рекурентні нейронні мережі. Рекурентними називають штучні нейронні мережі, в яких, поряд із прямими зв'язками, направленими від входів (рецепторів) мережі до її виходів (ефекторів), є зворотні, що мають протилежний напрямок. На відміну від штучних нейромереж прямого поширення, які здійснюють статичну проекцію поданих на входи векторів даних (стимулів) у вихідні вектори (реакції),

рекурентні нейромережі є динамічними системами, що оперують з послідовностями вхідних даних, перетворюючи їх на послідовності реакцій.

Поведінка рекурентних нейромереж віддзеркалює набуті при навчанні стереотипи, що робить їх близькими до цілеспрямованих адаптивних динамічних систем, запрограмованих на досягнення заздалегідь визначених цілей. Але на відміну від останніх програмування поведінки рекурентних нейромереж здійснюється шляхом навчання на прикладах, що не потребує формального визначення цілей. Вони здатні ефективно діяти в умовах невизначеності, зокрема, вирішувати задачі адаптивного керування поведінкою складних систем у нестационарному оточенні, приймати оперативні рішення в системах ситуаційного управління тощо. [11]

Будову рекурентної нейронної мережі схематично зображено на рис. 2.1. Її входам та виходам відповідають шари рецепторних та ефекторних нейронів або просто лінії передачі даних. Між цими шарами розташовано один або кілька шарів прихованих нейронів. Входи нейронів кожного шару мають прямі зв'язки з виходами нейронів попереднього шару та можуть мати зворотні зв'язки з виходами нейронів свого та наступних шарів. Зворотні зв'язки, як правило, містять елементи затримки, що надає нейромережі властивостей оперативної пам'яті. Нейрони різних шарів можуть бути однаковими або відрізнитись за типом активаційних функцій та характером нейропарадигми. Завдяки наявності затриманих зворотних зв'язків рекурентні мережі є динамічними системами, поведінка яких має зовнішню складову, що відповідає спостережуваним значенням входу та виходу, та приховану, яка характеризує внутрішній стан нейромережі. Ці складові поведінки представляють двома рівняннями, перше з яких є рівнянням вхід-вихід, а друге – рівнянням стану

$$Y(t) = F[Z(t), X(t)],$$

де  $Y(t)$ ,  $Z(t)$ ,  $X(t)$  – значення відповідно виходу, стану та входу динамічної системи в момент  $t$ .

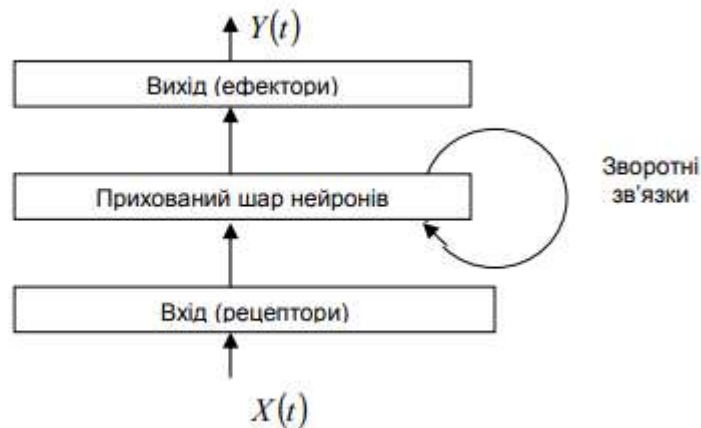


Рисунок 2.1 - Загальна структура рекурентної нейронної мережі

Поняття стану динамічної системи віддзеркалює її попередню поведінку. У випадку рекурентних нейронних мереж це поняття може мати різні значення, залежно від тривалості інтервалу спостереження  $(t_0, t]$ . При  $t - t_0 \rightarrow 0$  вектор  $Z(t)$  визначає поточний стан нейромережі як сукупність значень реакції нейронів. При  $t - t_0 \rightarrow \infty$   $Z(t)$  представляє глобальний стан, зафіксований в архітектурі нейромережі та значеннях ваги зв'язків між нейронами. Глобальний стан відповідає вмісту довготривалої пам'яті, що формується під час тренування нейронної мережі, тоді як поточний стан відповідає вмісту її оперативної пам'яті, що визначає поведінку нейронної мережі в даний момент часу. Поділ на сучасний і глобальний стан має сенс, коли процеси навчання та тестування нейронної мережі відокремлені.

Відповідно до архітектури періодичні нейронні мережі можна розділити на три типи:

- 1) відкриті рекурентні нейронні мережі;

- 2) періодичні персептрони;
- 3) ядерні рекурентні нейронні мережі.

Нейронні мережі, які не мають прихованих нейронів, відкриті. Прикладом може служити нейронна мережа Хопфілда, яка має лише один шар взаємопов'язаних нейронів. Її структуру представлено на рис. 2, де величини  $y$ ,  $s$  та  $b$  позначають відповідно входи, постсинаптичні потенціали та пороги нейронів, а  $w_{ij}$  – вагу зв'язків між нейронами. Виходи нейронів можуть мати значення  $+1$  або  $-1$ . Іноді в мережі Хопфілда замість знакової активаційної функції використовують сигмоїду. Вага зв'язків між нейронами обчислюється за формулою Хопфілда:

$$w_{ij} = N^{-1} \sum_{m=1}^M y_i^m y_j^m,$$

де  $N$  позначає число нейронів мережі, а  $M$  – кількість векторів навчальної послідовності. Ці вектори визначають стійкі стани (атрактори) нейромережі. Якщо її стан не збігається з жодним з атракторів, відбувається процес конвергенції, тобто нейромережа послідовно змінює стан, доки не досягне найближчого атрактора. Конвергенція нагадує процес асоціативного пошуку зразком, заданим у вигляді початкового стану нейромережі. Здатність до конвергенції дозволяє використовувати мережу Хопфілда як асоціативну пам'ять для відновлення збурених або частково зруйнованих даних. [12]

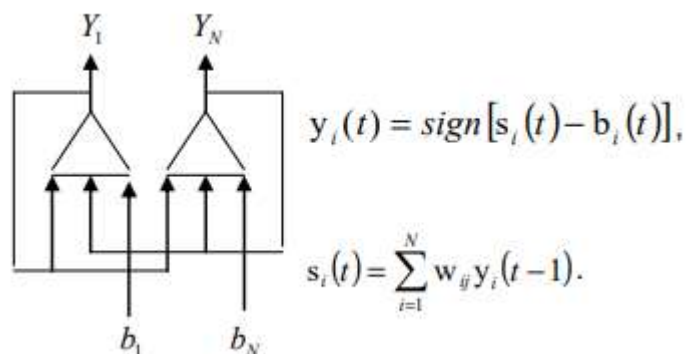
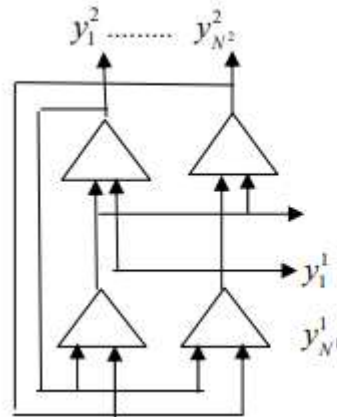


Рисунок 2.2 - нейромережа Хопфілда

На рис. 2.3 зображено двосторонню асоціативну пам'ять, що є прикладом відкритої двошарової рекурентної нейромережі. Її атракторами є пари векторів, що представляють реакції обох шарів нейронів.



$$y_i^2(t) = \text{sign} \left[ \sum_{j=1}^{N^2} w_{i,j} y_j^1(t-1) \right] =$$

$$= \text{sign} \left[ \sum_{j=1}^{N^2} w_{i,j} \text{sign} \left( \sum_{k=1}^{N^1} w_{j,k} y_k^1(t-2) \right) \right].$$

Рисунок 2.3 - Двостороння асоціативна пам'ять

Навчання відкритих нейромереж здійснюється шляхом розрахунку значень вагових коефіцієнтів на основі аналітичного рішення рівняння їхньої стабільності. Таке рішення існує за умови, що кількість атракторів не перевищує число нейронів мережі. Найбільш відомим типом рекурентної нейромережі є багатошаровий рекурентний перцептрон (RMLP-Recurrent Multi-Layer Perceptron). На рис. 2.4 дано схему рекурентного перцептрона, призначеного для апроксимації часових залежностей, який має лише один прихований шар нейронів та один ефекторний нейрон [14]. Для ефекторних нейронів мережі RMLP звичайно використовують лінійну активаційну функцію, а для прихованих – сигмоїдну. Прихований шар складається з N нейронів, на входи яких надходять затримані сигнали, що подаються на вхід мережі, затримані сигнали з виходів мережі, а також поріг  $b(1)$ . Така нейромережа здійснює відображення

$$y(t+1) = F[b^{(1)}, b^{(2)}, x(t), x(t-\tau), \dots, x(t-K\tau), y(t-\tau), \dots, y(t-L\tau)]$$

де  $\tau$  – крок затримки, K та L – кількість затримок сигналів на вході та виході

мережі.

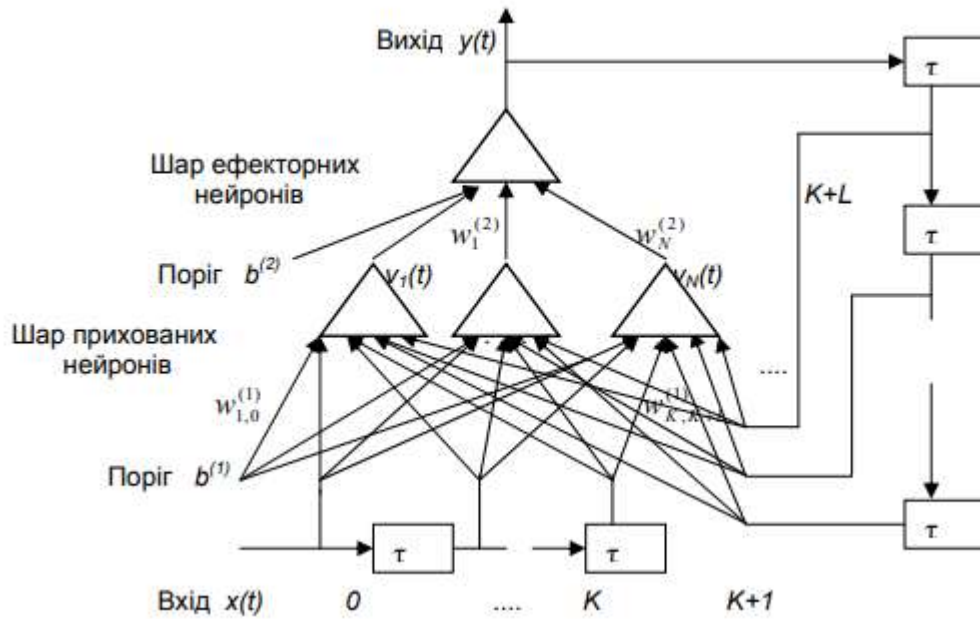


Рисунок 2.4 - Архітектура мережі RMLP

На рис. 2.5 представлено рекурентну нейромережу Ельмана [18], яка придатна для роботи з багатовимірними векторними даними і має відповідну кількість зовнішніх входів та виходів ( $N$  та  $M$ ). Щоб надати мережі необхідні динамічні властивості, у зворотні зв'язки включено елементи затримки. Зворотні зв'язки, які називають контекстним шаром, разом із зовнішніми входами утворюють вхід прихованого шару нейронів.

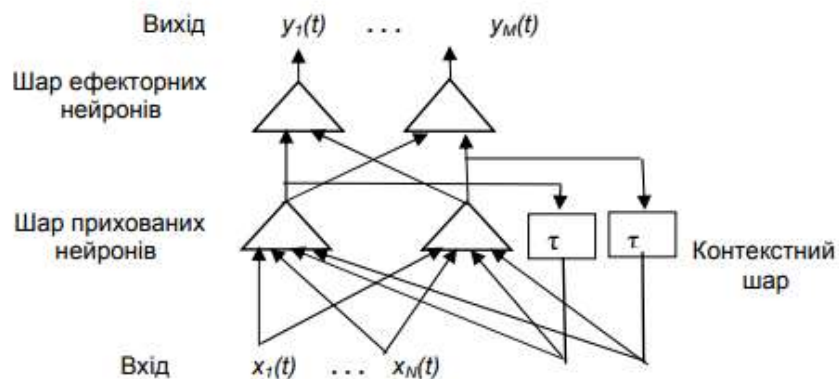


Рисунок 2.5 – Мережа Ельмана

Мережа Ельмана та RMLP є базовими моделями, що стали основою для розробки ядерних рекурентних нейромереж, в яких застосовуються різні способи перетворення вхідних даних з метою поліпшення роботи рекурентного шару нейронів. Виділяють рекурентні нейромережі, що містять шар радіально-базисних нейронів, який може самостійно виконувати функції контекстного шару або виконувати обробку даних на вході рекурентного шару. Популярним є використання методів мінімізації розмірності векторів на вході рекурентного шару за допомогою адаптивної послідовності лінійних проєкцій. Це забезпечує використання всієї наявної інформації про вирішувану задачу та характер вхідних даних для досягнення її рішення.

Зрозуміло, що мережі Ельмана потенційно придатні для прогнозування (зокрема, часових рядів). Однак відомо також, що нейронні мережі прямого поширення з успіхом справляються з цим завданням - правда, не у всіх випадках. Як приклад, пропонуємо розглянути одну з найбільш популярних варіацій завдання прогнозування - прогнозування часових рядів (ВР). Постановка завдання зводиться до вибору довільного ВР з  $N$  відліками. Далі дані поділяються на три вибірки - навчальну, тестуючу і контрольну - і подаються на вхід ІНС. Отриманий результат буде представлений у вигляді значення часового ряду в необхідний момент часу. [12]

У загальному випадку, завдання прогнозування часових рядів за допомогою ШНМ зводиться до наступної послідовності етапів:

- збір даних для навчання (етап, який вважається одним з найбільш складних);
- підготовка і нормалізація даних (приведення до виду ВР);
- вибір топології ШНМ (на цьому етапі приймається рішення про кількість шарів і наявності зворотного зв'язку);

- емпіричний (шляхом експерименту) підбір характеристик ШНМ;
  - емпіричний підбір параметрів навчання;
  - навчання ШНМ;
  - перевірка навченої моделі на адекватність поставленим завданням;
  - коригування параметрів з урахуванням попереднього кроку, остаточне навчання;
  - вербалізація ШНМ (мінімізоване опис з використанням декількох алгебраїчних або логічних функцій) з метою подальшого використання.
- [13]

Використання мережі Ельмана дозволяє вирішити проблему прогнозування навіть для рядів, що містять високу ступінь шуму (це особливо важливо для бізнесу). У загальному випадку ця мережа являє собою структуру з трьох шарів, а також набір додаткових "контекстних" елементів (входів). Зворотній зв'язок надходить від прихованого шару до цих елементів; кожна облігація має фіксовану вагу, рівну одиниці. На кожному інтервалі часу вхідні дані розподіляються по нейронах у прямому напрямку; тоді до них застосовується правило навчання. Завдяки фіксованому зворотньому зв'язку елементи контексту завжди зберігають копію значень із прихованого шару для попереднього кроку (оскільки вони надсилаються в протилежному напрямку до застосування правила навчання). Таким чином, шум часових рядів поступово нівелюється, а разом з цим помилка зводиться до мінімуму: ми отримуємо прогноз, який у загальному випадку буде точнішим за результат класичного підходу, який західні роботи підтверджують експериментально.



### 2.3 Темпоральні згорткові мережі для задач прогнозування

Практики глибокого навчання зазвичай розглядають рекурентні архітектури як вихідну точку за замовчуванням для завдань моделювання послідовностей. Розділи глибокого навчання присвячені прогнозуванню та моделюванню послідовностей в більшості випадків зупиняються на рекурентних мережах.

З іншого боку, останні дослідження вказують на те, що певні згорткові архітектури можуть досягти надвисокої точності у синтезі аудіо, моделюванні мови на рівні слова та машинному перекладі.

Це піднімає питання, чи обмежуються ці успіхи моделювання послідовності за допомогою згортки конкретними напрямками (доменами), чи зв'язок між обробкою послідовностей та рекурентними мережами потребує ширшого огляду.

Для вирішення цього питання наводиться системна та емпірична оцінка згорткових та рекурентних архітектур щодо широкого кола завдань моделювання послідовностей. Запропонована архітектура націлена на комплексний набір завдань, які вже неодноразово використовувались для порівняння ефективності різних рекурентних нейромережевих архітектур. Ці завдання включають поліфонічне моделювання музики, мову на рівні слова та характеру моделювання, а також синтетичні стрес-тести, які були навмисно розроблені та часто використовувались для порівняння RNN.

Таким чином, оцінка ефективності налаштована на порівняння згорткових та рекурентних підходів до моделювання послідовностей.

Для представлення згорткових мереж опишемо загальну архітектуру

темпоральної згорткової мережі (TCN), тобто застосування її до всіх завдань. Ця архітектура з'явилась в нещодавніх дослідженнях, але залишається доволі простою, поєднуючи деякі з найкращих практик сучасної архітектури згорткових мереж. Його порівнюють з канонічними рекурентними архітектурами такі як LSTM та GRU.

Результати свідчать про те, що TCN переконливо перевершують ефективність базових рекурентних архітектур для широкого кола завдань моделювання послідовностей. Це особливо помітно, оскільки завдання включають різноманітні орієнтири, які є загальноприйнятими та використовуються для оцінки нейромереж для роботи з даними у виді послідовностей. Це свідчить про недавні успіхи згорткових архітектур в таких додатках, як аудіо обробка але не обмежується цими доменами.

Для подальшого розуміння цих результатів ми аналізуємо більш глибоко характеристики збереження пам'яті періодичних мереж. Відомо, що незважаючи на теоретичні можливості рекурентних архітектур моделювати нескінченно довгу історію TCN архітектури показують набагато довшу пам'ять та краще підходять для доменів де необхідна довгострокова пам'ять.

Представлене дослідження є найбільш обширним систематичне порівняння згорткових та рекурентних архітектур на завданнях моделювання послідовностей. Результати свідчать що загальна асоціація між моделюванням послідовностей і періодичними мережами слід переглянути. TCN архітектура виглядає не тільки більш точною, ніж канонічні рекурентні мережі, такі як LSTM та GRU, але також простішою і зрозумілішою. Тому вона може бути більш підходящою відправною точкою у застосуванні глибинних мереж до моделювання послідовностей.

Конволюційні мережі застосовується до послідовностей протягом

десятиліть. Вони використовувались для розпізнавання мови в 80-90-х роках.

Згодом ConvNets почали застосовувати до завдань обробки природніх мов, таких як позначення частини мови та маркування семантичної ролі. Більше нещодавно почали застосовувати згорткові мережі для класифікація документів.

Особливо важливими для цієї роботи є останні згорткові архітектури для машинного перекладу та моделювання мови.

Рекурентні мережі - це виділені моделі послідовностей, які підтримують вектор прихованих активацій, які розповсюджуються через часовий лаг (propagation through time).

Це сімейство архітектур здобуло надзвичайну популярність завдяки видатним програмним додаткам до для моделювання природних мов та машинного перекладу.

Інтуїтивна привабливість рекурентного моделювання полягає в тому, що прихований стан може виступати як репрезентація функціональних залежностей що містяться в послідовності. Базові архітектури RNN, як відомо, складно навчати та замість них зазвичай використовуються складні архітектур як LSTM та GRU. Багато інших архітектурних нововведень та методів навчання для рекурентних мереж були впроваджені і продовжують активно досліджуватися.

Ці дослідження частково мотивовані багатьма ступенями свободи в проектуванні таких архітектур.

Ефективність RNN та інших архітектур для поліфонічного моделювання музики, моделювання мови на рівні символів та моделювання мови на рівні слова доведена. Хоча було проведено кілька ретельних оцінок архітектури RNN з моделювання репрезентативних послідовностей, не зустрічається ретельного порівняння згорткового та рекурентного підходів до моделювання послідовностей. Моделювання послідовностей вимагає архітектур, які можуть синтезувати цілі послідовності, елемент за елементом. Таке порівняння є

особливо інтригуючим у світлі згаданого недавнього успіху згорткових архітектур у цій області.

Робота спрямована на порівняння загальних згорткових та рекурентних архітектур із типовими завданнями моделювання послідовностей, а саме прогнозування продажів, які зазвичай використовуються для порівняння самих варіантів RNN. [14]

### 2.3.1 Архітектура згорткових темпоральних мереж

Перш ніж визначати структуру мережі, ми виділяємо характер завдання моделювання послідовності. Припустимо, що є задана вхідна послідовність  $x_0, \dots, x_T$ , і бажаємо передбачити деякі відповідні виходи  $y_0, \dots, y_T$  кожного разу.

Ключове обмеження полягає в тому, що передбачити вихідний  $y_t$  на деякий час  $t$ , ми змушені використовувати лише ті вхідні дані, які є в попередніх спостереженнях:  $x_0, \dots, x_t$ . Формально модель для прогнозування ряду - це будь-яка функція

$f: X_{T+1} \rightarrow Y_{T+1}$ , що формує відображення

$$\hat{y}_0, \dots, \hat{y}_T = f(x_0, \dots, x_T)$$

якщо воно задовольняє причинно-наслідкові обмеження, при яких  $y_t$  залежить лише від  $x_0, \dots, x_t$ , а не на будь-яких “майбутніх” входах  $x_{t+1}, \dots, x_T$ .

Мета навчання в умовах моделювання послідовностей полягає у пошуку мережі  $f$ , яка мінімізує деякі очікувані втрати між фактичними результатами та прогнозами

$$L(y_0, \dots, y_T, f(x_0, \dots, x_T)),$$

де послідовності та результати виводяться відповідно до певного

розподілу.

Цей формальний опис охоплює багато налаштувань, таких як авторегресійне передбачення (де ми намагаємося передбачити якийсь сигнал з урахуванням минулого), встановивши цільовий результат просто як вхід зміщений на один часовий крок. Однак це охоплює такі домени, як машинний переклад, або передбачення sequence-to-sequence загалом, оскільки в цих випадках можна використовувати всю послідовність введення (включаючи "майбутні" стани) та прогнозувати кожен результат (хоча методи можуть природно бути розширеним для роботи в таких налаштуваннях).

Як зазначалося вище, TCN базується на двох принципах: той факт, що мережа прогнозує результат такої довжини як вхідний сигнал, і той факт, що не може бути витоків з майбутнього в минуле. Щоб досягти першого пункту, TCN використовує 1D архітектуру повністю згорткової мережі (FCN), де кожен прихований шар такої ж довжини, як і вхідний шар, і нульове заповнення довжини (розмір ядра - 1) додано для збереження наступних шарів такої ж довжини, як і попередні. Щоб досягти другого пункту, TCN використовує причинно-наслідкові петлі, згортки, де вихід в момент часу  $t$  згортається лише з елементами від часу  $t$  і раніше в попередньому шарі.

Простіше кажучи:  $TCN = 1D\ FCN + \text{причинно-наслідкові звивини}$ .

Слід звернути увагу, що це, по суті, та сама архітектура, що і time-delay нейронні мережі, запропонована майже 30 років тому, з єдиним налаштуванням нульового заповнення до забезпечення однакових розмірів всіх шарів.

Основним недоліком цієї базової конструкції є те, що для того, щоб досягти тривалого ефективного розміру історії, потрібна надзвичайно глибока мережа або дуже великі фільтри, жодна з цих умов не була виконана, коли методи були вперше запроваджені.

### 2.3.2 Розширені згортки

Проста згортка здатна захопити лише спостереження з лінійним розміром в глибині мережі. Це ускладнює застосування вищезазначеної причинно-наслідкової згортки для завдань аналізу послідовностей, особливо тих, що вимагають більше історії спостережень. Рішення полягає у використанні розширених згорток, які використовують експоненціально велике рецептивне поле.

Більш формально, для одновимірної послідовності  $x \in \mathbb{R}^n$

і фільтру  $f: \{0, \dots, k-1\} \rightarrow \mathbb{R}$ , розширена згортка це операція  $F$  над елементом  $s$  послідовності, і визначається як

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i}$$

де  $d$  - коефіцієнт розширення,  $k$  - розмір фільтра, а  $s - d \cdot i$  пояснює «напрямок минулого» в спостереженнях. Таким чином, розширення вводиться еквівалентно введенню фіксованого кроку між кожними двома сусідніми фільтрами. Коли  $d = 1$ , розширена згортка зменшується до звичайної згортки. Використання більшого розширення забезпечує відображення ширшого діапазону вхідних даних, таким чином ефективно розширюючи рецептивне поле мережі.

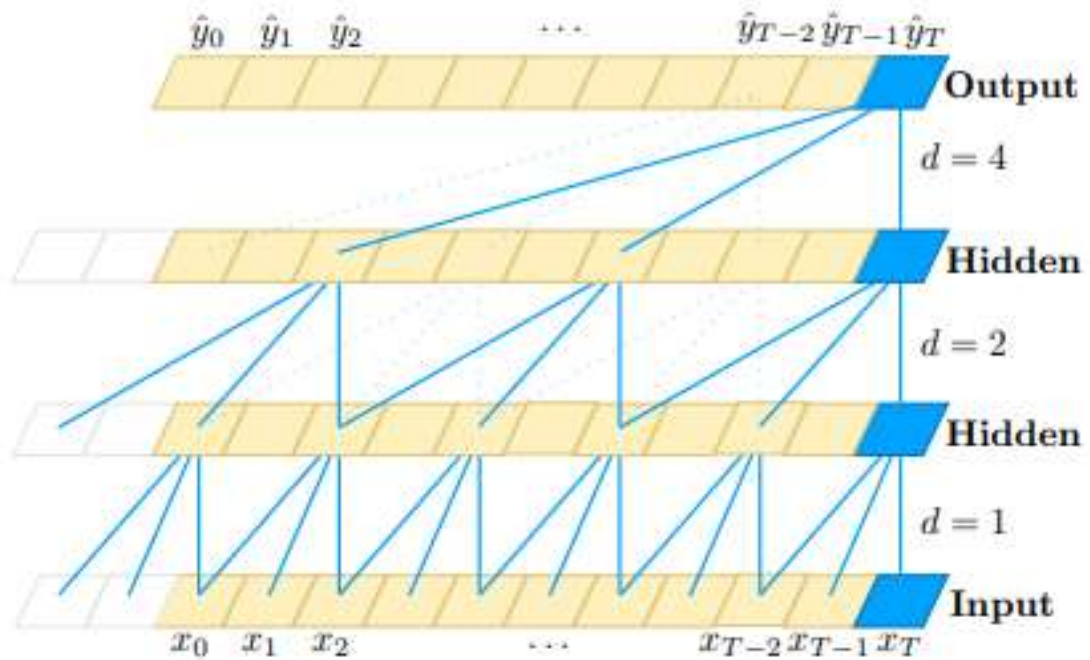


Рисунок 2.6 - Розширена згортка з коефіцієнтами розширення  $d = 1, 2, 4$  та розміром фільтра  $k = 3$ . Рецептивне поле здатне охоплювати всі значення з вхідної послідовності.

Це дає нам два шляхи збільшення рецептивного поля TCN: вибір великих розмірів фільтра  $k$  та збільшення коефіцієнта розширення  $d$ , де фактична історія одного такого шару -  $(k - 1) d$ . Як це часто буває при використанні розширених згорток, ми збільшуємо  $d$  експоненціально з глибиною мережі (тобто  $d = O(2^i)$  на рівні  $i$  мережі). Це гарантує, що є деякий фільтр, який потрапляє на кожен вхід в мережі, водночас допускає надзвичайно великий ефект історії за допомогою глибини мережі (рис 2.4)

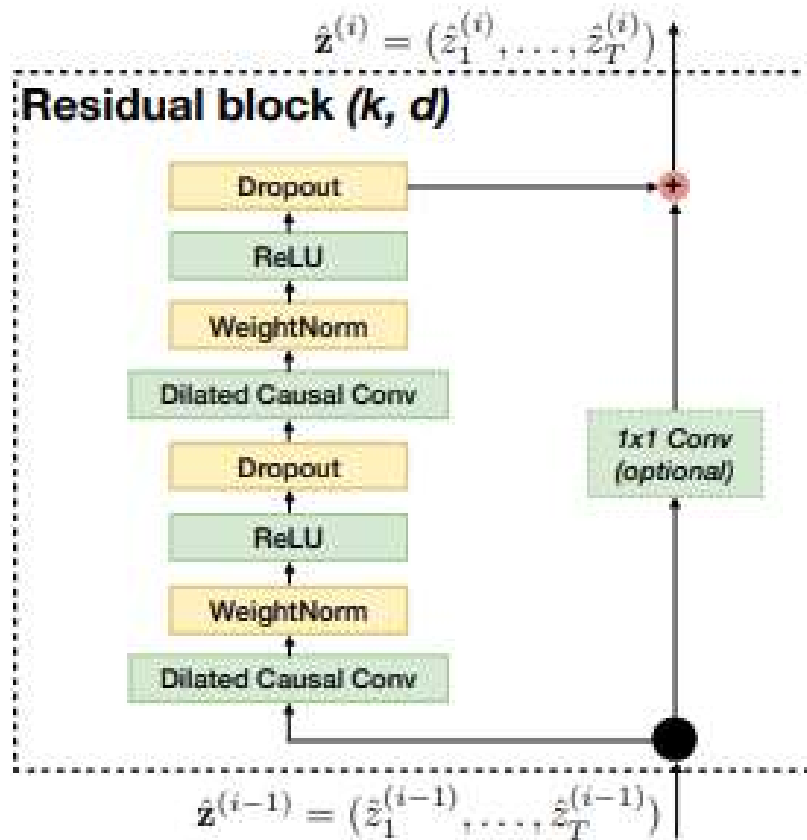


Рисунок 2.7 - Згортковий блок TCN. Згортка 1x1 додається, коли вхід і вихід нев'язки мають різні розміри.

Згортковий блок містить гілку, що веде до серії перетворень  $F$ , вихідні дані яких додаються до входу  $x$  блоку:

$$o = \text{активація}(x + F(x))$$

Це фактично дозволяє шарам вивчати модифікації відображення ідентичності, а не всієї трансформації, ефективність такого кроку неодноразово доведена в глибоких мережах.

Оскільки рецептивне поле TCN залежить від глибини мережі  $n$ , а також розміру фільтра  $k$  і коефіцієнту розширення  $d$ , стабілізація глибоких TCN з великою кількістю шарів стає важливою проблемою. Наприклад, у випадках, коли передбачення може залежати від історії розміру 2 в степені 12 і вхідних даних



великої розмірності, може знадобитися до 12 шарів. Кожен шар, більш конкретно, складається з декількох фільтрів для виділення функцій. Саме тому в архітектурі ми застосовуємо блок нев'язки.

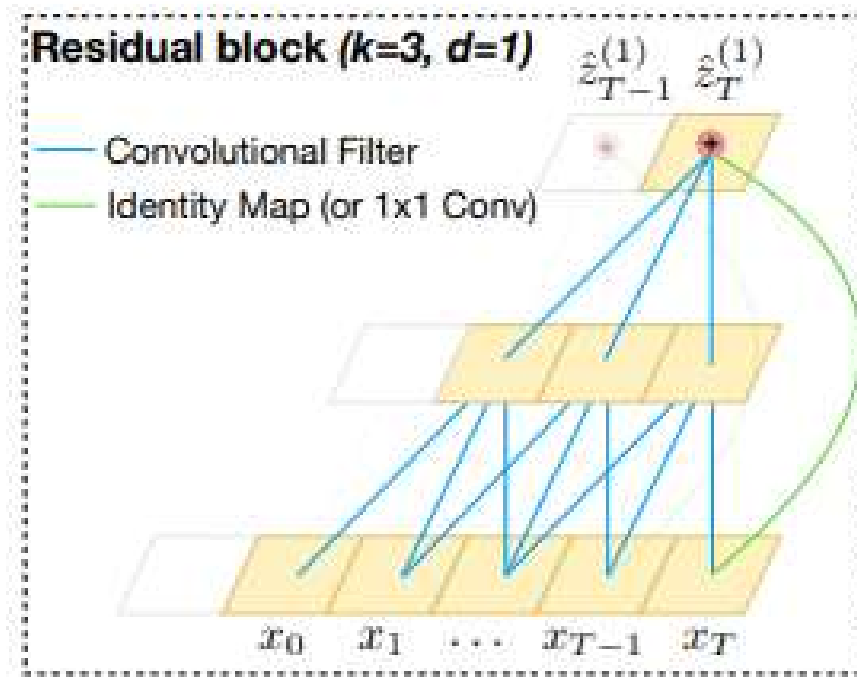


Рисунок 2.8 - Блок нев'язки, де сині лінії – виходи згорткових шарів, а зелена – обхідний сигнал (skip-connection)

Блок нев'язки для базової TCN показаний на Рис 2.6. Всередині блоку нев'язки TCN має два шари розширеної причинно-наслідкової згортки та нелінійності, для яких використано ReLU. Також до згорткових фільтрів застосовується нормалізація вагів.

Крім того, для стандартних мереж з нев'язкою вхідні дані додаються після кожної розширеної згортки для регуляризації: на кожному навчальному кроці обнуляється цілий канал.

Однак, тоді як у стандартному ResNet вхідні дані додаються безпосередньо на вихід функції шару, в TCN (iConvNets загалом) вхід і вихід можуть мати різну

ширину. Щоб врахувати невідповідні ширини вводу-виводу, ми використовуємо додаткову згортку  $1 \times 1$ , щоб переконатись, що поелементне додавання  $\oplus$  отримує тензори однакової форми.

Розглянемо переваги та недоліки використання TCN для моделювання послідовностей.

- Паралельність. На відміну від RNN, де прогнози для подальших кроків часу повинні чекати завершення своїх попередників, згортки можна робити паралельно, оскільки ті самі фільтри використовуються в кожному шарі. Тому як у навчанні, так і для прогнозування, довга входна послідовність може бути оброблена як цілий тензор в TCN, а не послідовно, як у RNN.
- Гнучкий розмір рецептивного поля. TCN може змінювати розмір рецептивного поля різними способами. Наприклад, додаванням більш розширеної (причинно-наслідкової) згортки, використання більших факторів розширення або збільшення розміру фільтра є допустимими варіантами (з можливою їх подальшою інтерпретацією). TCN таким чином забезпечує кращий контроль обсягу пам'яті моделі, і їх легко адаптувати до різних напрямів застосування.
- Стійкі градієнти. На відміну від періодичних архітектур, TCN має шлях зворотного розповсюдження, відмінний від часового напрямку послідовності. Таким чином, TCN уникає проблеми вибуху / зникнення градієнтів, що є основною проблемою для RNN (і що призвело до розвитку LSTM, GRU, HF-RNN)
- Низька потреба в пам'яті для тренувань. Особливо у випадку довгої входної послідовності LSTM та GRU можуть легко використовувати багато пам'яті для зберігання часткових результатів для їх входів з великими розмірностями. Однак у TCN фільтри спільно використовуються через шар із шляхом зворотного розповсюдження залежно лише від глибини мережі.

Тому на практиці виявляється, що керовані RNN мають мультиплікативні коефіцієнти використання пам'яті, на відміну від TCN.

- Вхід змінної довжини. Так само, як RNN, яка модель обробляє вхідні дані змінної довжини шляхом повторень, TCN також може приймати входи довільної довжини, використовуючи одномірні згорткові ядра. Це означає, що TCN можуть використовуватись як заміна рекурентним мережам в задачах роботи з послідовностями.

Є також два помітні недоліки використання TCN:

- Зберігання даних під час оцінки. При оцінці / тестуванні RNN потрібно лише підтримувати прихований стан і приймати в поточний вхід дані для створення прогнозу. Іншими словами "резюме" всієї історії спостережень надається як набір векторів фіксованої довжини  $ht$  і фактично послідовність спостережень можна поступово відкидати. На відміну від них, TCN охоплюють необроблену послідовність фактичної тривалості історії, таким чином, можливо, потрібно більше пам'яті під час оцінки.

- Потенційна зміна параметрів для зміни області використання. Різні домени можуть мати різні вимоги до кількості історії, необхідної моделі для прогнозування. Тому при використанні підходів Transfer learning з домену де потрібно лише мало пам'яті (тобто малі  $k$  та  $d$ ) до домену, де потрібна значно більша пам'ять (тобто значно більший  $k$  і  $d$ ), TCN може погано працювати, якщо не передбачено досить великого рецесивного поля. [14]

## Висновки до розділу

Проведено огляд існуючих методів прогнозування, у тому числі нейромережевих. Нейромережева архітектура TCN має чіткий, теоретично обґрунтований алгоритм, за яким можна побудувати якісну модель, проте конфігурація параметрів моделі ускладнює задачу повної автоматизації процесу прогнозування так як потребує втручання користувача. Основною перевагою такої архітектури є можливість паралелізації обчислень незалежно від часового лагу спостережень.

## РОЗДІЛ 3 АРХІТЕКУРА ТА АНАЛІЗ РЕЗУЛЬТАТІВ РОБОТИ

### 3.1 Вибір платформи та мови реалізації

Для реалізації було обрано мову Python та платформу Jupyter Notebook. Саме така комбінація здатна підтримувати достатньо універсальні інструменти, та забезпечити необхідну розширюваність.

Для попередньої обробки даних було використано бібліотеку pandas. Процес конструювання та навчання нейронних мереж забезпечується за допомогою фреймворка Tensorflow. Окрім автоматизованої роботи з різними нейромережевими топологіями даних фреймворк надає інструментарій для роботи з обчислювальним графом нейронної мережі (computational graph), що може бути важливим при потенційних потребах інтерпретації результатів роботи моделей та побудові СППР.

### 3.2 Структура та попередня обробка даних

Набір даних являє собою спостереження за факторами що можуть впливати на продаж товарів, та зберігаються в базах даних мережі роздрібної торгівлі. Всього входні фактори можна поділити на такі категорії:

- Фактори товару (product-based) – фактори що вказують на особливості товару, та дають можливість нейронній мережі розрізняти товари між собою. Такі фактори зазвичай є категорійними (як приклад можна навести бренд)
- Фактори тренду (trend-based) – фактори, що відстежують ступінь впливу

тренду на прогнозований результат. Тренд певним чином відображає зміну тенденцій на ринку, унікальних для даного моменту.

- Фактори сезону (season-based) – фактори, що дозволяють моделі відстежувати сезонні тренди та сезонні флуктуації попиту.
- Фактори підтримки (support-based) – маються на увазі промо-підтримка. Окрім самої транзакційної ціни на кількість продажів суттєво впливає наявність заохочувальних акцій та додаткових пропозицій.
- Фактори ціни (price-based) – фактори, що вказують ціну товару, її зміну з часом та цінові індекси.

Приклад набору даних наведено на рисунку нижче.

sales_quantity_lag_1_2_diff	week_promo_coverage	sample_weight	brand_id	brand_1	sales_quantity_lag_3	views_lag_1	period_beginning
0.485038	0	1.0	000001349	нал	69800.0	0.0	2018-06-18
-0.292988	0	1.0	000001349	нал	5870.0	0.0	2019-07-29
0.561603	0	1.0	000001349	нал	31100.0	0.0	2018-03-12
-0.010811	0	1.0	000001349	нал	7762.0	0.0	2019-01-28
-0.099226	0	1.0	000001349	нал	9130.0	0.0	2019-10-07

Рисунок 3.1 - Формат набору даних

Перевіримо розподіл продажів (цільового передбачуваного показнику)

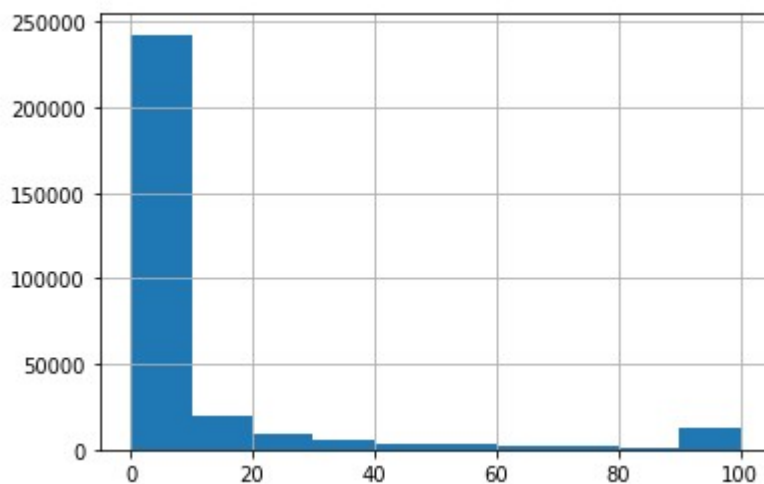


Рисунок 3.2 - Розподіл цільової величини

Для покращення ефективності моделі передбачувана величина повинна бути центрованою.

$$Y = \frac{S_r + const}{S_p + eps} - 1,$$

де  $S_r$  – кількість проданих одиниць,  $S_p$  – кількість проданих одиниць у минулому періоді. Перевіримо розподіл для  $Y$

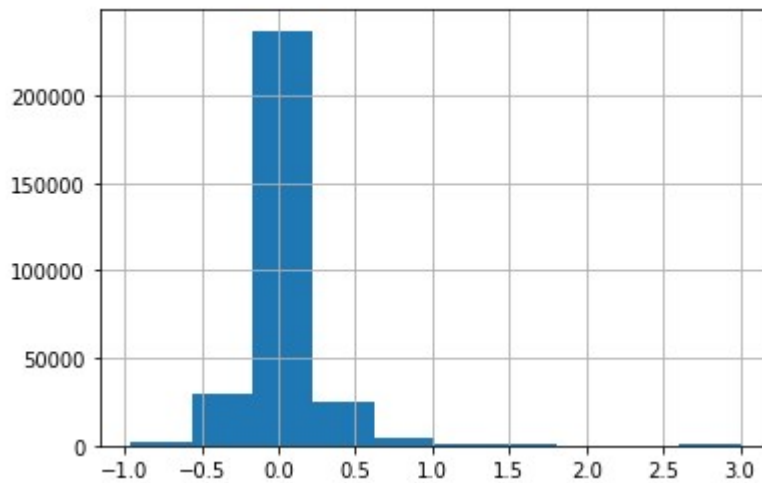


Рисунок 3.3 - Розподіл  $Y$  центрований

### 3.3 Дослідження ефективності роботи запропонованих методів.

Перевіримо якість передбачення рекурентної мережі. Застосуємо LSTM мережу, що містить 32 юніти. Прихований стан рекурентного шару, разом зі статичними факторами, що не змінюються на протязі часу передаємо до Dense шару (що відповідає архітектурі перцептрону), фактично сформувавши гібридну архітектуру.

Для навчання моделі використовується функція витрат Хьюбера, вигляду:

$$L_{\delta}(a) = \begin{cases} \frac{1}{2}a^2 & |a| \leq \delta, \\ \delta(|a| - \frac{1}{2}\delta), & |a| > \delta, \end{cases}$$

Для запобігання перенавчання використовується метод Dropout. В якості алгоритму градієнтного спуску використовується Adam.

### Процес тренування моделі на початкових епохах:

Train on 5608110 samples, validate on 301007 samples

Epoch 1/3

Epoch 00001: val\_loss improved from inf to 0.02087, saving model to /tmp/model\_checkpoint/

5608110/5608110 - 16s - loss: 0.0211 - mae: 0.1291 - val\_loss: 0.0209 - val\_mae: 0.1325

Epoch 2/3

Epoch 00002: val\_loss improved from 0.02087 to 0.02054, saving model to /tmp/model\_checkpoint/

5608110/5608110 - 14s - loss: 0.0195 - mae: 0.1251 - val\_loss: 0.0205 - val\_mae: 0.1303

Epoch 3/3

Epoch 00003: val\_loss improved from 0.02054 to 0.01996, saving model to /tmp/model\_checkpoint/

5608110/5608110 - 17s - loss: 0.0190 - mae: 0.1238 - val\_loss: 0.0200 - val\_mae: 0.1279

В якості основних метрик застосовується APE (Average Portfolio Error):

$$M = \frac{1}{n} \sum \left( \frac{Y_{pred} - Y_{true}}{Y_{true}} \right) * w - 1,$$

Та WAPE (Weighted Mean Average Percentage Error):

$$M = \frac{1}{n} \sum \left( \frac{|Y_{pred} - Y_{true}|}{Y_{true}} \right) * w - 1,$$

де  $n$  – кількість спостережень, що застосовано для обчислення значення метрики,  $w$  – ваги конкретного спостереження.

Використовуються варіанти більш загальноживаних метрик з вагами, тому що в даному випадку найбільш необхідно оцінити портфельну точність



моделі. Надійність саме нейромережевого підходу може бути невисокою для випадків розріджених продаж (кола кількість транзакцій за період є меншою за обсяг асортименту). Для адекватної оцінки портфельної точності, та зменшення впливу товарів з аномальними показниками можна використати у якості вагів логарифмовану суму прибутку для кожного товару.

Перевіримо метрики отриманої моделі. Колонка Bootstrapped містить показники, зважені на прибуток від товарів на яких вони рахувались.

Таблиця 3.1 – Точність LSTM

	Usual	Naive	Bootstrapped
lag	0.0000	0.0000	0
n_observations	131487.0000	131487.0000	131487
revenue_ape	-0.0729	0.0072	-0.21440380546158214
revenue_sum_pred	141192633.8744	141192633.8744	141192633.87440002
revenue_sum_true	140184905.6790	140184905.6790	140184905.67900002
revenue_wape	0.6650	0.6805	0.3416830361957656
revenue_wsmape_accuracy	66.6857	66.0972	80.86445029079914
sales_ape	-0.0746	-0.0083	-0.1379584404404329
sales_r^2	0.6559	0.6700	-
sales_sum_pred	613313.1001	643313.1001	643313.1001204252
sales_sum_true	680499.0000	680499.0000	680499.0
sales_wape	0.5342	0.4690	0.2679757693599474
sales_wsmape_accuracy	77.7827	76.4507	85.60837116837502

Виконаємо тренування мережі TCN та перевіримо метрики відповідно. Як видно, обрана архітектура призвела до покращення метрик.

Таблиця 3.2 – Точність TCN

	Usual	Naive	Bootstrapped
lag	0.0000	0.0000	0
n_observations	131487.0000	131487.0000	131487
revenue_ape	-0.0568	0.0072	-0.212739951632482
revenue_sum_pred	141192633.8744	141192633.8744	141192633.87440002
revenue_sum_true	140184905.6790	140184905.6790	140184905.67900002
revenue_wape	0.6284	0.6805	0.3451329707155428
revenue_wsmape_accuracy	67.6607	66.0972	80.68923252820969
sales_ape	-0.0312	-0.0083	-0.1366013046949478
sales_r^2	0.6696	0.6700	-
sales_sum_pred	638877.3260	638877.3260	638877.3259945512
sales_sum_true	680499.0000	680499.0000	680499.0
sales_wape	0.4311	0.4690	0.2589608917477996
sales_wsmape_accuracy	77.7652	76.4507	86.1026890128851

Не дивлячись на те що дані метрики перевіряються на тестовому датасеті, що ніяким чином не перетинається з датасетом для тренування та валідації (мається на увазі валідація під час тренування, на основі якої спрацьовує механізм ранньої зупинки для запобігання перенавчання) перенавчання все-одно відбувається. Але відбувається не перенавчання відносно навчального датасету, а перенавчання відносно вагомості факторів, а саме факторів тренду.

В багатьох дослідницьких роботах, присвячених прогнозуванню продажів не приділяють уваги тому, наскільки результуюча модель буде враховувати цінові фактори. Частими є випадки, коли модель мінімально враховуючи продуктові фактори, вивчає логіку подібну до бейслайну, і в своїх прогнозах спирається переважно на продажі минулого кроку, не враховуючи при цьому ціну товару. Для оцінки здатності моделі враховувати ціну використовується цінова еластичність попиту. Загальна формула точкової еластичності:

$$Elasticity = \frac{\Delta\%sales}{\Delta\%prices}$$

Розділимо тестовий датасет на декілька сегментів, що не перетинаються, та містять в собі елементи з кожної категорії (мається на увазі товарна категорія верхнього рівня з категорійного дерева магазину). На кожному сегменті збільшимо, а потім зменшимо ціни, в рамках діапазону 10% (стабільний діапазон ціноутворення), та перевіримо на скільки зміниться прогноз моделі.

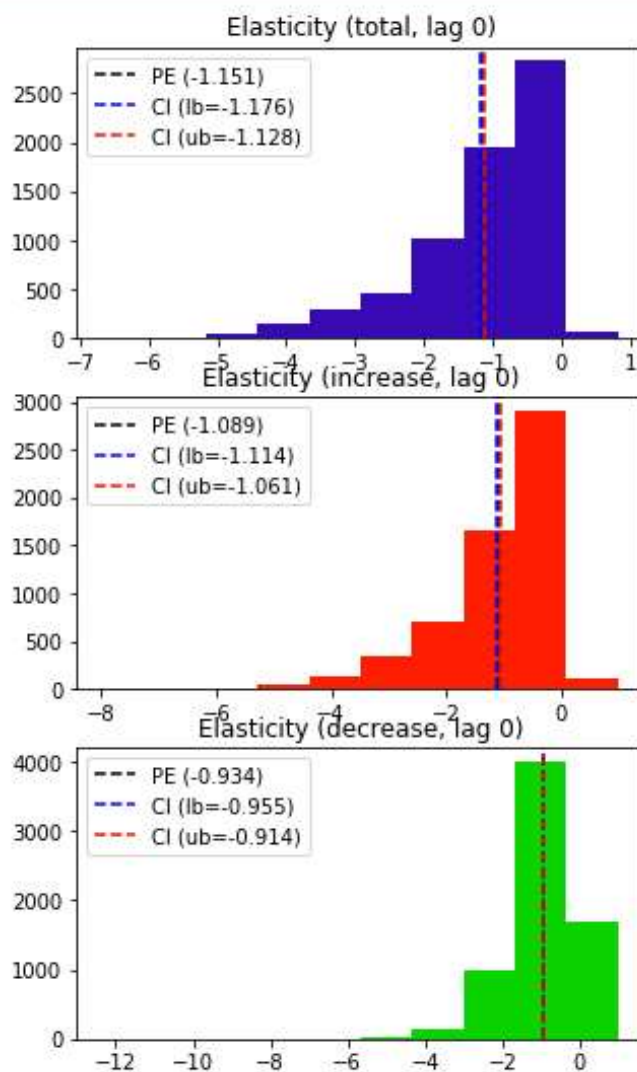


Рисунок 3.4 - Еластичність моделі

Як показано на Рис. 3.4 модель здатна відновлювати еластичність попиту, і відновлена еластичність є переважно від'ємною (що є логічним: зниження цін

приводить до росту продажів і навпаки). Точкова оцінка еластичності знаходиться в околиці -1, що відповідає експертним оцінкам для обраної області.

При прийнятті рішень в ціноутворенні, логічним буде не знижувати ціни на товари з низькою за модулем еластичністю, адже попит на них не залежить від ціни. Важливо розуміти, що така точкова оцінка еластичності є правдивою лише для заданого діапазону, і не здатна достатньо адекватно відобразити те, як змінюватимуться продажі за зміни цін в ширшому діапазоні. Більш того, повністю вірно відновити дугову еластичність не здатна сама модель, тому що набір реальних даних просто не міститиме екстремальних рішень ціноутворення, що призвели б до екстремальних результатів в достатній кількості. Оскільки історичні дані містять рішення по ціноутворенню котрі приймалися категорійними менеджерами, що керуються раціональними міркуваннями, то в моделі не буде можливості вивчити критичні випадки, наприклад коли збільшення ціни вдвічі привело до стрімкого падіння продажів.

### Перевіримо важливість вхідних факторів для моделі:

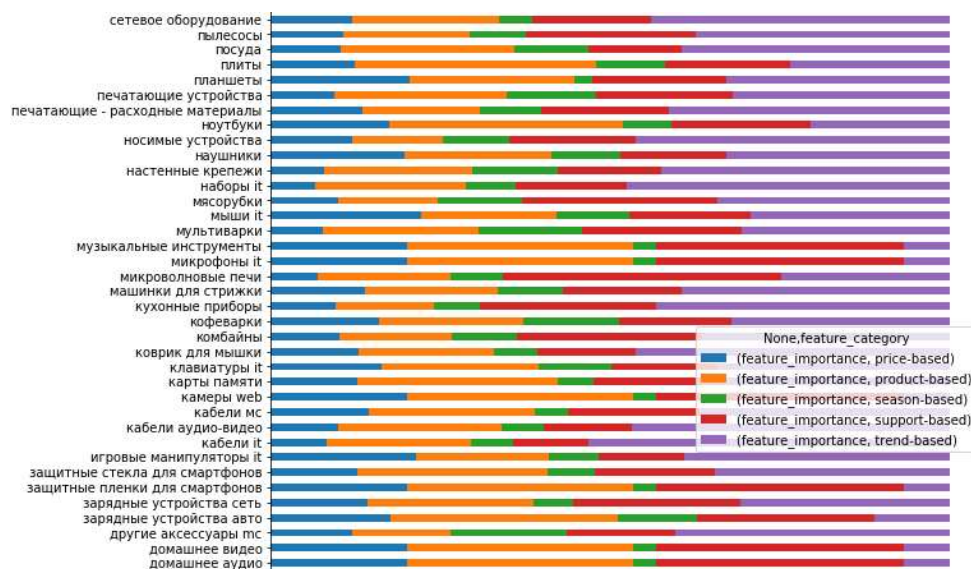


Рисунок 3.5 - Вага факторів для моделі

## Висновки до розділу

Проведено агрегацію та необхідну попередню обробку даних, проведено порівняльний аналіз нейронних мереж рекурентної архітектури та згорткової темпоральної архітектури. Розроблено програмний модуль для прогнозування продажів.

Для обраної моделі перевірено ступінь впливу факторів та можливість відновлення моделлю цінової еластичності попиту.

Розроблений програмний модуль відповідає критеріям придатності для використання у СППР для ціноутворення. Програмний модуль містить елементи пайплайну роботи з нейронною мережею та містить в собі необхідну інформацію

## РОЗДІЛ 4 СТАРТАП АНАЛІЗ ПРОЕКТУ

## 4.1 Інформаційна карта проекту

Ідея та реалізація стартап проекту зображена у таблицях 4.1 – 4.19

Таблиця 4.1 – інформаційна карта проекту

1. Назва проекту	Система прогнозування продажів на основі нейронних мереж
2. Автори проекту	Степаненко Євгеній
3. Коротка анотація (не більше 1/3 сторінки)	Дана система призначена для прогнозування обсягів продажів магазинів роздрібною торгівлі. На основі наявних історичних даних, система надає прогноз на майбутні періоди, з урахуванням виставленої ціни, та може використовуватись для портфельного ціноутворення
4. Термін реалізації проекту	6 місяців
5. Необхідні ресурси	Обладнання – комп'ютери, принтер, сканер, ноутбуки, мобільні телефони. Програмне забезпечення, операційна система, ліцензії на комерційну розробку ПЗ. Електрика, газ, водопостачання, Інтернет. Фінансові ресурси – заробітна плата працівникам на 6 місяців роботи, гроші на оплату комунальних послуг,

Продовження таблиці 4.1

	оренди, реклами тощо. Приміщення з усіма необхідними комунікаціями.
6. Опис проблеми, яку вирішує проект	Дана комплексна система дозволяє зберігати інформацію щодо продажів в мережі магазинів, а також надає прогноз майбутніх продажів для окремих магазинів, в заданий період часу.
7. Головні цілі та завдання проекту	Розробка комплексної системи, робота із реальними даними та створення комерційно успішного продукту.
8. Очікувані результати	Система дозволяє побудувати якомога точний прогноз для мережі магазинів або окремих крамниць, в заданий період часу, та наданою інформацією про зовнішні фактори.

## 4.2 Технологічний аудит ідеї проекту

Таблиця 4.2 - Опис ідеї стартап-проекту

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигоди для користувача</i>
Дана система призначена для прогнозування обсягів продажів магазинів роздрібною торгівлі. На основі наявних історичних даних, система надає прогноз на майбутні періоди, з урахуванням виставленої ціни, та може використовуватись для портфельного ціноутворення	1.Робота з існуючими даними.	Система дозволяє перевірити точність прогнозу та раціональність процесів ціноутворення в минулому
	2.Побудова прогнозів відносно обсягів продажів.	Система дозволяє замовникам зменшити витрати часу та людських ресурсів на створення прогнозу.



Таблиця 4.3 - Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ n/n	Техніко- економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			W	N	S
		Власний проект	Revionics	PriceFx			
1.	Точність прогнозування	Застосування моделі на базі нейромережі	Залежить від вхідних даних	Залежить від вхідних даних			+
2.	Ризики невірною прогнозу	Передбачувані, є високими тільки для аномальних ринкових ситуацій	Залежить від вхідних даних	Залежить від вхідних даних		+	
3.	Доступність зручність	Програмний модуль	Власний інтерфейс	Власний інтерфейс		+	

Таблиця 4.4 - Технологічна здійсненність ідеї проекту

<i>№ n/n</i>	<i>Ідея проекту</i>	<i>Технології реалізації</i>	<i>Наявність технологій</i>	<i>Доступність технологій</i>
1	Створення Системи підтримки	Використання фреймворку Tensorflow	Наявна	Доступні
2	прийняття рішень	Використання фреймворку Tensorflow	Наявна	Доступні
3		Використання фреймворку PyTorch	Наявна	Доступні
Обрана технологія реалізації ідеї проекту: Tensorflow				

### 4.3 Аналіз ринкових можливостей запуску стартап-проекту

Таблиця 1.5 - Попередня характеристика потенційного ринку стартап-проекту

<i>№ п/п</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
1	Кількість головних гравців, од	2
2	Загальний обсяг продаж, грн/ум.од	1000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	14%

Таблиця 4.6 - Характеристика потенційних клієнтів стартап-проекту

<i>№ п/п</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1	Статистичні параметри роботи мережі магазинів	Аналітики, аналітичні відділи	Низька ціна, велика кількість статистичних даних.	Простота використання.
2	Створення якісного прогнозу продажів при обмежених даних.	Малі та середні виробничі або комерційні підприємства.	Цікавить простота у використанні, низька ціна клієнтська підтримка	Низька ціна, репутація
3	Створення точного довгострокового прогнозу продажів.	Великі підприємства	Цікавить передусім точність довгострокових прогнозів, клієнтська підтримка, інтеграція до власних потреб	Висока якість, бренд, ім'я на ринку, успішний досвід

Таблиця 4.7 - Фактори загроз

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Конкуренція	Високий рівень конкуренції	Розробити маркетингову компанію з акцентом на персоналізацію продукту
	Збут	Ускладнення збуту через цінову політику конкурентів	Розміщення додаткової реклами

Таблиця 4.8 - Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1	Гнучкі ціни	Зменшення ціни товару задля збільшення попиту	Введення власних гнучких цін
2	Диференціація витрат	Зменшення витрат за рахунок їх перерозподілу	Зменшення витрат на додаткові, непрофільні задачі.

Таблиця 4.9 - Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Тип конкуренції - Досконала конкуренція	Багато систем/команд аналітиків	Розробити впізнаваний продукт, якість, що вирізнятиме нас від конкурентів
2. За рівнем конкурентної боротьби: міжнародний	На ринку присутні системи, розроблені в інших країнах.	Розширення аудиторії, розширення списку мов, яві підтримуються системою
3. За галузевою ознакою - міжгалузєва	Робота із різногалузєвими підприємствами.	Покращення можливостей персоналізації системи до вимог клієнта.
4. Конкуренція за видами товарів: товарно-родова	Конкуренція між прогнозами інших інформаційних систем та команд аналітиків.	Збільшення функціоналу системи
5. За характером конкурентних переваг: Нецінова	Різні способи прогнозування дають різну точність	Розробка кращих(точніших) алгоритмів
6. За інтенсивністю: марочна	Впізнаваний бренд надає великих переваг	Велику увагу приділити розвитку бренду

Таблиця 4.10 - Аналіз конкуренції в галузі за М. Портером

	<i>Прямі конкуренти в галузі</i>	<i>Потенційні конкуренти</i>	<i>Постачальники</i>	<i>Клієнти</i>	<i>Товари-замінники</i>
<i>Складові аналізу</i>	Інші СППР	Гнучкі ціни, розмір капіталовкладень	Змінні витрати постачальників, диференціація витрат	Контроль якості, система інформації	Ціна, лояльність споживачів
<i>Висновки</i>	Інтенсивна конкуренція	Є як можливості входження на ринок, так і нові потенційні конкуренти	Постачальники не диктують умови роботи на ринку	Клієнти не диктують умови роботи на ринку	Обмеження для роботи на ринку через товари замінники

Таблиця 4.11 - Обґрунтування факторів конкурентоспроможності

<i>№ n/n</i>	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Роботоспроможність на великих обсягах даних	Система підтримує роботу з великими обсягами даних
2	Якість	Висока якість прогнозу, велика кількість допоміжних статистичних даних
3	Обслуговування	Робота з клієнтами – передусім малим та середнім бізнесом

Таблиця 4.12 - Порівняльний аналіз сильних та слабких сторін системи

<i>№ n/n</i>	<i>Фактор конкурентоспроможності</i>	<i>Бали 1-20</i>	<i>Рейтинг товарів-конкурентів</i>						
			-3	-2	-1	0	+1	+2	+3
1	Роботоспроможність на великих обсягах даних	15		2			3		1
2	Якість	20	3	2	1		2		
3	Обслуговування	5					3	1	2



Таблиця 4.13 - SWOT- аналіз стартап-проекту

<p>Сильні сторони:</p> <p>Висока якість прогнозу</p> <p>Інтерфейс користувача</p> <p>Простота використання</p>	<p>Слабкі сторони:</p> <p>Відсутність клієнтської бази</p> <p>Відносно мала функціональність</p>
<p>Можливості:</p> <p>Попит</p> <p>Зміна рівня доходів підприємств</p> <p>Вдосконалення системи</p>	<p>Загрози:</p> <p>Конкуренція</p> <p>Збут</p>

Таблиця 4.14 - Альтернативи ринкового впровадження стартап-проекту

<i>№ п/п</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Швидкий вихід на ринок із «сирим» продуктом, можливі проблеми із точністю прогнозу та попитом	30%	3 місяці
2	Поступовий вихід з готовим, відлагодженим продуктом. Висока якість та конкурентоспроможна ціна.	70%	6 місяців

## 4.4 Розроблення ринкової стратегії проекту

Таблиця 4.15 - Вибір цільових груп потенційних споживачів

<i>№ п/п</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтовний попит в межах цільової групи (сегменту)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1	Окремі аналітики та аналітичні відділи невеликих компаній	Висока	2%	Висока	Середня
2	Малі та середні підприємства	Середня	1%	Висока	Середня
3	Великі підприємства	Низька	1%	Середня	Висока
Які цільові групи обрано: 1,2					

Таблиця 4.16 - Визначення базової стратегії розвитку

<i>№ n/n</i>	<i>Обрана альтернатива розвитку проекту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспроможні позиції відповідно до обраної альтернативи</i>	<i>Базова стратегія розвитку*</i>
1	2	Стратегія диференційованого маркетингу	Високі можливості персоналізації, багатогалузевість, висока якість, ціна.	Стратегія лідерства по витратах

Таблиця 4.17 - Визначення базової стратегії конкурентної поведінки

<i>№ n/n</i>	<i>Чи є проект «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конкурентної поведінки*</i>
	Ні	Так	Ні	Стратегія виклику лідера

Таблиця 4.18 - Визначення стратегії позиціонування

<i>№ п/п</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспромож ні позиції власного стартап-проекту</i>	<i>Вибір асоціацій, які мають сформувані комплексну позицію власного проекту (три ключових)</i>
1	Якість, точність, простота у використа нні	Стратегія лідерства по витратах	Якість прогнозу, велика кількість статистичної інформації	По іміджу Позиціонування на низькій ціні позиціонування за сферою застосування

#### 4.5 Розроблення маркетингової програми стартап-проекту

Таблиця 4.19 - Визначення ключових переваг концепції потенційного товару

<i>№ n/n</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1	Статистичні параметри роботи мережі магазинів	Базова статистика та довгострокові прогнози	Дана статистика має бути максимально розгорнутою та зрозумілою.
2	Створення якісного прогнозу продажів при обмежених даних.	Побудова прогнозів при малому обсязі даних	Можливість інтеграції СППР у підприємствах, які тільки недавно розпочали дію.
3	Створення точного довгострокового прогнозу продажів.	Точний прогноз продажів – основа функціонування будь-якого підприємства.	Розробка коротко- та довгострокових прогнозів, використання нейромережі для покращення точності прогнозу

#### 4.6 Висновки до розділу

Було побудовано стартап проект системи прогнозування продажів для магазинів роздрібною торгівлі (в тому числі інтернет-магазинів). Тривалість проекту 6 місяців. Були відокремлені основні чинники для фіксації ризиків та факторів, які зможуть суттєво вплинути на даний проект.

## ВИСНОВКИ

В даній роботі було проведено огляд методів та методик що використовуються для прогнозування рівнів продажу. Було визначено поняття прогнозу продажів, важливість наявності такого прогнозу для планування роботи торгівельних підприємств та електронної комерції. Визначено порядок їх формування, переваги, недоліки, та ситуації у яких варто скористатися конкретними методами.

У другій частині роботи було розглянуто та описано підходи, що можуть використовуватися для прогнозування продажів, у тому числі з застосуванням нейронних мереж. Описано архітектури моделей, та їх використання в задачах прогнозування. Було описано в загалом які задачі можуть бути виконані за допомогою даних мереж, особливості їх структури, недоліки та переваги.

В третій частині роботи розроблено програмний модуль для прогнозування продажів. Виконано порівняльний аналіз моделей різних архітектур. Перевірено можливість відновлення цінової еластичності попиту для обраної архітектури.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Електронна кммерція. URL:  
[https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F\\_%D0%BA%D0%BE%D0%BC%D0%BC%D0%B5%D1%80%D1%86%D0%B8%D1%8F](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BA%D0%BE%D0%BC%D0%BC%D0%B5%D1%80%D1%86%D0%B8%D1%8F).
2. Машинне навчання URL:  
<https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82-%D0%BC%D0%B0%D0%B3%D0%B0%D0%B7%D0%B8%D0%BD>.
3. Retail e-commerce sales worldwide from 2014 to 2020 URL:  
<https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/>.
4. 9 Factors That Impact Ecommerce Sales URL:: <http://gotvantage.com/9-factors-impact-ecommerce-sales/>.
5. Балик У. О., Колісник М. В., ЕЛЕКТРОННА КОМЕРЦІЯ ЯК ЕЛЕМЕНТ СИСТЕМИ СВІТОВОГО ГОСПОДАРСТВА., 2014
6. Бідюк П.І., Гожий О.П., Коршевніук Л.О, Проектування систем підтримки прийняття рішень, 2012
7. Robustness, iterative stochastic quasigradient procedures, and adaptive (artificial intelligence) learning for cat risk management. Conference: Обчислювальний інтелект. В.Є.Снитюк (ред.) Ужгород: УжНУ, 2019. С. 61–62
8. URL:<https://www.it.ua/knowledge-base/technology-innovation/machine-learning>



9. Mitchell, T. (1997). Machine Learning. McGraw Hill. p. 2. ISBN 978-0-07-042807-2
10. <https://studfile.net/preview/7096698/>
11. Є. С. САКАЛО, АНАЛІЗ ПРОЦЕСІВ РОБОТИ ДИНАМІЧНИХ ОБ'ЄКТІВ,
12. О.М. РІЗНИК, ДИНАМІЧНІ РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ
13. NEURAL NETWORK TIME SERIES PREDICTION BASED ON MULTILAYER PERCEPTRON, Oleg Rudenko, Oleksandr Bezsonov, Oleksandr Romanyk, Development Management, Volume 5, Issue 1, 2019
14. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling, Shaojie Bai, J. Zico Kolter, Vladlen Koltun, arXiv:1803.01271v2 [cs.LG] 19 Apr 2018
15. Data mining and e-commerce: methods, applications, and challenges[електронний ресурс] / Hamid Rasteragi, Mohd hamdan Md Noor // ResearchGate, 2008
16. Harron K. Methodological Developments in Data Linkage / K. Harron, H. Goldstein, C. Dibben., 2016. – 259 с. – (Willey). – (Willey series in probability and statistics).
17. A. Agresti. Categorical Data Analysis. Wiley, New York, NY, 1990.
18. Multi-Relational Record Linkage
19. Ismail, M., Ibrahim, M.M., Sanusi, Z.M. and Nat, M. (2015) Data Mining in Electronic Commerce: Benefits and Challenges. 8, 501-509. <http://dx.doi.org/10.4236/ijcns.2015.812045>
20. <https://www.slideshare.net/himanshurc007/data-mining-applications-in-ecommerce>
21. <http://www.cs.utexas.edu/~ml/papers/marlin-dissertation-06.pdf>

22. Methodological Developments in Data Linkage, Wiley. Katie Harron, Harvey Goldstein, Chris Dibben