

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра програмної інженерії

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр

(назва освітнього ступеня)

на тему: **Розробка мультиагентної програмної системи прогнозування курсу
криптовалют на основі нейромережевих технологій**

Виконав: студент 6 курсу, групи СПМ-61

спеціальності 121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

Корнєв О.О.

(підпис)

(прізвище та ініціали)

Керівник

Пастух О.А.

(підпис)

(прізвище та ініціали)

Нормоконтроль

Стоянов Ю.М.

(підпис)

(прізвище та ініціали)

Завідувач кафедри

Петрик М.Р.

(підпис)

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Тернопіль

2023

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
 (повна назва факультету)

Кафедра програмної інженерії
 (повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Петрик М.Р.

(підпис)

(прізвище та ініціали)

« » _____ 2023р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
 (назва освітнього ступеня)

за спеціальністю 121 Інженерія програмного забезпечення
 (шифр і назва спеціальності)

студенту Корнєву Олександр Олександровичу
 (прізвище, ім'я, по батькові)

1. Тема роботи Розробка мультиагентної програмної системи прогнозування курсу криптовалют на основі нейромережових технологій

Керівник роботи Пастух Олег Анатолійович, д.т.н., професор
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «___» _____ 2023 року № _____.

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до роботи Числові масиви даних – колекція часових рядів курсу криптовалюти біткойн ; програмні інструменти бібліотек Pандас, NumPy, Scikit-learn, Matplotlib для мови програмування Python

4. Зміст роботи (перелік питань, які потрібно розробити) Огляд та аналіз предметної області: фінансової інженерії, фінансових інструментів та фінансових технологій, проектування та розробка мультиагентної програмної системи прогнозування курсу криптовалюти біткойн, практична реалізація мультиагентної програмної системи прогнозування курсу криптовалюти біткойн.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів) Розглянуто актуальність теми, мету роботи, об'єкт дослідження, предмет дослідження, завдання дослідження, наукову новизну одержаних результатів, практичне значення результатів роботи. Дані мультиагентної програмної системи прогнозування курсу криптовалюти біткойн. Розбиття даних на вхідні та вихідні дані для мультиагентної програмної системи. Проектування архітектури мультиагентної програмної системи прогнозування курсу криптовалюти біткойн. Програмний агент на основі нейромережової архітектури багатошаровий перцептрон. Прогнозування курсу криптовалюти біткойн програмним агентом на основі нейромережової архітектури багатошаровий перцептрон. Програмний агент на основі згорткової нейромережі. Прогнозування курсу криптовалюти біткойн програмним агентом на основі згорткової нейромережі. Висновки.

6. Консультанти розділів роботи _____

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Осухівська Г.М.		
Безпека в надзвичайних ситуаціях	Стручок В.С.		
Нормоконтроль	Стоянов Ю.М.		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Вибір та затвердження теми		виконано
2.	Аналіз предметної області		виконано
3.	Постановка завдань розробки		виконано
4.	Проектування розробки		виконано
5.	Практична реалізація розробки		виконано
6.	Оформлення записки кваліфікаційної роботи магістра		виконано
7.	Охорона праці та безпека в надзвичайних ситуаціях		виконано
8.	Попередній захист		виконано
9.	Нормоконтроль		виконано
10.	Захист кваліфікаційної роботи магістра		виконано

Студент

_____ (підпис)

Корнєв О.О.

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

Пастух О.А.

_____ (прізвище та ініціали)

РЕФЕРАТ

Кваліфікаційна робота магістра. Тернопільський національний технічний університет імені Івана Пулюя, кафедра програмної інженерії, спеціальність 121 «Інженерія програмного забезпечення». ТНТУ, 2023. Сторінок - 91, рисунків - 36, додатків – 3, презентація.

Тема: Розробка мультиагентної програмної системи прогнозування курсу криптовалют на основі нейромережових технологій.

Метою кваліфікаційної роботи магістра є проектування, розробка та практична реалізація мультиагентної програмної системи прогнозування курсу криптовалюти на основі нейромережових технологій, зокрема, фінансового інструмента управління інвестиціями в криптовалюту Біткойн. В роботі здійснено огляд та аналіз предметної області, проектування мультиагентної програмної системи прогнозування курсу криптовалюти біткойн, практичну реалізацію мультиагентної програмної системи прогнозування курсу криптовалюти біткойн.

Результатом виконання кваліфікаційної роботи магістра є фінансовий інструмент для управління інвестиціями в криптоактив у вигляді мультиагентної програмної системи прогнозування курсу криптовалюти біткойн на основі нейромережових технологій.

Ключові слова: МУЛЬТИАГЕНТНА ПРОГРАМНА СИСТЕМА, НЕЙРОМЕРЕЖЕВІ ТЕХНОЛОГІЇ, ФІНАНСОВИЙ ІНСТРУМЕНТ, ФІНАНСОВА ІНЖЕНЕРІЯ.

ABSTRACT

Master's qualification work. Ivan Pulyuy Ternopil National Technical University, department of software engineering, specialty 121 "Software engineering". TNTU, 2023. Pages - 91, drawings - 36, appendices - 3, presentation.

Topic: Development of a multi-agent software system for forecasting the exchange rate of cryptocurrencies based on neural network technologies.

The purpose of the master's qualification work is the design, development and practical implementation of a multi-agent software system for forecasting the cryptocurrency exchange rate based on neural network technologies, in particular, a financial tool for managing investments in the Bitcoin cryptocurrency. The work includes an overview and analysis of the subject area, the design of a multi-agent software system for forecasting the exchange rate of the Bitcoin cryptocurrency, and the practical implementation of a multi-agent software system for forecasting the exchange rate of the Bitcoin cryptocurrency.

The result of the master's qualification work is a financial tool for managing investments in a crypto-asset in the form of a multi-agent software system for forecasting the exchange rate of the bitcoin cryptocurrency based on neural network technologies.

Keywords: MULTI-AGENT SOFTWARE SYSTEM, NEURONETWORK TECHNOLOGIES, FINANCIAL TOOL, FINANCIAL ENGINEERING.

ЗМІСТ

Перелік умовних позначень.....	8
Вступ.....	9
1. Огляд та аналіз.....	12
1.1. Огляд та аналіз предметної області: фінансової інженерії.....	12
1.2. Вхідні дані МПС прогнозування курсу КБ.....	17
2. Проектування та математичне забезпечення мультиагентної програмної системи прогнозування курсу криптовалюти біткойн.....	21
2.1. Проектування архітектури МПС прогнозування курсу КБ.....	21
2.2. Математичне забезпечення програмного агента на основі нейромережевої архітектури багатошаровий перцептрон.....	23
2.2.1. Модельна візуалізація біологічного нейрона.....	23
2.2.2. Графова модель штучного нейрона.....	24
2.2.3. Біологічний нейрон vs. штучний нейрон.....	27
2.2.4. Функції активації штучного нейрона.....	27
2.2.5. Програмний агент на основі нейромережевої архітектури багатошаровий перцептрон.....	30
2.2.6. Навчання програмного агента на основі нейромережевої архітектури багатошаровий перцептронпо по методу алгоритму оберненого поширення похибки.....	32
2.2.7. Перший нейромережевий програмний агент МПС прогнозування курсу КБ.....	34
2.3. Математичне забезпечення програмного агента на основі нейромережевої архітектури згорткової нейромережі.....	35
3. Практична програмна реалізація мультиагентної програмної системи прогнозування прогнозування курсу криптовалюти біткойн.....	44
3.1. Прогнозування курсу криптовалюти біткойн програмним агентом на основі нейромережевої архітектури багатошаровий перцептрон.....	45

3.2. Прогнозування курсу криптовалюти біткойн програмним агентом на основі нейромережевої архітектури згорткової нейромережі.....	52
3.3. Консолідація нейромережевих програмних агентів у МПС прогнозування курсу КБ.....	59
4. Охорона праці та безпека у надзвичайних ситуаціях.....	61
4.1. Охорона праці.....	61
4.2. Прогнозування можливої обстановки під час надзвичайних ситуацій мирного та воєнного часу.....	64
Висновки.....	67
Перелік джерел посилання.....	68
Додаток А Тези конференції.....	73
Додаток Б Диск із кваліфікаційною роботою магістра.....	76

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ФІ – фінансовий інструмент

ФТ (фінтех) – фінансові технології

КБ – криптовалюта біткойн

ІТ – інформаційні технології

ТД – технології даних

ШІ – штучний інтелект

МПС – мультиагентна програмна система

НТ – нейромережеві технології

БП (MLP) – багатошаровий перцептрон

ЗНМ (CNN) – згорткова нейронна мережа

ВСТУП

Актуальність теми роботи.

Однією із нових і перспективних галузей людської діяльності є фінансова інженерія. Сучасна фінансова інженерія на сьогодні представляє конвергенцію різних сфер науки, технологій та людської діяльності, а саме, сфер: фінансів, математичної статистики, програмування, технологій штучного інтелекту, психології, тощо.

Серед важливих прикладних завдань, які розглядаються у фінансовій інженерії, є проектування, розробка та практична реалізація нових фінансових продуктів: нових фінансових інструментів (ФІ) та нових фінансових технологій (ФТ, фінтех, fintech). До важливих галузей фінансової інженерії, які вирішують фінансові інженери, відносяться, наприклад,: комерція ф'ючерсами та опціонами, хеджування ризиків, управління інвестиціями, оптимізація затрат, тощо. Зокрема, у розрізі вище сказаного, буде доречно зазначити сучасні приклади проектування, розробки та практичної реалізації нових фінансових інструментів та фінансових технологій – проектування, розробка та практична реалізація ф'ючерсних та опціонних контрактів на криптоактиви з використанням технології блокчейн.

У кваліфікаційній роботі магістра розглядається галузь фінансової інженерії – управління інвестиціями, зокрема, управління інвестиціями у фінансові криптоактиви (цифрові фінанси) на прикладі криптовалюти біткойн.

Для конструктивного управління інвестицією в криптоактив біткойн розглядається задача прогнозування у часі курсу криптовалюти біткойн, як узагальнена задача прогнозування колекції часових рядів на основі даних у часі з криптобірж. Такий прогноз у часі може слугувати додатковою інформацією трейдеру для підтримки прийняття ним рішення щодо купівлі чи продажу криптовалюти біткойн. Проте, одержання такого роду якісної та реактивної у часі інформації для підтримки прийняття рішення трейдеру є не тривіальною задачею. Для цього йому потрібно використовувати фінансовий продукт – фінансовий

інструмент, який би міг з високою точністю, реактивністю та стійкістю давати прогноз курсу криптовалюти біткойн (КБ) у часі.

Такого роду фінансовий інструмент може бути спроектований, пізніше розроблений та практично реалізований на основі використання сучасних інформаційних технологій (ІТ), технологій даних (ТД), технологій штучного інтелекту (ШІ), тощо.

Тому для успішного трейдингу на ринку криптовалют, а саме, одержання інформації щодо прогнозування у часі курсу КБ у роботі розглядається проектування, розробка та практична реалізація ФІ – мультиагентної програмної системи (МПС) з використанням сучасних ІТ, а саме, технологій ШІ, зокрема, нейромережових технологій (НТ) на основі нейронних мереж з архітектурами багат шарового перцептрона (БП) та згорткової нейронної мережі (ЗНМ). Такі нейронні мережі слугуватимуть інтелектуалізованими програмними агентами МПС. Слід зазначити, що такі інтелектуалізовані програмні агенти на основі НТ показують стабільно хороші прогнози у часі не залежно від горизонту прогнозування – короткострокового, середньострокового чи довгострокового.

Фактично такий ФІ у вигляді МПС буде елементом ФТ для управління інвестиціями у криптоактив – КБ.

Мета роботи.

Суть мети кваліфікаційної роботи полягає в проектуванні, розробці і практичній реалізації МПС прогнозування курсу криптовалюти на основі нейромережових технологій, зокрема, ФІ управління інвестиціями в КБ.

Об'єкт дослідження.

МПС прогнозування курсу КБ на основі технологій нейронних мереж, як фінансовий інструмент для управління інвестиціями в КБ.

Предмет дослідження.

Методи та засоби математичного та програмного забезпечення МПС прогнозування курсу КБ на основі технологій нейронних мереж, як фінансового інструмента для керування інвестиціями в криптоактив біткойн.

Завдання дослідження.

1. Запропонувати нові методи та засоби математичного та програмного забезпечення для проектування МПС прогнозування курсу криптовалюти біткойн на основі технологій нейронних мереж – фінансового інструмента для керування інвестиціями в криптоактив біткойн.

2. На основі нових методів та засобів математичного та програмного забезпечення розробити та практично реалізувати МПС прогнозування курсу КБ на основі технологій нейронних мереж – фінансовий інструмент для керування інвестиціями в криптоактив біткойн.

Наукова новизна одержаних результатів.

Запропоновано нові методи та засоби математичного та програмного забезпечення для проектування, розробки і практичної реалізації МПС прогнозування курсу КБ на основі технологій нейронних мереж – фінансового інструмента для керування інвестиціями в криптоактив біткойн для зменшення ризиків.

Практичне значення результатів роботи.

Практичне значення роботи полягає у зменшенні ризиків для суб'єктів фінансових ринків завдяки використанню сучасного фінансового інструмента управління інвестиціями в КБ – МПС прогнозування курсу валют на основі нейромережевих технологій.

1. ОГЛЯД ТА АНАЛІЗ

1.1. Огляд та аналіз предметної області: фінансової інженерії

Фінансова інженерія галузь, що досліджує властивості фінансових структур. Окрім цього, фінансова інженерія досліджує, як раціональним чином керувати фінансами, що залежать від часу, випадкових характеристик зовнішнього середовища, ризиків, використовуючи фінансові інструменти, фінансові операції та фінансові технології.

Сучасна фінансова інженерія у переважній більшості інтенсивно використовує сучасні інструменти інформаційних технологій, технологій даних та конвергентних технологій.

Застосуванню інструментів штучного інтелекту у вигляді тих чи інших архітектур нейромережевих технологій та створенні на їх основі різного виду фінансових продуктів: фінансових інструментів, фінансових технологій у сфері фінансової інженерії та загалом у сфері фінансів присвячена велика кількість інформації [1-29].

Фінансова інженерія оперує такими об'єктами та структурами, як, наприклад, фінансовий ринок, люди – учасники фінансового ринку, підприємства та різного роду фірми, тощо. Серед фінансових об'єктів та структур фінансової інженерії особливе місце посідає фінансовий ринок. Фінансовий ринок – це узагальнена сутність, що охоплює в собі ринок фіатної валюти, ринок криптоактивів, ринок фінних паперів, ринок цінних металів, ринок фінансових інструментів у тому числі ринок програмних фінансових інструментів до якого належить розроблювальний у магістерській роботі програмний фінансовий інструмент.

Загалом, фінансові інструменти поділяють на дві групи: перша група – це первинні (елементарні) фінансові інструменти та друга група – це вторинні (складені з елементарних) фінансові інструменти. Відзначимо, що хоч до основних (первинних, елементарних) фінансових інструментів відносяться цінні папери акції, тощо, а до вторинних (складених) відносяться опціони, свопи, ф'ючерси,

тощо, у роботі розглядається розробка програмного фінансового інструменту для управління інвестиціями у криптоактив шляхом прогнозування у часі курсу КБ, щоб у подальшому його купувати або продавати.

Доволі часто, коли вживають поняття фінансова інженерія, то мають на увазі оперування з вторинними (складеними з елементарних) фінансовими інструмента

Значне місце у фінансовій інженерії посідає розробка математичного забезпечення щодо прогнозування тих чи інших фінансових активів, наприклад, цінних паперів, криптовалюти, тощо. Від точності розробленого математичного забезпечення у подальшому залежить прийняття того чи іншого рішення гравцями фінансового ринку, наприклад, купівля чи продаж тих чи інших акцій, тої чи іншої криптовалюти, тощо. Поряд із точністю велике значення має швидка динаміка прийняття таких рішень, оскільки фінансовий ринок є дуже динамічний до того ж і стохастичний. Тому висуваються надзвичайно високі вимоги до розробки математичного забезпечення, потім на його основі алгоритмічного забезпечення, після того програмного забезпечення із вибором тих чи інших програмних інструментів, а також обчислювальної інфраструктури на якій всі ці речі будуть реалізовуватися. Оскільки, затрати обчислювальних ресурсів: ресурсів пам'яті та ресурсів часу на виконання роботи програмним фінансовим інструментом також несуть фінансові витрати, то використання тих чи інших обчислювальних машин, хмарних сервісів, кластерних обчислювальних систем потребують також свого детального продумування.

Важливим моментом є вибір математичного забезпечення, що описує динаміку у часі ціни фінансового інструмента. У переважній більшості за основу математичного забезпечення такого процесу беруть випадкові функції із інтерпретацією аргументу, як час. Зокрема, вибирають випадкові функції, наприклад, Вінерівські випадкові функції, Марківські випадкові функції, умовні Вінерівські випадкові функції, умовні Марківські випадкові функції першого роду, умовні Марківські випадкові функції другого роду, лінійні випадкові функції, умовні лінійні випадкові функції, періодичнокорельовані випадкові функції, білі випадкові шумові функції з різними розподілами імовірностей, кольорові шумові

випадкові функції також із різними розподілами імовірностей, а також інші класи випадкових функцій. Окрім цього, використовують різного роду випадкові породжуючі інтегральні перетворення Стілтєса та Лебега в яких під диференціалом знаходяться випадкові функції із незалежними або не корельованими зростаннями похідні яких представляють собою білі або майжекольорові випадкові шумові функції. Також до таких випадкових інтегральних перетворень відносяться такі в яких використовуються різного роду детерміновані фінітні ядерні функції, котрі ще називають детермінованими фільтруючими функціями або детермінованими ваговими функціями, детермінованими функціями відгуку системи. Такого роду випадкові інтегральні перетворення взаємно та однозначно, а більш загально, ізоморфно представляються різними канонічними видами логарифма їх функції спектральної щільності Фур'є.

Для оцінювання точності прогнозу у часі курсу криптоактиву застосовують цілий пул різних підходів. Зокрема, оцінюють випадкову функцію залишків між реальним часовим рядом та прогнозною функцією на тестовій часовій вибірці. Оцінювання таких випадкових залишків відбувається наступним чином. Оцінюються їх автокореляційні та частково автокореляційні функції, зокрема, наскільки, перші, відносно початку координат, значення значимо відрізняються від нуля. В ідеальному випадку вони мали би нагадувати функцію Дірака (дельта функцію). Кількість значимо відрізняючихся значень від нуля, відносно початку координат, використовуються, як кількісна міра не ідеальності моделі. Наступним кроком здійснюється перевірка на значимо чи не значимо випадкова функція залишків відрізняється він нормального закону розподілу імовірностей. Для цього використовуються різного роду статистичні тести. Наприклад, може використовуватися статистичний тест Колмогорова-Смірнова або статистичний тест Неймана для розподілів різного роду імовірностей, тощо. Наступним етапом йде перевірка випадкової залишкової функції на стаціонарність. Для цього використовується статистичний тест Дікі-Фулера у якому гіпотеза H_0 відповідає твердженню, що випадкова залишкова функція є не стаціонарною випадковою функцією, а гіпотеза H_1 відповідає твердженню, що залишкова випадкова функція

є стаціонарною у часі випадковою функцією. У даному контексті слушно відзначити, що розглядаються різного роду стаціонарності у часі, а саме, стаціонарність у широкому розумінні, коли перший початковий момент і другий центральний момент є незмінними у часі детермінованими функціями, а також стаціонарність у вузькому розумінні, коли всі початкові моменти та всі центральні моменти є не змінними у часі детермінованими функціями. Або ж розглядають спектральні образи їх випадкових інтегральних представлень в канонічних видах логарифмів амплітудно-частотних характеристик.

Загалом, для задач прогнозування у часі використовують різного роду підходи та в рамках них різні алгоритми. Наприклад, клас алгоритмів експоненційного згладжування. Цей клас алгоритмів розглядає у своїй основі мультиплікативну суміш тренової функції, сезонної функції та випадкової функції або адитивну суміш тренової функції, сезонної функції та випадкової функції. В рамках адитивної суміші розглядаються різні моделі з різними трендовими функціями, наприклад, експоненційним трендом, лінійною трендовою функцією, логарифмічною трендовою функцією. Аналогічне місце має мультиплікативна суміш, тобто розглядаються різні мультиплікації з різними трендовими функціями, наприклад, експоненційною трендовою функцією, лінійною трендовою функцією, логарифмічною трендовою функцією. Вище наведений клас моделей у переважній своїй більшості використовується у середньостроковому прогнозуванні. Хоча непогані прогнози, прогнози з невеликою відносною похибкою можуть бути й на короткостроковому часовому горизонті прогнозування. До переваги цього класу алгоритмів слід віднести малі затрати обчислювальних ресурсів – ресурсів процесорного часу та пам'яті, тобто вони є швидкодіючими. Саме цим пояснюється їх широке використання у самих різних пакетах прикладних програм, наприклад Статистика, ексель, тощо.

Слід відмітити також, що для прогнозування у часі курсу криптоактивів часто застосовують клас моделей козного середнього різного ступеня також моделі авторегресії різного ступеня. І часто використовують алгоритми одночасно ковного середнього та авторегресії різного ступеня. В цих алгоритмів є також

великі переваги. Наприклад, вони також використовують малі затрати обчислювальних ресурсів, а саме, ресурсів процесорного часу та ресурсів пам'яті. Тобто вони є швидкодіючими алгоритмами і можуть добре застосовуватися навіть на мобільних обчислювальних пристроях не говорячи вже про стаціонарні обчислювальні пристрої чи то хмарні обчислювальні пристрої. А це в свою чергу означає, що потрібно менше фінансових затрат на їх підтримку роботи. Цей клас алгоритмів також у своїй структурі враховує трендовість (причому різні трендові форми, тобто лінійні трендові форми, експоненційні трендові форми, логарифмічні трендові форми, тощо) – трендові функції, сезонні компоненти різної форми. Цей клас моделей найвищу точність прогнозів показує на короткострокових часових горизонтах. Також не велику похибку показує на середньострокових часових горизонтах. Ці моделі набули широкого вжитку у різних секторах діяльності народного господарства та життя. Це видно з того, що вони використовуються у широкому спектрі пакетів прикладного програмного забезпечення призначених для найрізноманітніших фахівців.

Окремо для прогнозування у часі розглядають різного роду (виду) регресійні алгоритми. До них у переважній більшості відносяться лінійна регресія, гребнева лінійна регресія, лінійна регресія лассо, лінійна регресія еластичної мережі. Різні види цих регресій відрізняються між собою регуляризаторами – штрафними функціями. Регуляризатори розглядаються, як нормуючі вектори у різних нормуючих функціональних просторах часових рядів курсу криптоактивів. Перевагами алгоритмів лінійних регресій є їх обчислювальна легкість, тобто на їх роботу затрачається мало обчислювальних ресурсів: ресурсів процесорного часу та ресурсів пам'яті. Ці види алгоритмів прогнозування найчастіше використовуються при довгостроковому часовому горизонті, який дорівнює приблизно щонайменше одному періоду і більше основної сезонної компоненти. Хоча часто вважається від двох третин періоду сезонної компоненти. Також алгоритми лінійної регресії показують прогнози з малою похибкою, часто, на середньострокових часових горизонтах. Вони також, як клас алгоритмів експоненційного згладжування так і ковзного середнього і авторегресії набули широкого використання і саме тому їх

часто використовують у різних пакетах прикладних програм різного призначення у різних сферах людської діяльності.

1.2. Вхідні дані МПС прогнозування курсу КБ

Для прогнозування у часі МПС курсу КБ розглядаються вхідні дані у вигляді колекції часових рядів: "Open", "High", "Low", "Close", де часовий ряд "Open" – характеризує курс КБ при відкритті торгівлі на криптобіржі, часовий ряд "High" – характеризує найбільше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Low" – характеризує найменше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Close" – характеризує курс КБ при закритті торгівлі на криптобіржі. Рядковий індекс "Date" – набуває значень моментів часу.

На рис.1.1 наведено частковий вид вхідних даних МПС – колекцію часових рядів: "Open", "High", "Low", "Close" курсу КБ в моменти часу, що представляються рядковим індексом "Date" у вигляді набору даних Data Frame, як об'єкта бібліотеки Pandas.

На рис.1.2 наведено візуалізацію у вигляді графіків вхідних даних МПС – колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ у моменти часу представлених рядковим індексом "Date".

У даному дослідженні вхідні дані МПС прогнозування курсу КБ – колекція часових рядів: "Open", "High", "Low", "Close", де часовий ряд "Open" – характеризує курс КБ при відкритті торгівлі на криптобіржі, часовий ряд "High" – характеризує найбільше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Low" – характеризує найменше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Close" – характеризує курс к КБ при закритті торгівлі на криптобіржі. Рядковий індекс "Date" – набуває значень моментів часу; безпосередньо у виді, який наведено на рисунку вище, не подаються.

Date	Open	High	Low	Close
2018-01-01T00:00:00.000000000	13860.1	13887.2	14169.9	13412.4
2018-01-02T00:00:00.000000000	13412.4	15216.8	14169.9	14740.8
2018-01-03T00:00:00.000000000	14740.8	15394	14169.9	15134.6
2018-01-04T00:00:00.000000000	15134.6	15395	14169.9	15155.2
2018-01-05T00:00:00.000000000	15155.2	17118.4	14169.9	16937.2
2018-01-06T00:00:00.000000000	16937.2	17211.9	14169.9	17135.8
2018-01-07T00:00:00.000000000	17135.8	17144.8	14169.9	16178.5
2018-01-08T00:00:00.000000000	16178.5	16269.7	14169.9	14970.4
2018-01-09T00:00:00.000000000	14970.4	15360.1	14169.9	14439.5
2018-01-10T00:00:00.000000000	14439.5	14890.7	13455.7	14890.7
2018-01-11T00:00:00.000000000	14890.7	14942.6	12836.9	13287.3
2018-01-12T00:00:00.000000000	13287.3	14095.1	12878.6	13812.7
2018-01-13T00:00:00.000000000	13812.7	14536.5	13810.8	14188.8

Рис.1.1. Частковий вид вхідних даних МПС – колекції часових рядів:
 "Open", "High", "Low", "Close" курсу КБ.

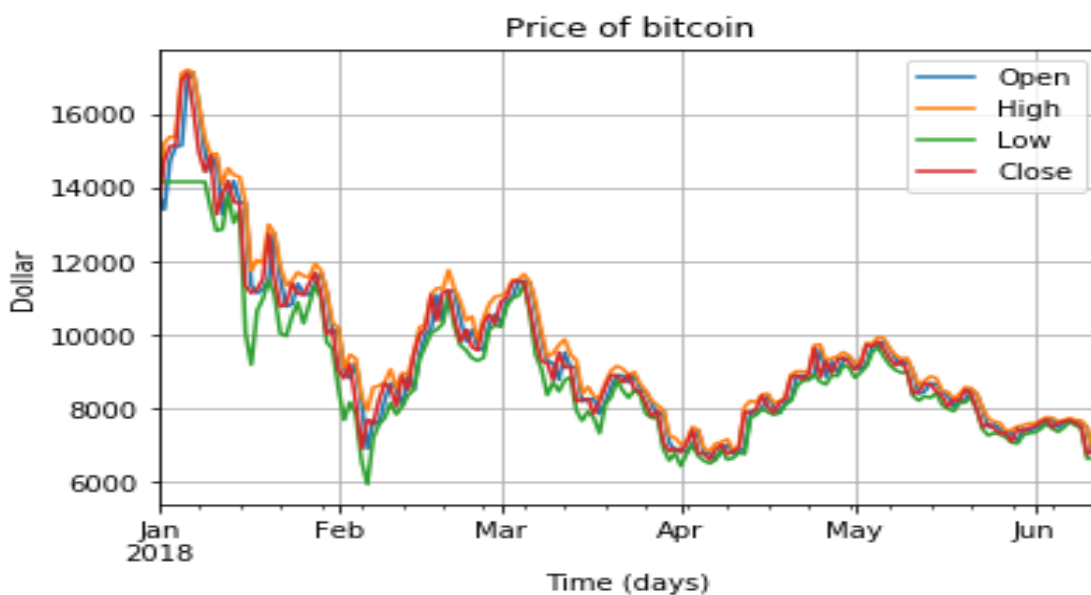


Рис.1.2. Візуалізація вхідних даних МПС – колекції часових рядів:
 "Open", "High", "Low", "Close" курсу КБ.

Ап'іорі над ними виконується попередня переструктуризація, тобто зміна їх структури. Data Frame об'єкт бібліотеки Pandas перетворюється в одновимірний масив Series, який також є об'єктом бібліотеки Pandas. Потім, цей одновимірний масив розбивається двома ковзними вікнами по вісі часу на вхідні дані та вихідні дані для кожного з неймережевих програмних агентів мультиагентної програмної системи прогнозування у часі курсу КБ. Для наочності приклад цієї процедури розбиття у часі двома ковзними вікнами даних на вхідні дані та вихідні дані наведено на рис.1.3.

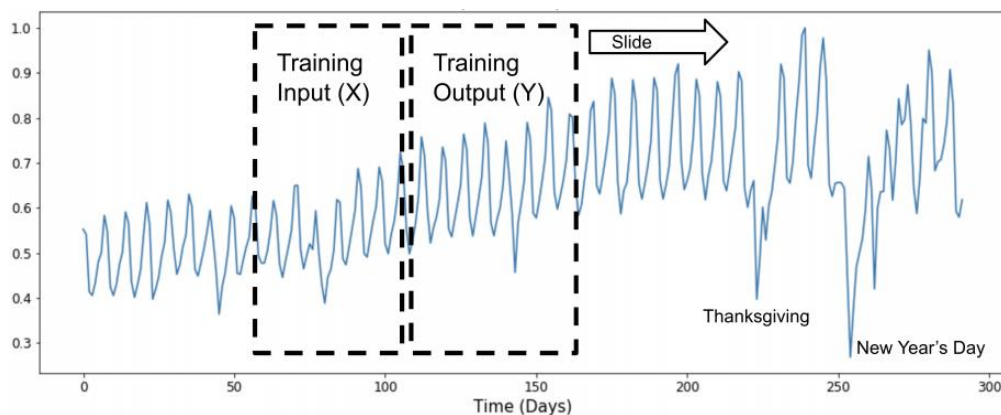


Рис.1.3. Візуалізація розбиття часового ряду двома ковзними вікнами на вхідні та вихідні дані для прогнозування у часі курсу КБ.

Після цього для навчання вхідні дані у такій структурі подаються відповідно на вхід та вихід неймережевого програмного агента МПС прогнозування у часі курсу КБ. Загальна структура неймережевого програмного агента наведена нище на рис.1.4.



Рис.1.4. Загальна структура неймережевого програмного агента мультиагентної програмної системи прогнозування у часі курсу КБ.

Для тестування якості роботи нейромережевого програмного агента щодо прогнозування у часі курсу КБ у роботі за тестову вибірку було обрано останній місяць часу даних курсу КБ. Тобто, дні останнього місяця були розбиті двома ковзними вікнами на вхідні дані на основі яких нейромережевий агент здійснював прогноз у часі курсу КБ та вихідні дані з якими порівнювалися прогнозні дані та обчислювалося середнє значення відносної похибки.

Отже, тепер можна перейти безпосередньо до процесів проектування та розробки (у тому числі й розробки математичного забезпечення) нейромережевих програмних агентів та на їх основі МПС прогнозування у часі курсу КБ.

2. ПРОЕКТУВАННЯ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ МУЛЬТИАГЕНТНОЇ ПРОГРАМНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ КУРСУ КРИПТОВАЛЮТИ БІТКОЙН

2.1. Проектування архітектури МПС прогнозування курсу КБ

Існує велике сімейство нейронних мереж (нейромереж) різної архітектури, наприклад, багат шаровий перцептрон, згорткові нейронні мережі (нейромережі), рекурентні нейронні мережі (нейромережі), нейронні мережі (нейромережі) Кохонена, нейронні мережі (нейромережі) Хопфілда, нейронні мережі (нейромережі) Жордана, нейронні мережі (нейромережі) Елмана, модулярні нейронні мережі (нейромережі), накшталт, автоенкодер (енкодер-декодер): повнозв'язні автоенкодер, згорткові автоенкодер, рекурентні автоенкодер, енкодер рекурентна нейронна мережа (нейромережа), а декодер згорткова нейронна мережа (нейромережа), енкодер згорткова нейронна мережа (нейромережа), а декодер рекурентна нейронна мережа (нейромережа), тощо. Серед цих різних архітектур нейромереж багато можуть використовуватися в основі програмних агентів МПС прогнозування курсу КБ. Кожна із нейромереж має, як свої переваги так і свої недоліки, тому однозначно обґрунтованої відповіді, яку саме архітектуру нейромереж використовувати в основі роботи програмних агентів МПС прогнозування курсу КБ не існує. Виходячи із принципу бритви Оккама та ефективності у розумінні високої точності у даній роботі були використані дві архітектури нейронних мереж, а саме, багат шаровий перцептрон та згорткова нейронна мережа (нейромережа).

Таким чином, в основі проектування програмних агентів МПС прогнозування курсу КБ використано, як перший нейромережевий програмний агент – це багат шаровий перцептрон, а другий нейромережевий програмний агент – це згорткова нейронна мережа (нейромережа). Кожен із двох нейромережевих програмних агентів приймав рішення щодо прогнозування курсу КБ не залежно із

власними затратами обчислювальних ресурсів та величиною похибки (або точності).

Розглянемо архітектуру МПС прогнозування курсу КБ на основі вище наведених нейромережових програмних агентів, яка наведена на рис.2.1, де TS#1 – часовий ряд "Open", що характеризує курс КБ при відкритті торгівлі на криптобіржі, TS#2 – часовий ряд "High", що характеризує найбільше значення курсу КБ протягом дня торгівлі на криптобіржі, TS#3 – часовий ряд "Low", що характеризує найменше значення курсу КБ протягом дня торгівлі на криптобіржі, TS#4 – часовий ряд "Close", що характеризує курс КБ при закритті торгівлі на криптобіржі, MLP – нейромережовий програмний агент на основі нейромережової архітектури багатошаровий перцептрон, CNN – нейромережовий програмний агент на основі нейромережової архітектури згорткової нейромережі, TS Forecast – прогноз часових рядів.

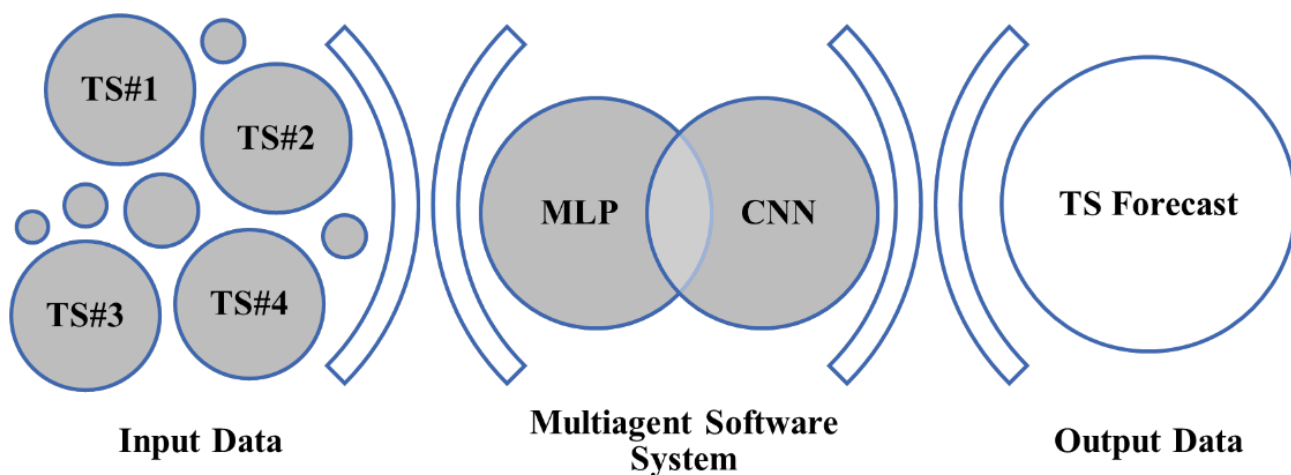


Рис.2.1. Архітектура МПС прогнозування курсу КБ на основі двох нейромережових програмних агентів: багатошарового перцептрона та згорткової нейромережі.

Для переходу з архітектурного рівня проектування МПС прогнозування курсу КБ до конкретної та конструктивної розробки програмного забезпечення нейромережових програмних агентів необхідно апріорі розглянути математичне забезпечення для цих нейромережових програмних агентів.

Таким чином, далі будуть розглянуті, спершу математичне забезпечення програмного агента на основі нейромережевої архітектури багат шаровий перцептрон, а після того математичне забезпечення програмного агента на основі нейромережевої архітектури згорткової нейромережі.

2.2. Математичне забезпечення програмного агента на основі нейромережевої архітектури багат шаровий перцептрон

Передусім корисно розглянути, яку попередню обробку вхідних даних необхідно виконувати перед їх подачею на вхід програмного агента на основі нейромережевої архітектури багат шаровий перцептрон.

Попередня обробка (підготовка) вхідних даних для програмного агента на основі нейромережевої архітектури багат шаровий перцептрон містить наступні опції:

- кодування категоріальних (номінальних) ознак,
- видалення викидів,
- видалення пропущених значень,
- стандартизація (нормалізація) даних.

2.2.1. Модельна візуалізація біологічного нейрона

Відомо, що розробка нових прогресивних технологій базується на нових наукових методах, які одержуються вченими при дослідженні тих чи інших об'єктів та процесів природи.

Аналогічно відбувається і в сучасній сфері розробки нових архітектур нейронних мереж до яких відноситься багат шаровий перцептрон. Чим глибше і

детальніше вивчається біологічна нейронна мережа (біологічний коннектом) тим вченими одержуються кращі архітектури штучних програмних нейронних мереж. Елементом біологічного коннектома є біологічний нейрон. Чим змістовніші та предметніші знання нейрофізіології функціонування біологічного нейрона тим краще проектуються та розробляються штучні програмні нейрони. Саме тому детальний розгляд біологічної моделі біологічного нейрона дає змогу будувати математичну модель його роботи, яка слугуватиме основою штучного програмного нейрона та дає змогу розглянути основи його математичного забезпечення.

Таким чином, доцільно детально розглянути модельну візуалізацію біологічного нейрона, яка наведена на рис.2.2.

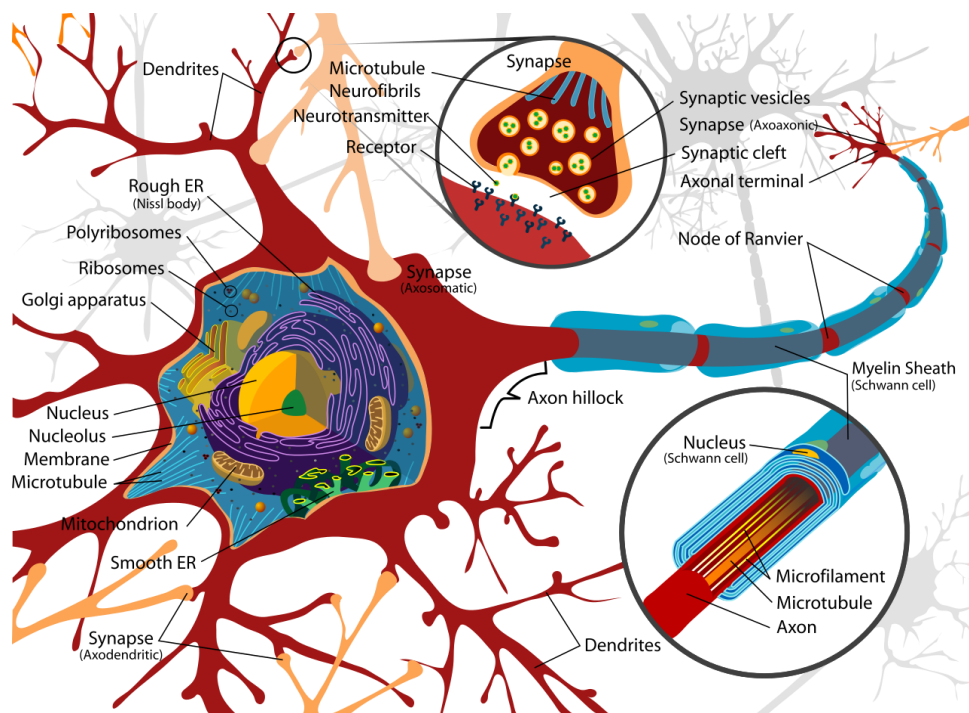


Рис.2.2. Модельна візуалізація біологічного нейрона.

2.2.2. Графова модель штучного нейрона

У попередньому пункті даної роботи зазначено, що детальний розгляд біологічної моделі біологічного нейрона дає змогу будувати оптимальнішу

математичну модель його роботи, яка слугує основою штучного програмного нейрона та дає змогу розглядати основи його математичного забезпечення.

У цьому пункті розглянуто основи математичного забезпечення для штучного програмного нейрона та для довільних штучних нейронних мереж різної архітектури, зокрема, багатошарового перцептрона.

Ці основи математичного забезпечення дають змогу виконувати не лише різного роду обчислення на основі штучних нейронних мереж, але й доволі зручно їх візуалізувати.

Візуалізація та обчислювальні процеси в штучному нейроні та штучних нейронних мережах добра формалізуються за допомогою теорії графів. Тому нище наведено графічні компоненти теорії графів, які використовуються для візуалізації структури та обчислювального процесу у штучному нейроні (рис.2.3).

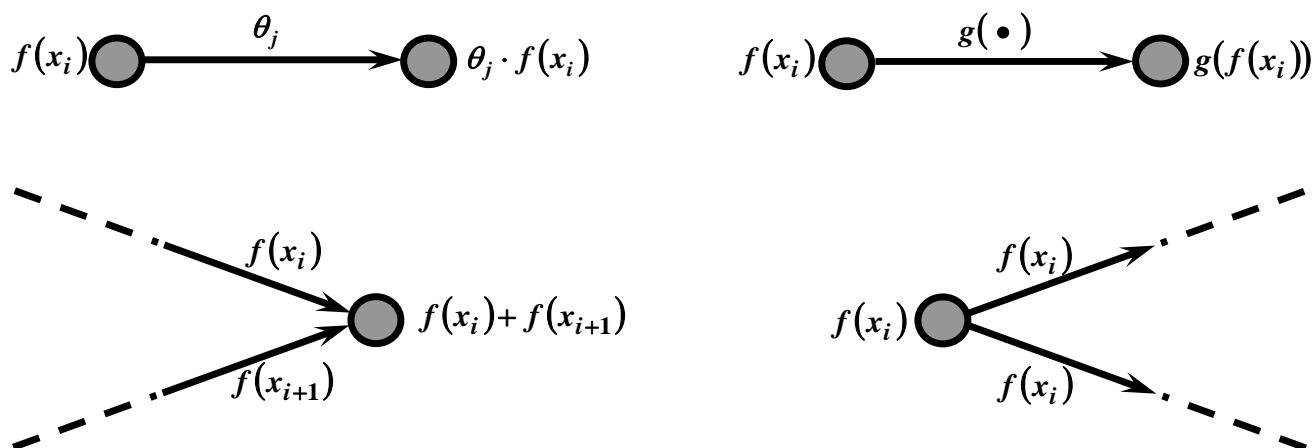
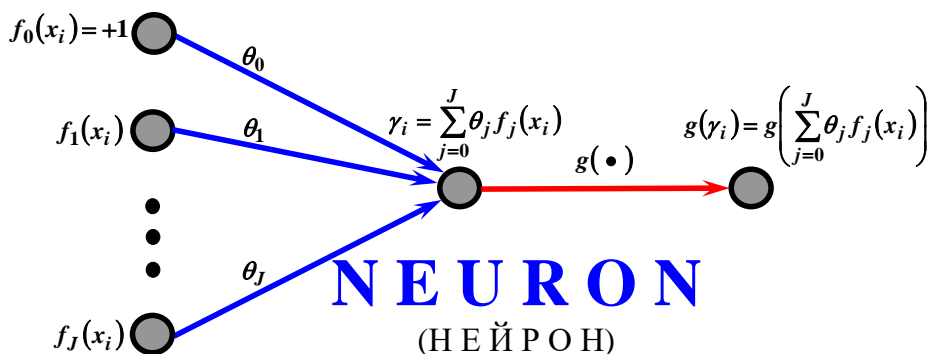


Рис.2.3. Графічні компоненти теорії графів для формалізації та візуалізації структури і обчислень у штучному нейроні (граф передачі даних).

Граф передачі даних для одного штучного нейрона (модель нейрона Маккалока-Піттса, 1943 рік) наведено нище:



Уоррен Маккалок



Уолтер Пітс

➔ – синапси, ➔ – аксон, $f_0 = 1, f_1, \dots, f_J$ – ознаки, $\theta_0, \theta_1, \dots, \theta_J$ – синаптичні ваги, $g(\bullet)$ – функція активації,

$$g(\gamma_i) = g\left(\sum_{j=0}^J \theta_j f_j(x_i)\right) \equiv a(\theta, x_i).$$

Як видно з вище наведеного математичного забезпечення, у формі графу передачі даних, штучний програмний нейрон здатен виконувати різного роду нелінійні перетворення над даними, що забезпечується наявністю функції активації. Окрім цього з даного математичного забезпечення випливає, що штучний нейрон, загалом, представляє собою універсальний обчислювальний засіб, тобто його математичне забезпечення дає можливість вирішувати багатокласові задачі, задачі регресії та задачі кластерного аналізу. Так, наприклад, із графу передачі даних, як математичного забезпечення штучного програмного нейрона видно, що можливе вирішення задач лінійної регресії, логістичної регресії, багатокласової класифікації, зокрема, бінарної та тернерної класифікації, а також задач кластеризації.

Розгляньмо порівняння біологічного нейрона та штучного програмного нейрона, що представлений формально математично у вигляді графу передачі даних та зробимо висновки з цього.

2.2.3. Біологічний нейрон vs. штучний нейрон

Порівняльний аналіз між функціональними характеристиками біологічного нейрона та штучного нейрона є надзвичайно важливим питанням. Оскільки чим краще (з більшою точністю) математичне забезпечення штучного нейрона описує функціонування біологічного нейрона тим високоякіснішими є штучні нейронні мережі на основі штучних нейронів.

Проте, не все так просто. Останні досягнення у сфері модулярних штучних нейронних мереж, архітектури яких стають все менше схожими на архітектури конкретних ділянок – нейрональних груп біологічного коннектома, на подив показують в експериментах результати високої якості, кращі у порівнянні з біологічними нейрональними групами біологічного коннектома.

Ці експериментальні факти вказують на те, що нові архітектури нових штучних нейронних мереж, не дивлячись на те, що вони не подібні на біологічні нейромережі, будуть часто використовуватися для вирішення різного роду прикладних задач, наприклад, прогнозування курсу різного роду криптоактивів.

Проте, на разі, залишається актуальним та доцільним розглянути порівняння біологічного та штучного нейронів, яке наведено на рис.2.4.

2.2.4. Функції активації штучного нейрона

Принциповим елементом у математичному забезпечення штучного нейрона є наявність функції активації. Функція активації фактично є перетворенням зваженої суми синаптичних сигналів, які потрапляють нейрону на вхід. Відомо, що існує велика кількість різного виду функцій активації. В залежності від виду функції активації нейрон здатен виконати різного роду перетворення над зваженою сумою

від синаптичних щілин своїх дендритів, зокрема, як лінійні перетворення так і не лінійні перетворення над вхідними даними. Найбільш поширеною та найпростішою є функція активація – функція Хевісайда. Дана функція активації приймає значення виключно нуль або одиниця, що еквівалентно відповідно пасивному стану та активному стану штучного нейрона.

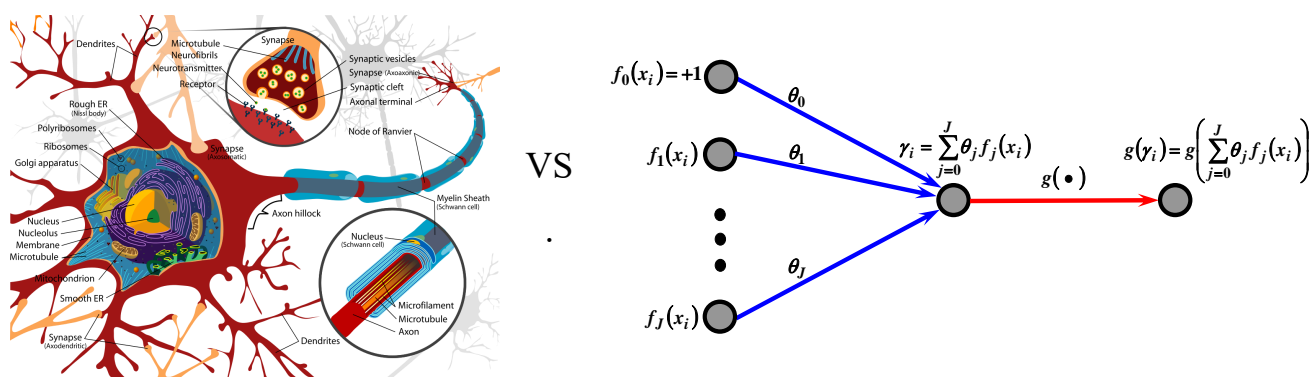


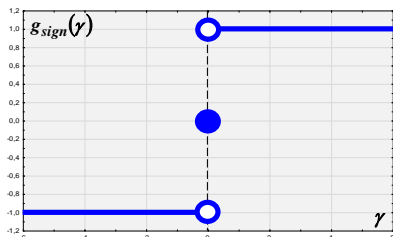
Рис.2.4. Біологічний нейрон vs. Штучний нейрон.

З однієї сторони – це описує просту роботу штучного нейрона і швидко обчислюється, але з іншої сторони низька чутливість штучного нейрона до вхідних сигналів (вхідних даних). Окрім функції активації у вигляді функції Хевісайда, існує лінійна функція активації. Її перевага полягає у тому, що вона більш чутлива до вхідних сигналів (вхідних даних) штучного нейрона, а також може реалізовувати не лише лінійну класифікацію даних, але й лінійну регресію. Тобто покриває вже дві області машинного навчання, а саме, класифікацію та регресію. Поряд із вище названими видами функцій активації існують ще інші, які також мають ряд власних переваг у процесі обробки даних штучним нейроном. До них відносяться: функція активації – гіперболічний тангенс (біполярний сигмоїд), сигмоїдна функція (уніполярний сигмоїд). Основна їх перевага полягає у можливості реалізації м'якої класифікації даних, коли ті знаходяться поблизу поверхні розмежування класів у багатовимірному просторі даних. Як правило в задачах класифікації функції активації гіперболічний тангенс (біполярний сигмоїд) та сигмоїдна функція (уніполярний сигмоїд) дають найвищу точність вирішення задачі. Тому, з точки зору прикладного програмування такого роду інформація представляє важливу

еврістику. Також важливе місце в прикладному програмуванні займає Гауссова функція активації, особливо в задачах в яких дані займають нелінійні кластерні області багатовимірного простору даних.

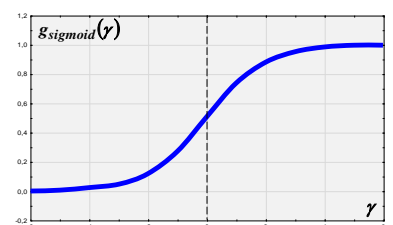
Підводячи підсумок під вище зазначеним варто зазначити важливе значення, яке посідають функції активації у математичному забезпеченні штучного нейрона.

Розгляньмо деякі види функцій активації, що використовуються у математичному забезпеченні штучних нейронів:



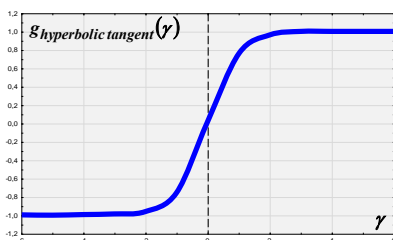
кусково-постійна функція

$$g_{\text{sign}}(\gamma) = \begin{cases} +1, & \gamma > 0 \\ 0, & \gamma = 0 \\ -1, & \gamma < 0 \end{cases}$$



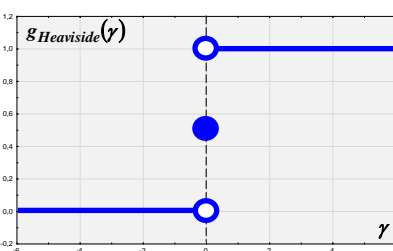
сигмоїдна функція, уніполярний сигмоїд

$$g_{\text{sigmoid}}(\gamma) = \frac{1}{1 + e^{-\text{const} \cdot \gamma}}$$



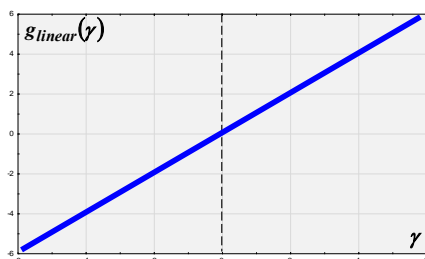
гіперболічний тангенс, біполярний сигмоїд

$$g_{\text{hyperbolic tangent}}(\gamma) = \text{th}(\gamma)$$



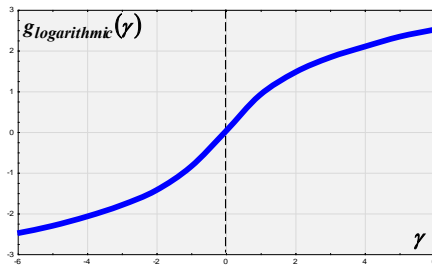
функція Хевісайда

$$g_{\text{Heaviside}}(\gamma) = \begin{cases} 1, & \gamma > 0 \\ 1/2, & \gamma = 0 \\ 0, & \gamma < 0 \end{cases}$$



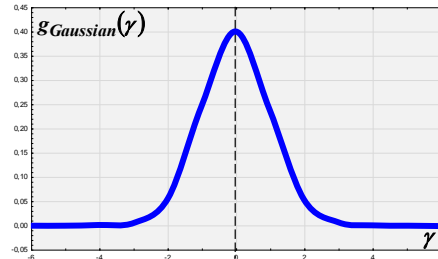
лінійна функція

$$g_{\text{linear}}(\gamma) = \gamma$$



логарифмічна функція

$$g_{\text{logarithmic}}(\gamma) = \ln\left(\gamma + \sqrt{\gamma^2 + 1}\right)$$



Гауссова функція

$$g_{\text{Gaussian}}(\gamma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

2.2.5. Програмний агент на основі нейромережевої архітектури багат шаровий перцептрон

При розробці різного роду програмних агентів для різних задач одного штучного нейрона не достатньо. На практиці використовують побудовані з допомогою них штучні нейронні мережі різної архітектури для найрізноманітніших задач. В принципі використовуючи штучні нейрони з їх нелінійними функціями активації можна побудувати довільну архітектуру штучної нейронної мережі для вирішення будь-якої алгоритмічно розв'язної задачі. Таким чином, штучні нейронні мережі є універсальною обчислювальною парадигмою. Хоча, заради справедливості, необхідно відзначити, що нейромережева парадигма не завжди є оптимальною в обчислювальному розумінні. В останні роки надзвичайно широкого прикладного використання набули модулярні штучні нейронні мережі, які представляють собою надвеликі мета-мережі, у яких атомами є окремі архітектури штучних нейронних мереж. Такі модулярні штучні нейронні мережі відносять до нової галузі нейромережевих технологій – глибокого навчання.

Базовим елементом багатьох видів модулярних штучних нейронних мереж, що відносяться до технології глибокого навчання є штучна нейронна мережа –

багатошаровий перцептрон.

Багатошаровий перцептрон використовується, як нейромережевий програмний агент в МПС прогнозування курсу КБ. Неподільними елементами будови багатошарового перцептрона є штучні нейрони. Логічно, що математичне забезпечення багатошарового перцептрона базується на математичному забезпеченні штучного нейрона. Різні архітектури багатошарового перцептрона мають різні математичні формалізації, хоча основа математичного забезпечення їх одна. Розгляньмо базові види багатошарового перцептрона та їх застосування для того чи іншого виду задач.

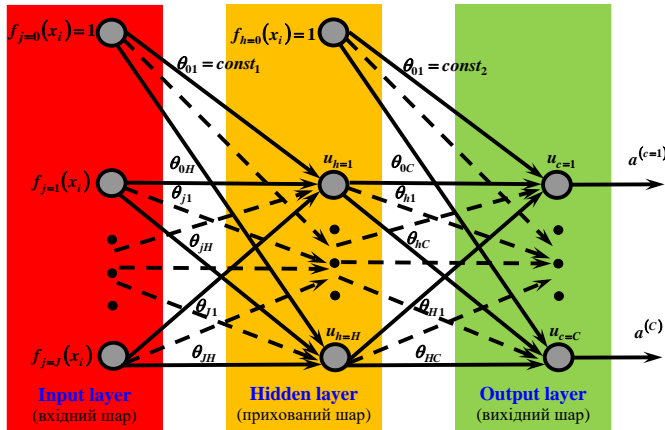
Загалом, перш за все потрібно зазначити, що нейронна мережа багатошаровий перцептрон – універсальна нейронна мережа в обчислювальному розумінні, а саме, у сфері задач класифікації та регресії. Тобто, довільну прикладну задачу класифікації чи регресії, яка алгоритмічно розв'язувана, можна вирішити за допомогою багатошарового перцептрона.

Багатошаровий перцептрон має цілий ряд гіперпараметрів завдяки яким можна керувати його архітектурою. До основних відносяться такі гіперпараметри: кількість нейронів у шарі, кількість шарів, вид функції активації, метод навчання. Як правило, евристична інформація вказує на те, що кількість шарів (прихованих шарів) не варто вибирати більше трьох, тобто гіперпараметром кількість шарів (прихованих шарів) керувати в діапазоні від одного до трьох; кількістю нейронів у кожному з шарів керувати в діапазоні від декілька десятків до декілька сотень; з різних видів функцій активації пріоритетними у виборі можна брати сигмоїдну функцію активації (уніполярний сигмоїд) або "Relu"; із пулу методів навчання нейронної мережі доцільно обирати метод стохастичного градієнтного спуску або ж довільний інший метод навчання.

Важливим моментом в різних архітектурах нейронних мережі, наприклад, багатошаровий перцептрон є кількість нейронів у вихідному шарі. В задачах регресії у вихідному шарі використовується завжди один нейрон. В задачах класифікації використовується стільки нейронів скільки є класів у задачі класифікації. Наприклад, для задач бінарної класифікації у вихідному шарі

багатошарового перцептрона є два нейрони, для задач тернарної класифікації у вихідному шарі є три нейрони, в задачах n-арної класифікації є n нейронів.

Нище, як приклад, наведено приклади багатошарових перцептронів з одним та двома прихованими шарами (рис.2.5 та рис.2.6).



Френк Розенблатт,
1957 рік.

Рис.2.5. Багатошаровий перцептрон з одним прихованим шаром.

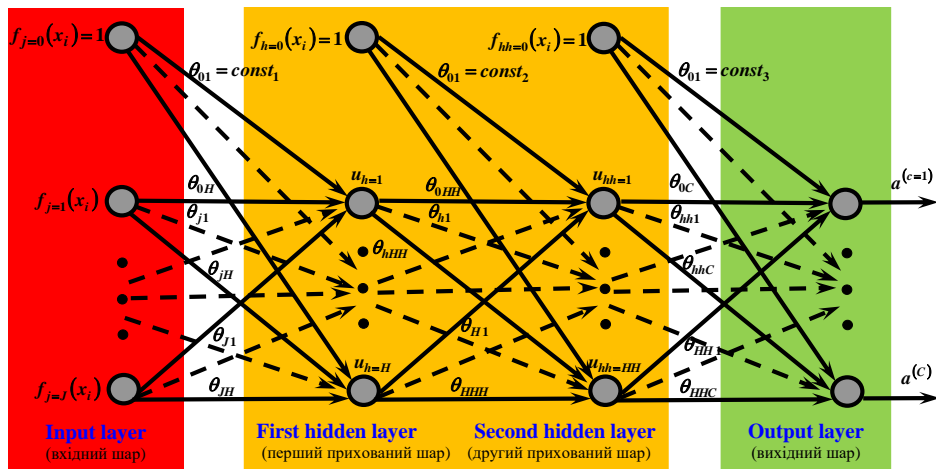


Рис.2.6. Багатошаровий перцептрон з двома прихованими шарами.

2.2.6. Навчання програмного агента на основі нейромережевої архітектури багатошаровий перцептронпо по методу алгоритму оберненого поширення похибки

Однією із найважливіших сутностей при навчанні нейронної мережі багатoshаровий перцептрон, як і нейронної мережі довільної іншої архітектури є організація процедури навчання. В основі теоретичних засад навчання використовується алгоритм зворотнього поширення похибки. Для наочності і зручності розгляньмо черговий раз графову модель штучного нейрона, яка наведена на рис.2.7.

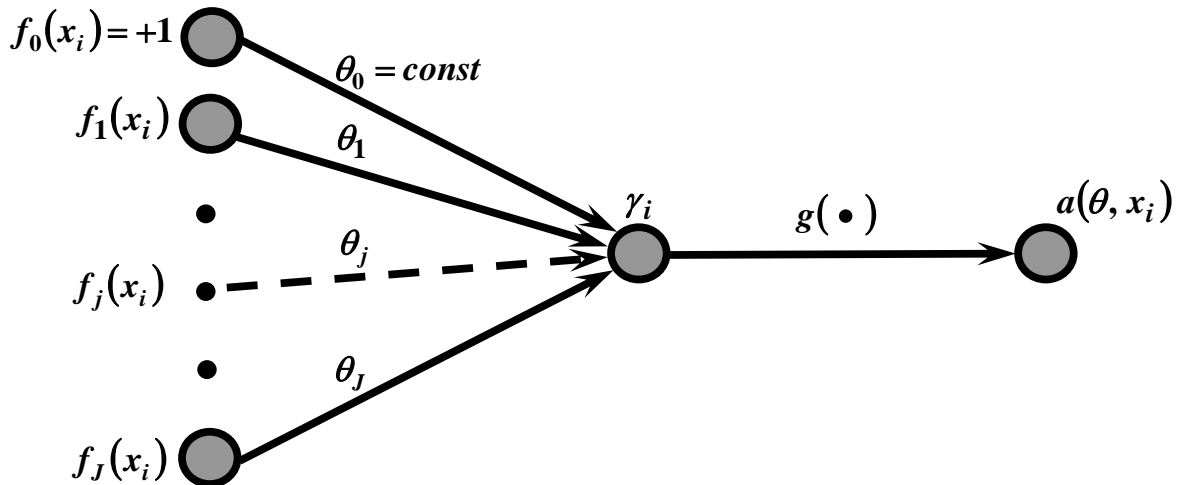


Рис.2.7. Графова модель штучного нейрона.

Концептуальна суть алгоритму оберненого розповсюдження похибки полягає в наступному.

Розгляньмо формулу розрахунку похибки:

$$e(x_i) = a(\theta, x_i) - y(x_i) = (g(\gamma_i) - y(x_i)) = \left(g \left(\theta_0 \cdot f_0(x_i) + \sum_{j=1}^J \theta_j \cdot f_j(x_i) \right) - y(x_i) \right).$$

На основі розрахунку похибки розраховується функція втрати (функція ціни):

$$L(\theta, x_i) = \frac{1}{2} e(x_i)^2 = \frac{1}{2} (g(\gamma_i) - y(x_i))^2 = \frac{1}{2} \left(g \left(\theta_0 \cdot f_0(x_i) + \sum_{j=1}^J \theta_j \cdot f_j(x_i) \right) - y(x_i) \right)^2,$$

де $f(x_0) = +1$, $\theta_0 = const$,

$$\theta^{(t+1)} = \theta^{(t)} - \eta \cdot \text{grad}_{\theta} L(\theta, x_i)$$

$$\begin{aligned} \theta^{(t+1)} &= \theta^{(t)} - \eta \cdot \underbrace{\frac{\partial L(\theta, x_i)}{\partial e(x_i)}}_{e(x_i)} \cdot \underbrace{\frac{\partial e(x_i)}{\partial g(\gamma_i)}}_1 \cdot \underbrace{\frac{\partial g(\gamma_i)}{\partial \gamma_i}}_{g'(\gamma_i)} \cdot \underbrace{\frac{\partial \gamma_i}{\partial \theta}}_{f(x_i)} = \theta^{(t)} - \eta \cdot e(x_i) \cdot 1 \cdot g'(\gamma_i) \cdot f(x_i) = \\ &= \theta^{(t)} - \eta \cdot e(x_i) \cdot g'(x_i) \cdot f(x_i) \\ \theta^{(t+1)} &= \theta^{(t)} - \eta \cdot e(x_i) \cdot g'(x_i) \cdot f(x_i) \end{aligned}$$

де $\theta^{(t=0)} = \mathbf{0}$ або $\theta^{(t=0)} = \text{random}\left(-\frac{1}{2J}, \frac{1}{2J}\right)$,

$\eta \in (0, 1)$ або $\eta^{(t)} = 1/t$ – швидкість навчання.

Як видно з вище описаного, алгоритм оберненого розповсюдження похибки дає змогу одержати розрахункові ітераційні формули для оновлення числових значень синаптичних ваг у нейронній мережі при її навчанні.

Якщо видозмінювати вид формули розрахунку функції втрат (функції ціни), то видозмінюються і вид ітераційних формул розрахунку синаптичних ваг при навчанні, відповідно може бути інше числове значення похибки у знаннях нейронної мережі багатошаровий перцептрон при тестуванні її на тестовій вибірці.

2.2.7. Перший нейромережевий програмний агент МПС прогнозування курсу КБ

Таким чином, в основі проектування та математичного забезпечення першого програмного агента на основі нейромережевої архітектури багатошаровий перцептрон МПС прогнозування курсу КБ використовується штучна програмно-орієнтована нейромережа – багатошаровий перцептрон із навчання на основі алгоритму зворотнього поширення похибки.

2.3. Математичне забезпечення програмного агента на основі нейромережевої архітектури згорткової нейромережі

Передусім корисно розглянути структури вхідних даних, які можна подавати на вхід програмних нейромережевих агентів з архітектурою у вигляді згорткових нейромереж, а також структури даних, які мають дані в середині згорткових нейромереж. Такі структури даних називаються тензорними структурами даних, по простому – багатовимірні масиви даних. Види даних із тензорною структурою наведені нище на рис.2.8.

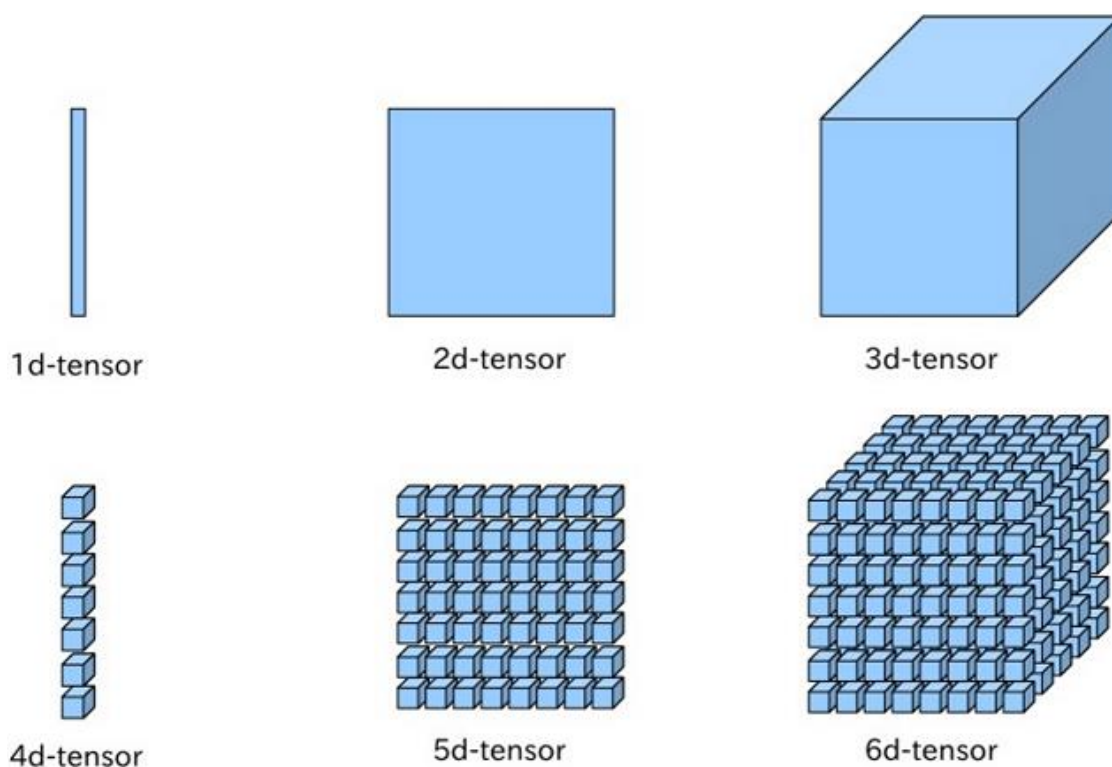


Рис.2.8. Тензорна структура даних.

Розглядатимемо проектування та математичне забезпечення програмного агента МПС на основі архітектури згорткової нейромережі у вигляді конвеєра операцій (шарів), які представляють собою композицію функціональних перетворень над даними, як наведено на рис.2.9.



Рис.2.9. Програмний агент на основі нейромережевої архітектури згорткової нейромережі у вигляді конвеєра операцій (шарів).

Для початку розглянемо природну, біологічну нейромережу зору, як частини біологічного коннектома, що обробляє візуальні дані у вигляді конвеєра операцій (шарів), як наведено на рис.2.10.

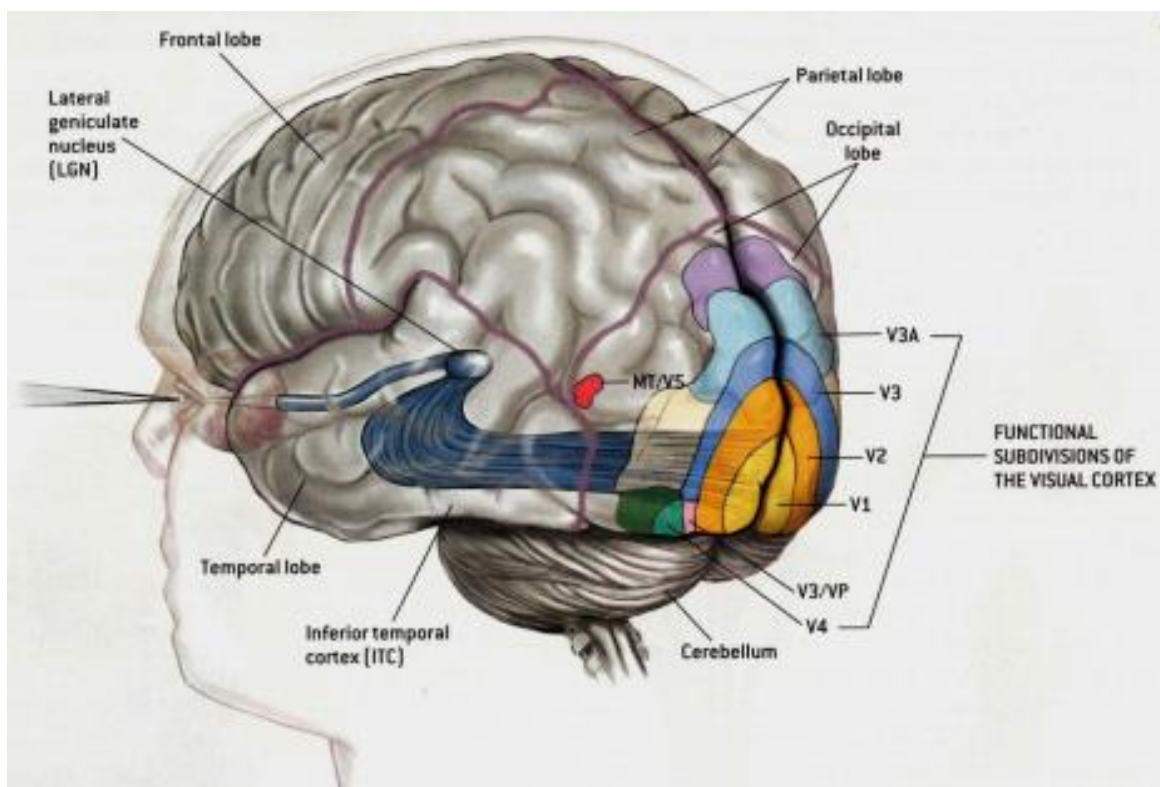


Рис.2.10. Функціональні ділянки біологічної зорової кори (біологічної нейромережі, біологічного коннектома).

Для наочності представлення штучної зорової системи у вигляді штучної програмної згорткової нейромережі нище наведено асоціації ділянок біологічної зорової нейромережі, як частини біологічного коннектома та конвеєра операцій (шарів) штучної згорткової нейромережі (рис2.11).

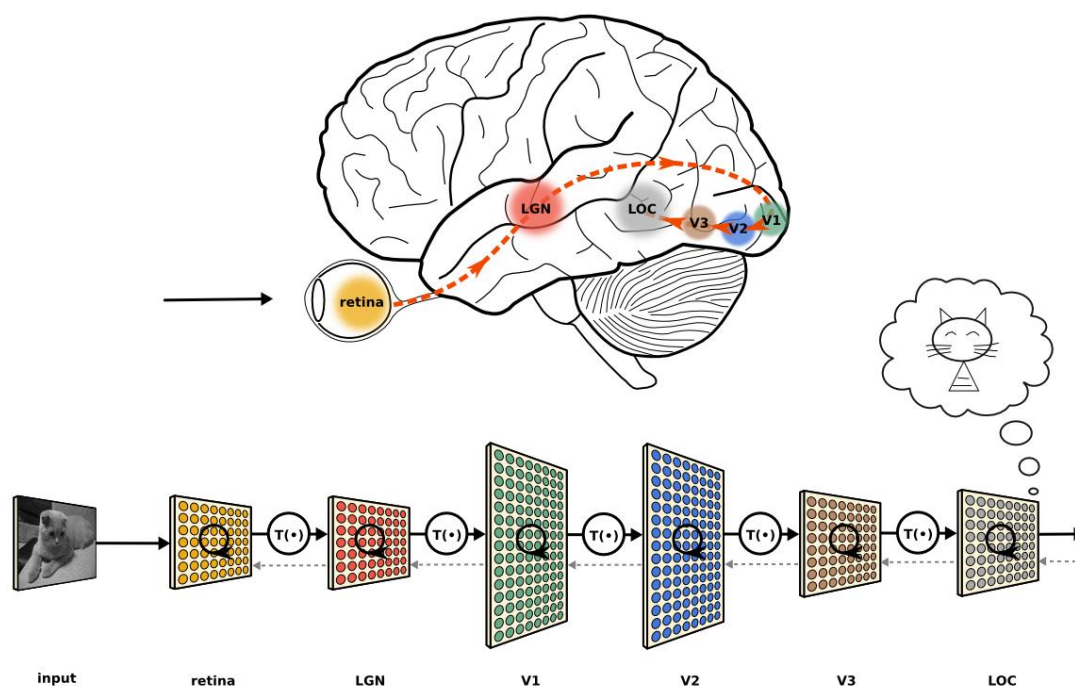


Рис.2.11. V1 та V2 використовуються в базових функціях зору; V3 та V5 використовуються у виявленні рухів, просторовій локалізації, рухів очей та рук; V3 використовується в сприйнятті форми; V4 використовується при обробці кольорових зображень.

Архітектура написання програмного коду щодо проектування та практичної реалізації штучної згорткової нейромережі (штучної зорової системи) у вигляді конвеєра операцій (шарів) обробки даних має вид, який наведений на рис.2.12.

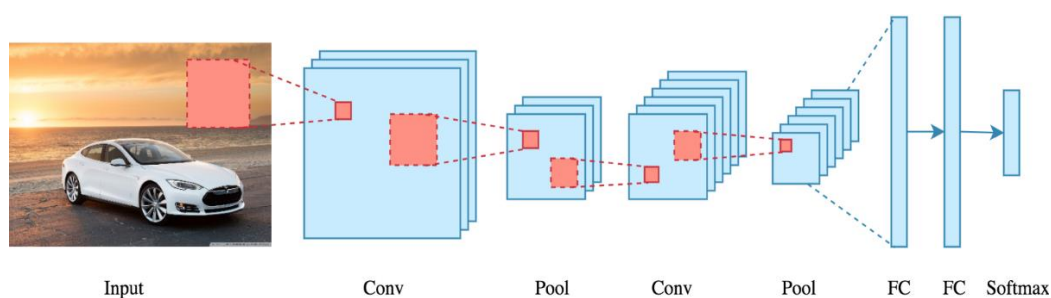


Рис.2.12. Візуалізація штучної згорткової нейромережі у вигляді конвеєра операцій (шарів) обробки даних.

Операція (шар) згортки.

Суть операції (шару) згортки полягає в наступному. Ядерна функція або ще для простоти називають ядро, а також фільтр покроково сканує вхідний масив

даних зліва на право і зверху вниз. Крок сканування є гіперпараметром, тому його значенням можна керувати, як по горизонталі так і по вертикалі. Розмір ядра (фільтра) також є гіперпараметром, тому ним також можна керувати вздовж кожної із координатних вісей вимірності масиву вхідних даних. Значення ядра (фільтру) генерує генератор псевдовипадкових чисел. Також гіперпараметром є число ядер (фільтрів) яким можна керувати. Операція (шар) згортки дає можливість ідентифікувати максимально інформативні елементи даних в масиві вхідних даних завдяки скануванню та псевдовипадковим значенням ядра (фільтру). Для наочності тлумачення дії операції (шару) згортки на дані наведено нище відповідні візуалізації: рис.2.13, рис.2.14, рис.2.15, рис.2.16.

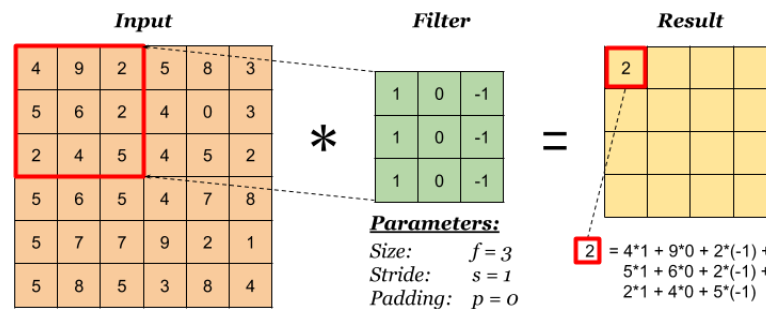


Рис.2.13. Візуалізація дії накладання ядра (фільтру) на вхідний масив даних – поелементне множення та додавання результату.

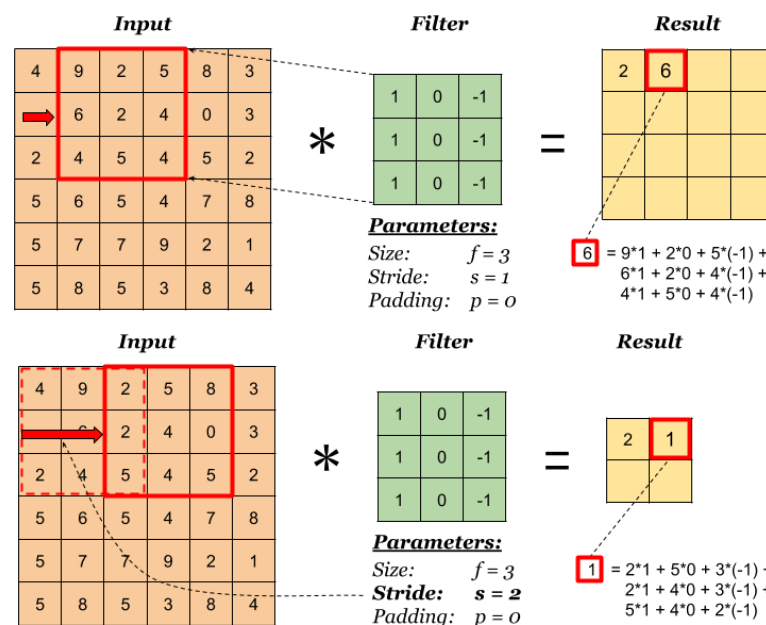


Рис.2.14. Візуалізація дії гіперпараметру крок при скануванні вхідного масиву даних.

виконання операції (шару) об'єднання пулу розмірність вихідних даних зменшується. На рис.2.17 наведено візуалізацію дії операції (шару) об'єднання пулу.

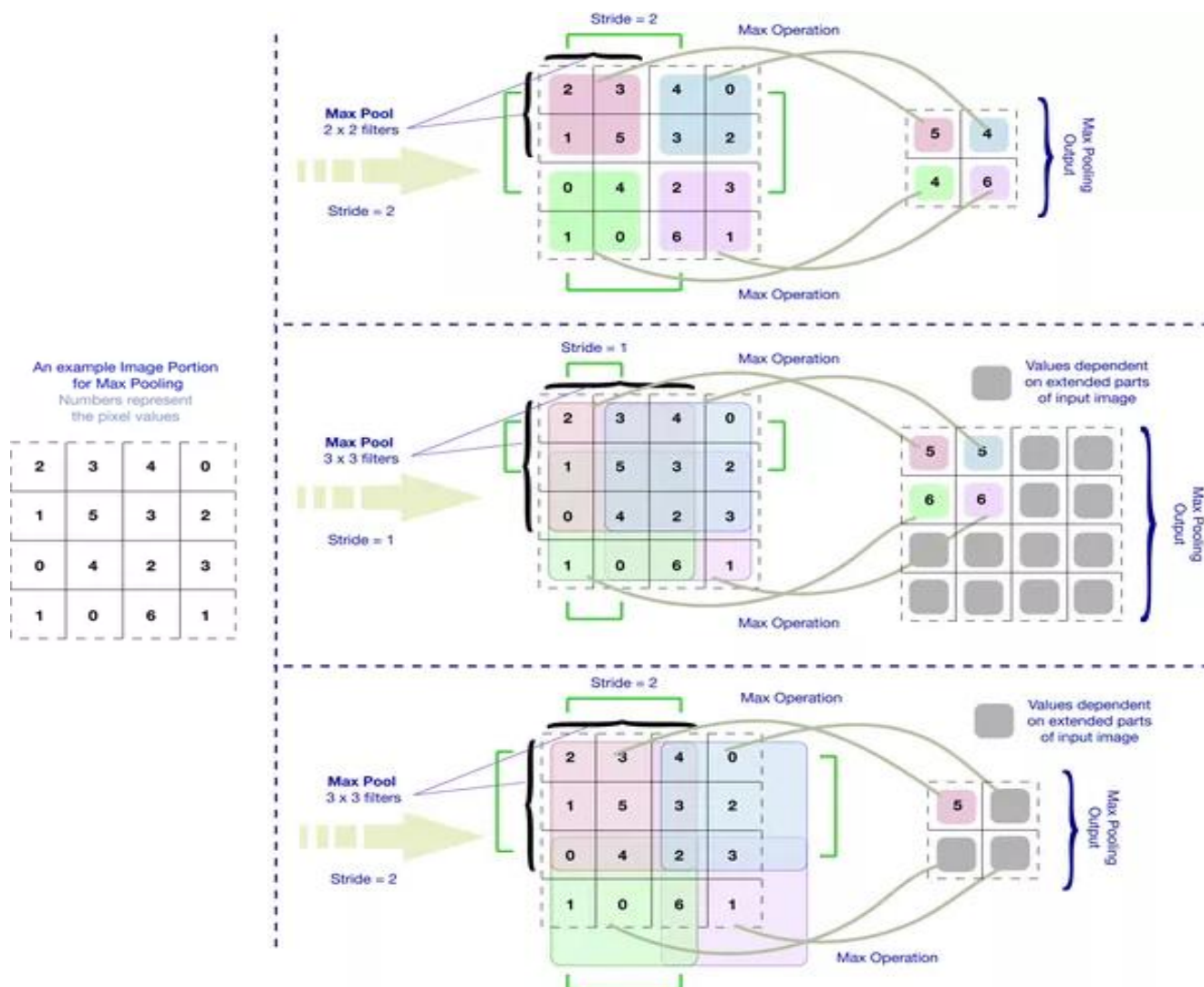


Рис.2.17. Візуалізація дії операції (шару) об'єднання пулу.

Операція (шар) виключення даних.

Операція (шар) виключення даних – це метод випадкового відключення даних – випадкового відключення певних нейронів у шарі на етапі навчання. Операція (шар) виключення організовує процес вивчення сутності вхідних даних з різних сторін, тобто вивчення концептуальної суті по-різному. Робота операції (шару) виключення даних може бути організована у різний спосіб. Результати дії операції (шару) виключення даних наведені на рис.2.18 та рис.2.19.

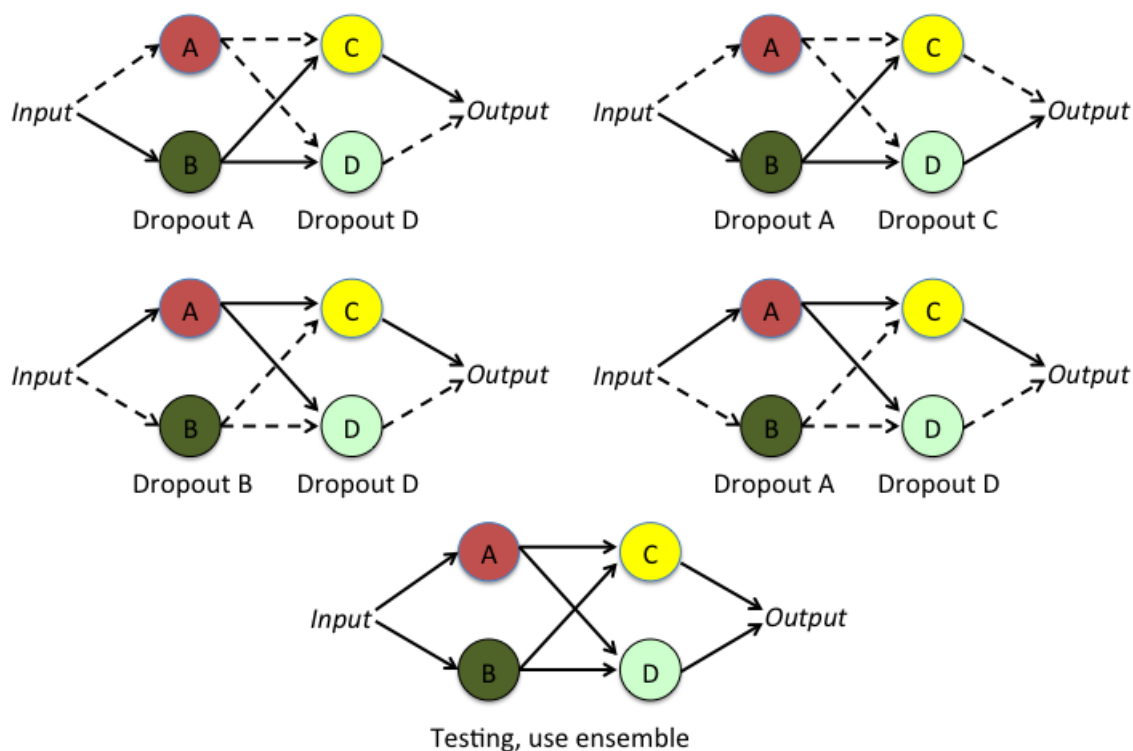


Рис.2.18. Візуалізація дії операції (шару) повнозв'язне виключення даних.

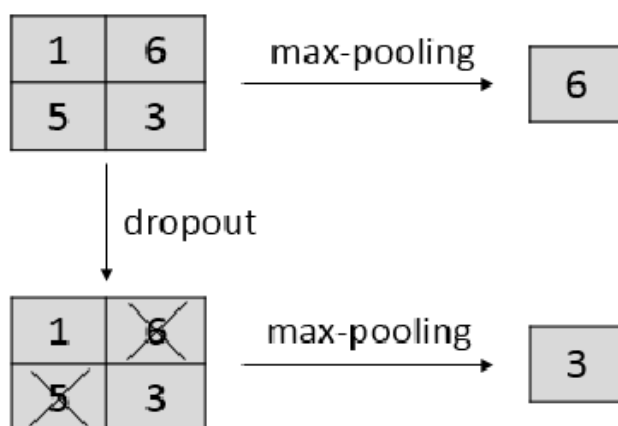


Рис.2.19. Візуалізація дії операції (шару) макс-пулівське виключення даних.

Операція (шар) вирівнювання даних.

Операція (шар) вирівнювання даних займає проміжне місце у конвеєрі згорткової нейромережі. А саме, між блоком операцій (шарів) – видобування ознак та мета-ознак із даних з одної сторони та блоком операцій (шарів) – прийняття рішень з іншої сторони, тобто коли потрібно багатовимірні дані перетворити в одновимірний масив даних. Візуалізація дії операції (шару) вирівнювання даних наведена на рис.2.20.

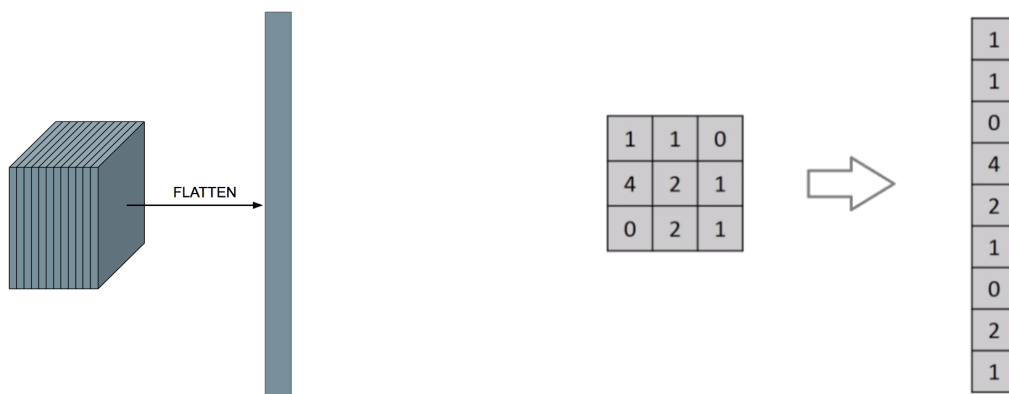


Рис.2.20. Візуалізація операції (шару) вирівнювання даних – перетворення багатовимірних даних в одновимірний масив даних.

Операція (шару) нормалізації даних.

Аналітичний опис операції (шару) нормалізації даних має вид

$$\hat{x}_i \leftarrow \frac{x_i - m_x}{\sqrt{\sigma_x^2}}$$

Візуалізація дії даної операції (шару) нормалізації даних наведена на рис.2.21.

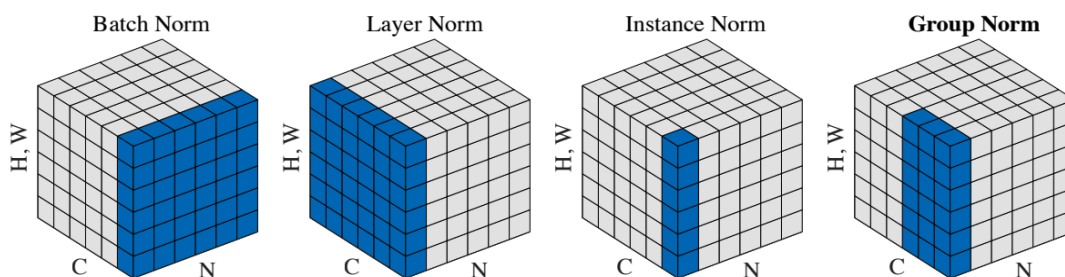


Рис.2.21. Візуалізація дії операції (шару) нормалізації даних.

Методи нормалізації. На рисунках наведено тензор карт ознак із кількістю пакетів об'єму N, вісь каналів позначено, як C, координати (H,W) – це просторові вісі. Синім кольором розфарбовані пікселі, які нормовані певними середніми значеннями та середньоквадратичними відхиленнями, обчислення шляхом додавання значень цих пікселів.

Пакетна нормалізація зазвичай використовується після повнозв'язних або згорткових операцій (шарів) та перед операцією (шаром) нелінійність.

Гіперпараметр – число epoch навчання згорткової нейромережі.

Окрім вище наведених операцій (шарів) згорткової нейромережі важливим гіперпараметром для керування якістю її роботи є гіперпараметр число epoch навчання. З одного боку, чим більша кількість epoch навчання, тим параметри – синаптичні ваги згорткової нейромережі задають кращу точність її роботи, але з іншого боку недоліком є збільшення затрат обчислювальних ресурсів на навчання згорткових нейромережі та застосування її в режимі роботи прийняття рішень, а також втрачається згортковою нейромережею властивість до узагальнення вихідних даних.

Нище, на рис.2.22, наведено залежність похибки прийняття рішень згортковою нейромережею від кількості epoch навчання із врахуванням дії різних операцій (шарів) конвеєра.

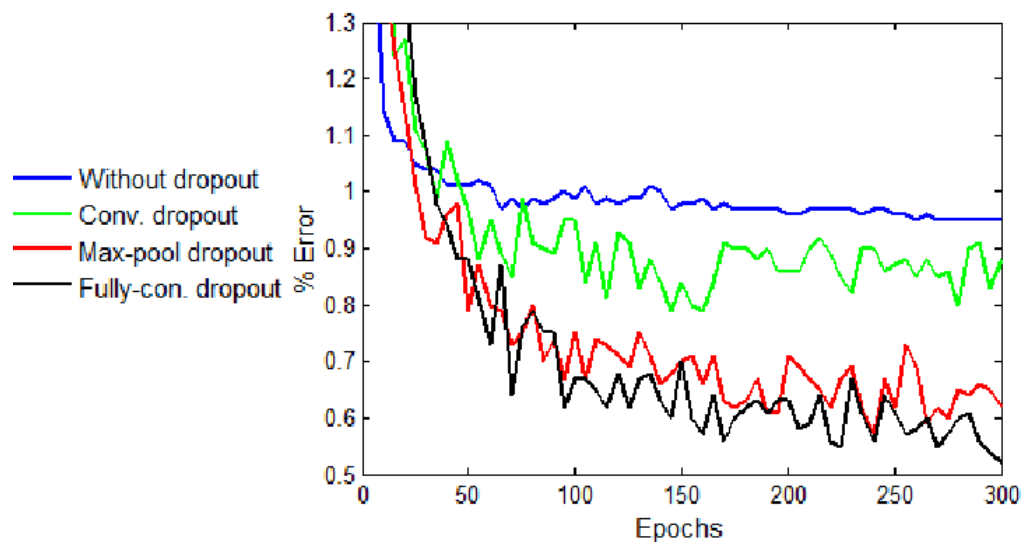


Рис.2.22. Залежність похибки прийняття рішень згортковою нейромережею від кількості epoch навчання із врахуванням дії різних операцій (шарів).

Таким чином, в основі проектування та математичного забезпечення другого програмного агента на основі нейромережевої архітектури згорткової нейромережі МПС прогнозування курсу КБ використовується конвеєр операцій (шарів) обробки тензорної структури даних.

3. ПРАКТИЧНА ПРОГРАМНА РЕАЛІЗАЦІЯ МУЛЬТИАГЕНТНОЇ ПРОГРАМНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ КУРСУ КРИПТОВАЛЮТИ БІТКОЙН

МПС для прогнозування курсу КБ складається з двох нейромережових програмних агентів, а саме, нейромережі багатосарового перцептрона та згорткової нейромережі.

Кожен з нейромережових програмних агентів МПС незалежно навчається та тестується на колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ, де часовий ряд "Open" – характеризує курс КБ при відкритті торгівлі на криптобіржі, часовий ряд "High" – характеризує найбільше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Low" – характеризує найменше значення курсу КБ протягом дня торгівлі на криптобіржі, часовий ряд "Close" – характеризує курс КБ при закритті торгівлі на криптобіржі. Рядковий індекс "Date" – набуває значень моментів часу.

Кожен з нейромережових програмних агентів МПС дає власний незалежний прогноз у часі курсу КБ. Також кожен з них при цьому має своє середнє значення відносної похибки прогнозу. Середнє значення відносної похибки прогнозу є важливою та інформативною величиною. У даній МПС для прогнозування курсу КБ нейромережові програмні агенти дають власні прогнози і ці вихідні дані фінансового інструменту використовуються для потреб управління інвестиціями у КБ. Але варто відзначити, що потенційно МПС – фінансовий інструмент, може реалізовувати інші вихідні дані, наприклад, шляхом простого голосування нейромережових програмних агентів або зваженого голосування нейромережових програмних агентів. У випадку, якщо розглядати зважене голосування нейромережових програмних агентів, то, як вагові коефіцієнти їх голосу можуть використовуватися величини, що дорівнюють оберненій величині до середнього значення відносної похибки прогнозу.

Перейдемо до розгляду практичної реалізації кожного із нейромережових програмних агентів МПС, один з яких базується на нейромережі багатосаровий

перцептрон, а інший на згортковій нейромережі, та їх прогнозування у часі курсу КБ.

3.1. Прогнозування курсу криптовалюти біткойн програмним агентом на основі нейромережевої архітектури багатошаровий перцептрон

Як зазначалося вище, перший програмний агент МПС прогнозування курсу КБ є нейромережа багатошаровий перцептрон. У даній роботі використовувалася нейромережа багатошаровий перцептрон з наступними гіперпараметрами: один прихований шар вміщав 100 нейронів, функція активації була вибрана у виді 'relu', методом навчання був обраний 'adam', швидкість навчання становила 0.001, розмір пакетів даних був у режимі замовчування; решта гіперпараметри використовували значення за замовчуванням.

Навчальна та тестові вибірки формувалися наступним чином. Перші 2724 днів у колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ представляли собою навчальну вибірку, а останні 30 днів представляли тестову вибірку.

Таким чином, аналогічно до процедур машинного навчання, програмний агент нейромережевої архітектури багатошаровий перцептрон навчався на навчальній вибірці об'ємом у 2724 дні та тестувався на тестовій вибірці об'ємом у 30 днів.

Як приклад прогнозування у часі був обраний часовий ряд курсу КБ при відкритті торгівлі на біржі "Open" з колекції часових рядів: "Open", "High", "Low", "Close".

На рис.3.1 різними кольорами візуалізовано навчальну та тестову вибірки у колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ для програмного агента нейромережевої архітектури багатошаровий перцептрон. Також на даній

візуалізації наведено прогнозний графік "MLP Forecast" курсу КБ при відкритті торгівлі на біржі.



Рис.3.1. Прогноз у часі "MLP Forecast" курсу КБ на основі програмного агента нейромережевої архітектури багатoshаровий перцептрон для часового ряду "Open" з колекції часових рядів: "Open", "High", "Low", "Close".

Для більшої наочності сприйняття прогнозних у часі значень курсу КБ нейромережевою архітектурою багатoshаровий перцептрон нище наведено масштабовану графічну візуалізацію (рис.3.2).

Також розглядаються відповідні дані у вигляді числового масиву, який візуалізовано наведений на рис.3.3.

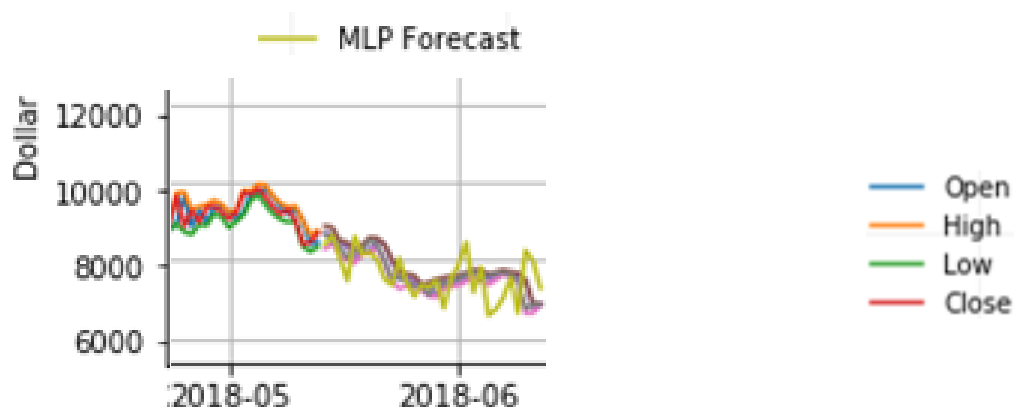


Рис.3.2. Більший масштаб прогнозних значень "MLP Forecast" курсу КБ нейромережевою архітектурою багатoshаровий перцептрон для часового ряду "Open" з тестової колекції часових рядів: "Open", "High", "Low", "Close".

Важливо зазначити, що повно і детально проводилася процедура оцінки точності прогнозу у часі.

Для цього був одержаний часовий ряд відхилень "residuals" між реальним часовим рядом "Open" – реальними даними та прогнозним часовим рядом "MLP Forecast" – прогнозними даними.

Середня відносна похибка прогнозу, яка обчислена на основі часового ряду відхилень "residuals", склала $MLP_MAPE=5.85\%$.

Окрім цього, як відомо, часовий ряд відхилень "residuals" повинен бути максимально подібним до нормального білого шуму. Тому оцінювалися його автокореляційна функція "ACF of residuals" та часткова автокореляційна функція "PACF of residuals", що наведені відповідно на рис.3.4 та рис.3.5.

Date	Open	High	Low	Close	Forecast
2018-05-14T00:00:00.000000000	8688.03	8878.94	8296.14	8675.21	8701.38
2018-05-15T00:00:00.000000000	8675.21	8835.51	8433.43	8474.24	9118.6
2018-05-16T00:00:00.000000000	8474.24	8493.31	8110.33	8340.3	8245.08
2018-05-17T00:00:00.000000000	8340.3	8458.53	7999.59	8058.6	8317.37
2018-05-18T00:00:00.000000000	8058.6	8273.54	7931.43	8240.72	7964.18
2018-05-19T00:00:00.000000000	8240.72	8392.77	8154.15	8234.15	7254.57
2018-05-20T00:00:00.000000000	8234.15	8582.8	8172.18	8520.81	8018.13
2018-05-21T00:00:00.000000000	8520.81	8566.14	8324.99	8395.23	8692.21
2018-05-22T00:00:00.000000000	8395.23	8405.18	7959.49	7983.52	7785.34
2018-05-23T00:00:00.000000000	7983.52	8023.73	7442.97	7502.56	8209.6
2018-05-24T00:00:00.000000000	7502.56	7721.83	7272.64	7578.69	8186.96
2018-05-25T00:00:00.000000000	7578.69	7649.95	7344.9	7460.69	7101.07
2018-05-26T00:00:00.000000000	7460.69	7611.56	7298.57	7334.16	7472.4
2018-05-27T00:00:00.000000000	7334.16	7387.12	7226.65	7344.97	7852.42
2018-05-28T00:00:00.000000000	7344.97	7431.44	7080.28	7105.67	8224.97
2018-05-29T00:00:00.000000000	7105.67	7519.2	7054.14	7460.58	8006.27
2018-05-30T00:00:00.000000000	7460.58	7551.88	7286.17	7375.67	7423.94
2018-05-31T00:00:00.000000000	7375.67	7598.7	7336.63	7487.19	7747.98
2018-06-01T00:00:00.000000000	7487.19	7599.84	7354.69	7518.24	6933.29
2018-06-02T00:00:00.000000000	7518.24	7683.2	7453.13	7636.19	6422.36

Рис.3.3. Частковий вид реальних тестових даних – колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ та прогнозних даних "MLP Forecast" для часового ряду "Open".

Для перевірки розподілу імовірностей значень часового ряду відхилень "residuals" на відповідність розподілу імовірностей нормальному закону розподілу

використовувався статистичний тест Колмогорова (статистичний тест Колмогорова-Смірнова). Даний статистичний тест Колмогорова із рівнем значимості 0.05, показав, що виконується гіпотеза H_0 . Таким чином, розподіл імовірностей значень часового ряду відхилень "residuals" не значимо відрізняється від розподілу імовірностей нормального закону білого шуму.

Нище описано практичну реалізацію в кодї програмного агента нейромережевої архітектури багат шаровий перцептрон для прогнозування часового ряду "Open" курсу КБ при відкритті торгівлі на біржі з колекції часових рядів: "Open", "High", "Low", "Close".

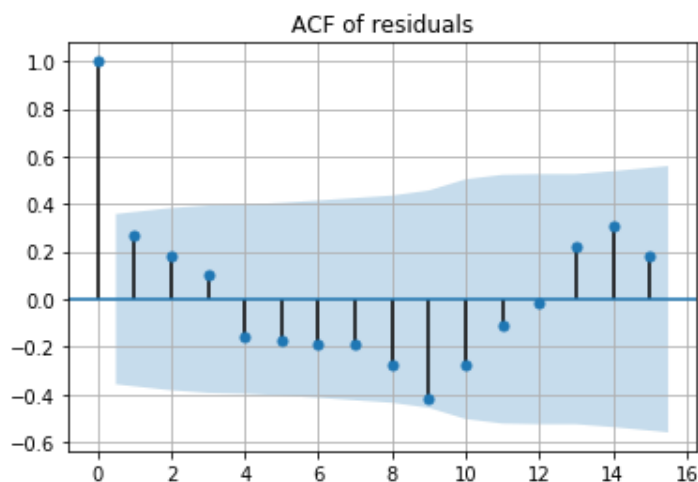


Рис.3.4. Автокореляційна функція "ACF of residuals" часового ряду відхилень "residuals" для "MLP Forecast".

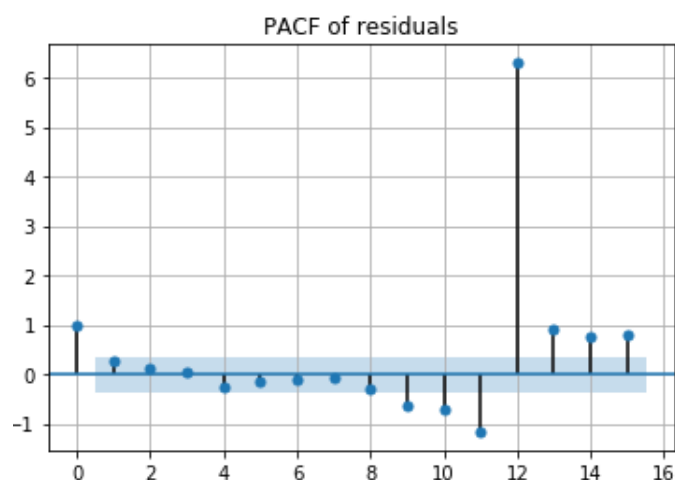


Рис.3.5. Часткова автокореляційна функція "PACF of residuals" часового ряду відхилень "residuals" для "MLP Forecast".

З дистрибутиву анаконди для створення скрипта імпортуються необхідні бібліотеки та фреймворки:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.neural_network import MLPRegressor
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.graphics.tsaplots import plot_pacf
from statsmodels.stats.diagnostic import kstest_normal
```

Читання даних з файлу їх попередня обробка та візуалізація:

```
"""Reading data, preprocessing, visualization"""
ts = pd.read_csv('./DATA/TS_OF_BITCOIN_PRICE/TS_OF_BITCOIN_PRICE.csv',
parse_dates=True)

#Dropping very old historical data
ts.drop(ts.index[:2724], inplace=True)

#Converting column 'Data' to Indexes
ts.Timestamp = pd.to_datetime(ts.Date)
ts.index = ts.Timestamp
ts = ts.resample('D').mean()

#Visualization of bitcoin price
ts.plot()
plt.grid()
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Price of bitcoin')
plt.show()
```

Створення навчальної вибірки та тестової вибірки:

```
"""Creating training time series and test time series"""
#Target
t = "Open"
#Horizon of test (days)
h = 30

#Creating training time series
ts_train = ts[:-h]
#Creating test time series
ts_test = ts[-h:]
```

```

#Creating training features
x_train = ts_train[:-h]
x_train = [ts_train.iloc[i:i+h].values.transpose() for i in
range(0,len(ts_train.index)-2*h+1)]

array_train = []
for i in x_train:
    array = i.flatten()
    array_train.append(array)

x_train = np.vstack(array_train)

#Creating test features
x_test = ts_train[-h:]
x_test = x_test.values.flatten()
x_test = x_test.reshape((x_test.shape[0],1)).transpose()

#Creating training target
y_train = ts_train[[t]].iloc[h:]
y_train = [y_train.iloc[i:i+h].values.transpose() for i in
range(0,len(y_train.index)-h+1)]
y_train = np.vstack(y_train)

#Creating test target
y_test = ts_test[[t]].values

#Plotting Training time series and Test time series
plt.figure(figsize=(8,3))
plt.plot(ts_train, label='Training time series')
plt.plot(ts_test, label='Test time series')
plt.legend()
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Price of bitcoin')
plt.grid()
plt.show()

```

Програмування багатосарового перцептрона та прогнозування часового ряду на основі MLP:

```

"""Time series forecast based on MLP model"""
#Initialization MLP model
estimator = MLPRegressor()
#Fitting
estimator.fit(x_train, y_train)
#Prediction
predict = estimator.predict(x_test)

```

Візуалізація колекції часових рядів та прогнозного часового ряду:

```

"""Visualization of forecast"""
#Copy
forecast = ts_test.copy()

#Preparing
forecast['Forecast'] = predict.transpose()
y_pred = forecast.Forecast.values

#Reshape
y_pred = y_pred.reshape((y_pred.shape[0]),1)

#Plotting Training time series, Test time series and MLP Forecast
plt.figure(figsize=(8,3))
plt.plot(ts_train, label='Training time series')
plt.plot(ts_test, label='Test time series')
plt.plot(forecast['Forecast'], label='MLP Forecast')
plt.legend(loc='best')
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Training time series, Test time series, MLP Forecast')
plt.grid()
plt.show()

```

Обчислення похибки:

```

"""Calculation of MAPE, % """
mape = 100 * np.mean(np.abs((y_pred - y_test) / y_test))

```

Аналіз часового ряду відхилень "residuals":

```

"""Residuals analysis"""
#Calculation of residuals
res = y_pred - y_test

#Plot the autocorrelation function of 'residuals'
plot_acf(res, alpha=0.05)
plt.title('ACF of residuals')
plt.grid()
plt.show()

#Plot the partial autocorrelation function of 'residuals'
plot_pacf(res, alpha=0.05)
plt.title('PACF of residuals')
plt.grid()
plt.show()

```

```

#Calculation of Kolmogorov's test for 'residuals'
res_test = kstest_normal(res)
#Calculation of p-value of Kolmogorov statistic for 'residuals'
res_p_value = res_test[1]

#Calculation of p-value of Kolmogorov statistic for 'residuals'
res_p_value = res_test[1]

```

3.2. Прогнозування курсу криптовалюти біткойн програмним агентом на основі нейромережевої архітектури згорткової нейромережі

Другим програмним агентом МПС прогнозування курсу КБ є згорткова нейромережа. Використовувалася згорткова нейромережа, яка в кодї будувалася на основі програмного об'єкта "Sequential". Таким чином, практична реалізація згорткової нейромережі в кодї має вид програмного конвеєра. Даний програмний конвеєр містить наступні шари: 1-ий шар Conv1D – це одновимірний згортковий шар із гіперпараметрами – 64 фільтри, розмір ядра (8,8), функція активації 'relu'; 2-ий шар Maxpooling1D – це вибір максимального значення з пула, який має гіперпараметр розмір пула рівний 2; 3-ий шар Flatten; 4-ий шар Dense – це повнозв'язний шар нейронів кількістю 50 та функцією активації 'relu'. Компілювалася згорткова нейромережа з гіперпараметрами: мінімізація помилки – це середньоквадратичне відхилення, методом навчання був обраний 'adam', розмір пакетів даних був у режимі замовчування; решта гіперпараметри використовували значення за замовчуванням.

Аналогічно, як і у випадку нейромережі багатошаровий перцептрон – першого програмного агента МПС прогнозування курсу КБ, навчальна та тестові вибірки формувалися наступним чином. Перші 2724 днів у колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ представляли собою навчальну вибірку, а останні 30 днів представляли тестову вибірку.

Таким чином, аналогічно до вище названої процедур машинного навчання, програмний агент згорткова нейромережевої навчалася на навчальній вибірці об'ємом у 2724 дні та тестувалася на тестовій вибірці об'ємом у 30 днів.

Як приклад прогнозування у часі був обраний часовий ряд курсу КБ при відкритті торгівлі на біржі "Open" з колекції часових рядів: "Open", "High", "Low", "Close".

На рис.3.6 різними кольорами візуалізовано навчальну та тестову вибірки у колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ для програмного агента згорткової нейромережі. Також на даній візуалізації наведено прогнозний графік "CNN Forecast" курсу КБ при відкритті торгівлі на біржі.

Для більшої наочності сприйняття прогнозних у часі значень курсу КБ згортковою нейромережею нище наведено масштабовану графічну візуалізацію (рис.3.7).



Рис.3.6. Прогноз у часі "CNN Forecast" курсу КБ на основі програмного агента згорткової нейромережі для часового ряду "Open" з колекції часових рядів: "Open", "High", "Low", "Close".

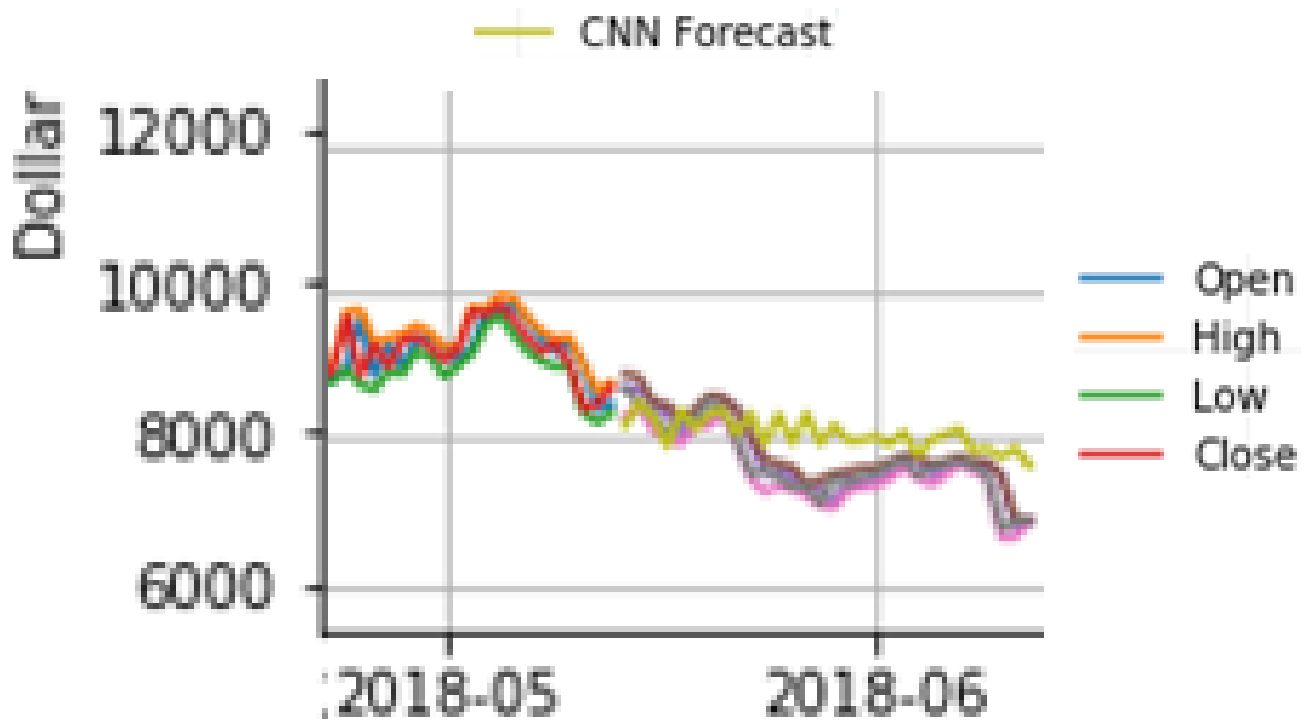


Рис.3.7. Більший масштаб прогнозних значень "CNN Forecast" курсу КБ згортковою нейромережею для часового ряду "Open" з тестової колекції часових рядів: "Open", "High", "Low", "Close".

Окрім цього, розглядаються відповідні дані у вигляді числового масиву, який візуалізовано наведений на рис.3.8.

Слід відзначити, що повно і детально проводилася процедура оцінки точності прогнозу у часі курсу КБ.

Для цього був одержаний часовий ряд відхилень "residuals" між реальним часовим рядом "Open" – реальними даними та прогнозним часовим рядом "CNN Forecast" – прогнозними даними.

Середня відносна похибка прогнозу, яка обчислена на основі часового ряду відхилень "residuals", склала $CNN_MAPE=4.78\%$.

Відзначимо, що відомо – часовий ряд відхилень "residuals" повинен бути максимально подібним до нормального білого шуму. Тому оцінювалися його автокореляційна функція "ACF of residuals" та часткова автокореляційна функція "PACF of residuals", що наведені відповідно на рис.3.9 та рис.3.10.

Date	Open	High	Low	Close	Forecast
2018-05-14T00:00:00.000000000	8688.03	8878.94	8296.14	8675.21	8175.6553
2018-05-15T00:00:00.000000000	8675.21	8835.51	8433.43	8474.24	8492.49
2018-05-16T00:00:00.000000000	8474.24	8493.31	8110.33	8340.3	8290.51
2018-05-17T00:00:00.000000000	8340.3	8458.53	7999.59	8058.6	7895.576
2018-05-18T00:00:00.000000000	8058.6	8273.54	7931.43	8240.72	8388.984
2018-05-19T00:00:00.000000000	8240.72	8392.77	8154.15	8234.15	8125.4224
2018-05-20T00:00:00.000000000	8234.15	8582.8	8172.18	8520.81	8329.067
2018-05-21T00:00:00.000000000	8520.81	8566.14	8324.99	8395.23	8398.535
2018-05-22T00:00:00.000000000	8395.23	8405.18	7959.49	7983.52	8058.647
2018-05-23T00:00:00.000000000	7983.52	8023.73	7442.97	7502.56	8347.347
2018-05-24T00:00:00.000000000	7502.56	7721.83	7272.64	7578.69	7928.3325
2018-05-25T00:00:00.000000000	7578.69	7649.95	7344.9	7460.69	8326.4
2018-05-26T00:00:00.000000000	7460.69	7611.56	7298.57	7334.16	7944.7183
2018-05-27T00:00:00.000000000	7334.16	7387.12	7226.65	7344.97	8339.349
2018-05-28T00:00:00.000000000	7344.97	7431.44	7080.28	7105.67	7929.13
2018-05-29T00:00:00.000000000	7105.67	7519.2	7054.14	7460.58	8179.581
2018-05-30T00:00:00.000000000	7460.58	7551.88	7286.17	7375.67	7970.117
2018-05-31T00:00:00.000000000	7375.67	7598.7	7336.63	7487.19	7980.2383
2018-06-01T00:00:00.000000000	7487.19	7599.84	7354.69	7518.24	8024.551
2018-06-02T00:00:00.000000000	7518.24	7683.2	7453.13	7636.19	7924.167

Рис.3.8. Частковий вид реальних тестових даних – колекції часових рядів: "Open", "High", "Low", "Close" курсу КБ та прогнозних даних "CNN Forecast" для часового ряду "Open".

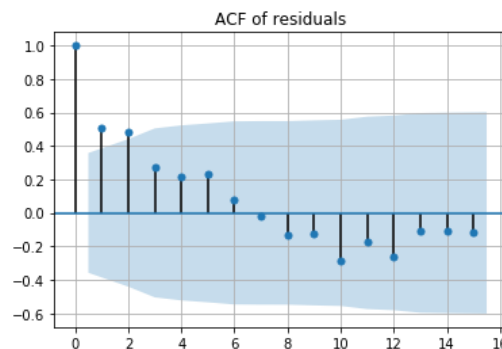


Рис.3.9. Автокореляційна функція "ACF of residuals" часового ряду відхилень "residuals" для прогнозу "CNN Forecast".

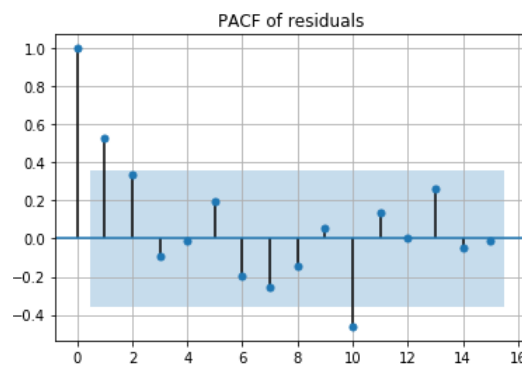


Рис.3.10. Часткова автокореляційна функція "PACF of residuals" часового ряду відхилень "residuals" часового ряду відхилень "residuals" для "CNN Forecast".

Для перевірки розподілу імовірностей значень часового ряду відхилень "residuals" для "CNN Forecast" на відповідність розподілу імовірностей нормальному закону розподілу використовувався статистичний тест Колмогорова (статистичний тест Колмогорова-Смірнова). Даний статистичний тест Колмогорова із рівнем значимості 0.05, показав, що виконується гіпотеза H_0 . Таким чином, розподіл імовірностей значень часового ряду відхилень "residuals" для "CNN Forecast" не значимо відрізняється від розподілу імовірностей нормального закону білого шуму.

Нище описано практичну реалізацію в коді програмного агента нейромережевої архітектури на основі згорткової нейромережі для прогнозування часового ряду "Open" курсу КБ при відкритті торгівлі на біржі з колекції часових рядів: "Open", "High", "Low", "Close".

З дистрибутиву анаконди для створення скрипта імпортуються необхідні бібліотеки та фреймворки:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Flatten
from keras.layers import Conv1D
from keras.layers import MaxPooling1D
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.graphics.tsaplots import plot_pacf
from statsmodels.stats.diagnostic import kstest_normal
```

Читання даних, попередня їх обробка та візуалізація

```
"""Reading data, preprocessing, visualization"""
ts = pd.read_csv('./DATA/TS_OF_BITCOIN_PRICE/TS_OF_BITCOIN_PRICE.csv',
parse_dates=True)

#Dropping very old historical data
ts.drop(ts.index[:2724], inplace=True)
```



```

#Converting column 'Data' to Indexes
ts.Timestamp = pd.to_datetime(ts.Date)
ts.index = ts.Timestamp
ts = ts.resample('D').mean()

#Visualization of bitcoin price
ts.plot()
plt.grid()
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Price of bitcoin')
plt.show()

```

Створення навчальної вибірки та тестової вибірки для "CNN Forecast":

```

"""Creating training time series and test time series"""
#Target
t = "Open"
#Horizon of test (days)
h = 30

#Creating training time series
ts_train = ts[:-h]
#Creating test time series
ts_test = ts[-h:]

#Creating training features
x_train = ts_train[:-h]
x_train = [ts_train.iloc[i:i+h].values.transpose() for i in
range(0,len(ts_train.index)-2*h+1)]

array_train = []
for i in x_train:
    array = i.flatten()
    array_train.append(array)

x_train = np.vstack(array_train)

#Reshape
x_train = x_train.reshape((x_train.shape[0], x_train.shape[1], 1))

#Creating test features
x_test = ts_train[-h:]
x_test = x_test.values.flatten()

#Reshape
x_test = x_test.reshape((1, x_test.shape[0], 1))

#Creating training target
y_train = ts_train[[t]].iloc[h:]
y_train = [y_train.iloc[i:i+h].values.transpose() for i in
range(0,len(y_train.index)-h+1)]
y_train = np.vstack(y_train)

#Creating test target
y_test = ts_test[[t]].values

```

```

#Plotting Training time series and Test time series
plt.figure(figsize=(8,3))
plt.plot(ts_train, label='Training time series')
plt.plot(ts_test, label='Test time series')
plt.legend()
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Price of bitcoin')
plt.grid()
plt.show()

```

Програмування згорткової нейромережі та прогнозування часового ряду на основі CNN:

```

"""Time series forecast based on CNN model"""
#Creating CNN model
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=8,
activation='relu', input_shape=(4*h,1)))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten())
model.add(Dense(50, activation='relu'))
model.add(Dense(h))
model.compile(optimizer='adam', loss='mse')
#Fitting
model.fit(x_train, y_train, epochs=100, verbose=0)
#Prediction
predict = model.predict(x_test, verbose=0)

```

Візуалізація колекції часових рядів та прогнозованого часового ряду:

```

"""Visualization of forecast"""
#Copy
forecast = ts_test.copy()

#Preparing
forecast['Forecast'] = predict.transpose()
y_pred = forecast.Forecast.values

#Reshape
y_pred = y_pred.reshape((y_pred.shape[0]),1)

```

```
#Plotting Training time series, Test time series and CNN Forecast
plt.figure(figsize=(8,3))
plt.plot(ts_train, label='Training time series')
plt.plot(ts_test, label='Test time series')
plt.plot(forecast['Forecast'], label='CNN Forecast')
plt.legend(loc='best')
plt.xlabel('Time (days)')
plt.ylabel('Dollar')
plt.title('Training time series, Test time series, CNN Forecast')
plt.grid()
plt.show()
```

Обчислення похибки:

```
"""Calculation of MAPE, % """
```

```
mape = 100 * np.mean(np.abs((y_pred - y_test) / y_test))
```

Аналіз часового ряду відхилень "residuals":

```
"""Residuals analysis"""
```

```
#Calculation of residuals
```

```
res = y_pred - y_test
```

```
#Plot the autocorrelation function of 'residuals'
```

```
plot_acf(res, alpha=0.05)
```

```
plt.title('ACF of residuals')
```

```
plt.grid()
```

```
plt.show()
```

```
#Plot the partial autocorrelation function of 'residuals'
```

```
plot_pacf(res, alpha=0.05)
```

```
plt.title('PACF of residuals')
```

```
plt.grid()
```

```
plt.show()
```

```
#Calculation of Kolmogorov's test for 'residuals'
```

```
res_test = kstest_normal(res)
```

```
#Calculation of p-value of Kolmogorov statistic for 'residuals'
```

```
res_p_value = res_test[1]
```

3.3. Консолідація нейромережових програмних агентів у МПС прогнозування курсу КБ

Вище було наведено практичну реалізацію двох нейромережевих програмних агентів для прогнозування у часі курсу КБ. Для кожного з цих нейромережевих програмних агентів оцінено похибку – середнє значення відносної похибки прогнозу у часі курсу КБ. Для першого нейромережевого програмного агента у вигляді багат шарового перцептрона середнє значення відносної похибки склало $MLP_MAPE=5.85\%$. А для другого нейромережевого програмного агента у вигляді згорткової нейромережі середнє значення відносної похибки склало $CNN_MAPE=4.78\%$.

Таким чином, постає питання щодо об'єднання роботи обох нейромережевих програмних агентів в єдину МПС прогнозування у часі курсу КБ.

Загалом, є три стратегії консолідації даних двох нейромережевих програмних агентів у МПС. Перша, практично реалізувати процедуру простого голосування – результуючий прогноз у часі курсу КБ обчислити, як середнє значення прогнозів кожного з нейромережевих програмних агентів. Друга, практично реалізувати процедуру зваженого голосування – результуючий прогноз у часі курсу КБ обчислити, як середнє зважене значення прогнозів кожного з нейромережевих програмних агентів. Третя, не згортати два прогнози нейромережевих програмних агентів в один інтегральний прогноз МПС, а залишити їх окремо.

Заради справедливості, слід зазначити, що кожна із стратегій має, як переваги так і недоліки. Перша – враховує консолідоване рішення обох агентів, але втрачається диференціальна прогнозна картина. Друга – враховує консолідоване зважене рішення обох агентів, але також втрачається диференціальна прогнозна картина. Третя – не враховує консолідоване рішення обох агентів, але не втрачається диференціальна прогнозна картина.

У роботі було вибрано третю стратегію, оскільки диференціальна прогнозна картина була визначена, як пріоритет. Тому МПС прогнозування у часі курсу КБ складається з двох прогнозів двох окремих, вище практично реалізованих нейромережевих програмних агентів. Таким є остаточна практична реалізація фінансового інструменту для управління інвестиціями у КБ.

4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

В роботі розробляється мультиагентна програмна система для прогнозування курсу криптоактиву біткойн. Зокрема, вирішуються завдання: проектування, розробка та практична реалізація даної мультиагентної програмної системи. Вирішення цих завдань передбачає використання різного роду обладнання - комп'ютерної техніки, зокрема, персональних комп'ютерів, периферійних пристроїв, тощо. Таким чином, виникає цілий ряд важливих питань пов'язаних з обов'язковим дотримання вимог з охорони праці, зокрема, техніки безпеки, пожежної безпеки на робочому місці, відповідно з діючими нормативно-правовими актами та встановленими нормами, оскільки це лежить в основі збереження життя людей.

Для належної та безпечної роботи людей з проектування, розробки та практичної реалізації програмної системи необхідно організувати безпечні умови праці. При цьому керівник несе безпосередню відповідальність за порушення нормативно-правових актів з охорони праці.

До початку виконання всіх видів робіт серед інженерів розробників програмного забезпечення проведено інструктаж з техніки безпеки на робочому місці відповідно до вимог типового положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України ДБН В.1.1-7-2016 „Пожежна безпека об'єктів будівництва та НПАОП 0.00-7.15-18. Розробка програмного забезпечення та проведення досліджень відбувались відповідно до Закону України "Про охорону праці" та "Правил охорони праці при експлуатації електронно-обчислювальних машин" та Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, а також НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [30].

Організація робочого місця інженера розробника програмного забезпечення відповідає усім ергономічним вимогам щодо робочого місця: ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги». Відстань від екрана до ока фахівців, які працюють за комп'ютером визначається згідно з вимогами ДСанПіН 3.3.2.007-98. Розміщення принтера або іншого пристрою введення-виведення інформації на робочому місці має забезпечувати добру видимість екрана комп'ютера, зручність ручного керування пристроєм введення-виведення інформації в зоні досяжності моторного поля згідно з вимогами ДСанПіН 3.3.2.007-98.

Електробезпека на робочому місці інженера розробника програмного забезпечення повинна відповідати вимогам Правил безпечної експлуатації електроустановок споживачів, затверджених наказом Держнаглядохоронпраці від 09.01.98 N 4 , зареєстрованих у Міністерстві юстиції України 10.02.98 за N 93/2533 (НПАОП 40.1-1.21-98).

Для забезпечення електробезпеки при роботі персональних комп'ютерів та периферійних пристроїв під час проведення досліджень електромережа відповідала наступним правилам: лінія електромережі живлення виконана, як окрема групова трипровідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників; нульовий захисний провідник використовується для заземлення електроприймачів; електричне устаткування, електропроводи та кабелі мають апаратуру захисту від струму короткого замикання та інших аварійних режимів; під час монтажу та експлуатації електромережі необхідно повністю унеможливити виникнення електричного джерела загоряння внаслідок короткого замикання та перевантаження проводів, обмежено застосування проводів з легкозаймистою ізоляцією та застосовано негорючу ізоляцію; електричне устаткування підключено до мережі лише за допомогою справних штепсельних з'єднань і розеток заводського виготовлення; у штепсельних з'єднаннях та електророзетках є спеціальні контакти для підключення нульового захисного провідника.

Також враховано умови та норми вимог до приміщення, де розміщені робочі місця інженерів розробників програмної системи з точки зору пожежної безпеки: державні будівельні норми ДСТУ Б.В.1.1-36:2016 „Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою”; ДБН В.1.1-7-2016 „Пожежна безпека об’єктів будівництва”. Приміщення з робочими місцями інженерів розробників програмного забезпечення облаштоване автоматичними установками пожежогасіння та пожежної сигналізації, що відповідає державним будівельним нормам "Інженерне обладнання будинків і споруд. Пожежна автоматика будинків і споруд", затверджених наказом Держбуду України від 28.10.98 N 247. Окрім цього, приміщення, де розміщені робочі місця інженерів розробників програмного забезпечення, має бути оснащено вогнегасниками, кількість яких визначається згідно з вимогами ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників». Загальні технічні вимоги і з урахуванням граничнодопустимих концентрацій вогнегасної рідини відповідно до вимог НАПБ А.01.001-2014. Проходи до засобів пожежогасіння мають бути вільними.

Приміщення для роботи повинно мати природне та штучне освітлення відповідно до ДБН В.2.5-28-2018. Природне освітлення забезпечується через світлові прорізи, коефіцієнт природної освітленості (КПО) не нижче ніж 1,5%. Розраховується КПО за методикою викладеною в ДБН В.2.5-28-2018. Штучне освітлення в приміщенні з робочими місцями для інженерів розробників програмного забезпечення забезпечувалось системою загального рівномірного освітлення. В якості джерела світла для штучного освітлення використані люмінесцентні лампи типу ЛБ.

Таким чином, в результаті аналізу вимог щодо охорони праці визначено особливості організації робочих місць інженерів розробників програмного забезпечення, а саме, ергономічні вимоги, вимоги з електробезпеки, вимоги пожежної безпеки, вимоги щодо природного та штучного освітлення для ефективної і безпечної роботи.

4.2. Прогнозування можливої обстановки під час надзвичайних ситуацій мирного та воєнного часу

У роботі розглянуто процеси проектування, розробки та практичної реалізації мультиагентної програмної системи для прогнозування у часі курсу криптоактиву біткойн. Дана система в основі своєї роботи використовує програмні агенти на основі нейромережових технологій. Оскільки, нейромережові технології є універсальними обчислювальними технологіями, тобто можуть бути використаними для вирішення довільних алгоритмічно розв'язуваних задач із довільної сфери людської діяльності, то доцільно розглянути їх застосування для прогнозування у часі різного роду динамічних процесів із сфери безпеки в надзвичайних ситуаціях в мирний та військовий час. В даному контексті конструктивно буде зазначити використання методичного посібника з курсу: "Безпека в надзвичайних ситуаціях" [31].

Отже, нейромережові програмні агенти, які використовуються у даній мультиагентній системі можуть бути використаними для прогнозування у часі поширення різного роду вірусів у спільноті людей, тобто прогнозування зростання кількості хворих на той чи інший вірус, а значить апріорі готувати визначену кількість ліжок у лікарнях для госпіталізації майбутніх вірусоінфікованих хворих. На основі таких прогнозів у часі можна заздалегіть підготувати лікарні до прийому необхідної кількості інфікованих людей і цим самим уникнути надзвичайної ситуації – колапсу медичної системи, як на регіональному рівні так і на державному рівні. В контексті цього слушно буде зазначити, що автор даної магістерської роботи проводив розробку та тестування нейромережових програмних агентів при прогнозування у часі кількості хворих на коронавірус в регіоні Тернопільської області на основі динамічних у часі даних, які були в Інтернет просторі. Оновлення прогнозування у часі кількості хворих на коронавірус відбувалося щоденно з горизонтів прогнозування на один тиждень вперед. Середня

похибка прогнозування у часі кількості інфікованих на коронавірус у Тернопільській області склала 3-4 %. Цінність такого роду прогнозних у часі даних уможливорює уникнення колапсу медичної системи у надзвичайній ситуації експоненційного зростання кількості інфікованих людей на коронавірус.

Також можна застосовувати такого роду нейромережеві програмні агенти для прогнозування у часі різного роду процесів надзвичайних ситуацій у військовий час. Наприклад, що стосується прогнозування у часі імовірності застосування неконвенційної зброї, наприклад, застосування зброї масового ураження, наприклад, ядерної зброї, яка приведе до виникнення надзвичайної ситуації у вигляді радіоактивного забруднення, різного роду масових пошкоджень на великій території за короткий проміжок часу. В контексті вище названого доречно розглянути методичний посібник з курсу: "Безпека в надзвичайних ситуаціях" [31].

На сьогодні у військовій науці та безпеці надзвичайних ситуацій прогнозування імовірності виникнення у часі надзвичайної ситуації, як наслідку застосування неконвенційної зброї у вигляді ядерного вибуху з плином часу стає надзвичайно актуальним.

Прогнозування у часі імовірнісної міри виникнення такого виду надзвичайної ситуації на основі нейромережевих програмних агентів є цілком реальною задачею, оскільки такого роду нейромережеві агенти володіють високою точністю прогнозування у часі різного роду процесів і дають стабільно точні результати на різних часових горизонтах, тобто короткостроковому часовому горизонті, середньостроковому часовому горизонті та довгостроковому часовому горизонті. Така висока точність досягається завдяки технології глибокого навчання та того факту, що такі нейронні мережі роблять прогнозування на основі інформації різної модальності, яку вони збирають з найрізноманітніших джерел цифрового світу, наприклад, Інтернету, зокрема, різних соціальних мереж, месенджерів, веб-порталів, розвідок, тощо. Під різними модальностями розуміються всі види інформації, тобто, текстова, візуальна, числова, тощо.

Таку задачу під час військового стану – прогнозування у часі виникнення надзвичайної ситуації, як наслідку застосування неконвенційної зброї, наприклад,

ядерної бомби, можна алгоритмічно формалізувати. Розгляньмо алгоритмічну формалізацію задачі прогнозування у часі імовірнісного значення виникнення надзвичайної ситуації, як наслідку ядерного удару. Отже, триває у часі війна, яку веде дві або більше сторін з використанням конвенційної зброї. У часі розгортаються перемінні успіхи кожної із сторін, тобто, процес розгортається у часі із певною сезонною компонентою, також у часі може бути трендова компонента успішності однієї із сторін і не успішності іншої із сторін, причому різної форми, лінійної, або логарифмічної, або експоненційної. Також розгортання такого роду військового процесу супроводжується шумовими ефектами типу локальними боями, які приводять до перемінних успіхів кожної із сторін на локальному проміжку фронту. Загальний часовий проце можна розглянути як адитивну, або як мультиплікативну суміші вище названих процесів. Тривалий у часі процес ведення бойових дій виснажує ресурси не лише техніко-військові, людські, але й головне психічні. Виснаження психіки кожного з головнокомандувачів ворогуючих сторін приводить до збільшення значення імовірності застосування ними ядерного удару, який у свою чергу викличе надзвичайну ситуацію у вигляді масових пошкоджень, радіаційного забруднення, тощо. Використовуючи останні знання з нейробіології щодо існування дзеркальних нейронів маси в соціальних мережах, месенджерах в стані відзеркалювати психічний стан своїх політичних лідерів, які є головнокомандувачами військ кожної із сторін і у великій статистиці постів переписок величезної кількості людей нейронні мережі, які лежать в сучасних системах штучного інтелекту, наприклад ChatGPT, та технології глибокого навчання можуть з високою точністю спрогнозувати значення імовірнісної міри виникнення психічного стану у тої чи іншої сторони щодо нанесення ядерного удару і як наслідок виникнення надзвичайного стану.

Завершуючи даний підрозділ роботи і підсумовуючи вище сказане слушно буде навести доцільність використання методичного посібника з дисципліни: "Безпека в надзвичайних ситуаціях" [31].

ВИСНОВКИ

За результатами виконання даної кваліфікаційної роботи магістра одержано наступні висновки:

1. Запропоновано нові методи та засоби математичного та програмного забезпечення, які дали можливість спроектувати МПС прогнозування курсу КБ на основі технологій нейронних мереж – фінансовий інструмент для керування інвестиціями в КБ.

2. На основі нових методів та засобів математичного та програмного забезпечення розроблено та практично реалізованої МПС прогнозування курсу КБ на основі технологій нейронних мереж – фінансовий інструмент для керування інвестиціями в КБ, що дає змогу на практиці зменшити ризики управління інвестиціями на ринку криптоактивів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Фінансовий інжиніринг. URL: https://pidru4niki.com/11050519/bankivska_sprava/finansoviy_inzhiniring (дата звернення: 09.11.2023)
2. Фінансовий інжиніринг: механізм здійснення. URL: <https://economics.net.ua/files/archive/2018/No6/25.pdf> (дата звернення: 09.11.2023)
3. Фінансовий інжиніринг як драйвер світової фінансової системи. URL: http://irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/efek_2014_5_81.pdf (дата звернення: 09.11.2023)
4. Фінансовий інжиніринг – генератор фінансових інновацій. URL: http://www.market-infr.od.ua/journals/2020/39_2020_ukr/59.pdf (дата звернення: 09.11.2023)
5. Фінансовий інжиніринг. іновації та загрози. URL: <https://core.ac.uk/download/pdf/84006963.pdf> (дата звернення: 09.11.2023)
6. Фінансовий інжиніринг на ринку корпоративних боргових цінних паперів України. URL: http://bulletin-econom.univ.kiev.ua/wp-content/uploads/2016/01/116_11.pdf (дата звернення: 10.11.2023)
7. Інноваційні інструменти фінансового ринку. URL: http://dspace.s.msu.edu.ua:8080/bitstream/123456789/3884/1/Innovative_financial_market_instruments.pdf (дата звернення: 10.11.2023)
8. Фінансова статистика. URL: <https://drive.google.com/file/d/15O68vOvNyuId15cmOBuKdnPayWNYH4NZ/view> (дата звернення: 10.11.2023)
9. Фінансова статистика. URL: <https://core.ac.uk/download/pdf/48403329.pdf> (дата звернення: 10.11.2023)

10. Python для обробки даних і моделювання фінансово-економічних показників. URL: <https://itce.vntu.edu.ua/index.php/itce/article/view/823/544> (дата звернення: 10.11.2023)

11. Комп'ютерне моделювання: аналіз, управління, оптимізація. URL: [https://kmauo.org/wp-content/uploads/2020/2/%E2%84%962\(8\)2020.pdf](https://kmauo.org/wp-content/uploads/2020/2/%E2%84%962(8)2020.pdf) (дата звернення: 10.11.2023)

12. Тренди ШІ: як штучний інтелект вплине на фінансову сферу. URL: <https://speka.media/trendi-si-yak-stucnii-intelekt-vpline-na-finansovu-sferu-vmjkzv> (дата звернення: 10.11.2023)

13. Магія штучного інтелекту: аналітика, фінанси та стандартизація. URL: <https://business.diia.gov.ua/cases/tehnologii/magia-stucnogo-intelektu-analitika-finansi-ta-standartizacia-abo-prosto-dodaj-si> (дата звернення: 10.11.2023)

14. Штучний інтелект у сфері фінансів. URL: https://elartu.tntu.edu.ua/bitstream/lib/31485/2/FMZKP_2020_Lutsko_Z-Artificial_intellectual_135-137.pdf (дата звернення: 10.11.2023)

15. Штучний інтелект у FinTech: 5 програм, які використовують компанії. URL: <https://blog.easypay.ua/shtuchniy-intelekt-u-fintech-5-program-yaki-vikoristovuyut-kompaniyi/> (дата звернення: 10.11.2023)

16. ШІ у ролі фінансового консультанта – один із провідних трендів фінтеху. URL: <https://www.ukrinform.ua/rubric-technology/3684961-si-u-rol-i-finansovogo-konsultanta-odin-iz-providnih-trendiv-fintehu.html> (дата звернення: 10.11.2023)

17. Як штучний інтелект змінює світ фінансів. URL: <https://ua.news.ua/technologies/kak-yskusstvennyj-yntellekt-menyaet-myr-fynansov> (дата звернення: 10.11.2023)

18. Психологія на службі у фінансів: які звички прирікають нас на невдачі з грошима. URL: <https://psm7.com/uk/analytics/psixologiya-na-sluzhbe-u-finansov-kakie-privychki-obrekaют-nas-na-neudachi-s-dengami.html> (дата звернення: 10.11.2023)

19. Психологія фінансів як основа сучасної фінансової науки. URL: <https://lib.udau.edu.ua:8443/server/api/core/bitstreams/6f78ed43-d97d-43c0-bd80-fbc8f823c013/content> (дата звернення: 10.11.2023)

20. Machine Learning And Neural Networks In The Fintech Industry. URL: <https://techround.co.uk/finance/machine-learning-neural-networks-fintech-industry/> (дата звернення: 11.11.2023)

21. Neural Network Technologies of Investment Risk Estimation Taking into Account the Legislative Aspect. URL: <https://ceur-ws.org/Vol-2805/paper23.pdf> (дата звернення: 11.11.2023)

22. Blockchain financial investment based on deep learning network algorithm. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0377042720300145> (дата звернення: 11.11.2023)

23. An Analysis of the Applications of Neural Networks in Finance. URL: https://web.archive.org/web/20170809115741id_/http://cns-classes.bu.edu/cn710/Fall2007/file_repository/112.pdf (дата звернення: 11.11.2023)

24. Neural Network Applications in Finance. URL: https://www.researchgate.net/publication/282829832_Neural_Network_Applications_in_Finance (дата звернення: 11.11.2023)

25. Neural network technologies in the financial and credit sector. URL: <https://dl.acm.org/doi/fullHtml/10.1145/3487757.3490865> (дата звернення: 11.11.2023)

26. Evaluation of Neural Network Technology in the Financial Sector. URL: <https://ibima.org/accepted-paper/evaluation-of-neural-network-technology-in-the-financial-sector/> (дата звернення: 11.11.2023)

27. How to use Neural Networks in Trading. URL: <https://www.eliftech.com/insights/how-to-use-neural-networks-in-trading/> (дата звернення: 11.11.2023)

28. Financial Asset Management Using Artificial Neural Networks. URL: <https://www.igi-global.com/pdf.aspx?tid=258572&ptid=229667&ctid=4&oa=true&isxn=9781799806554> (дата звернення: 11.11.2023)

29. How Artificial Intelligence Predicts Trading Market. URL: <https://intellias.com/artificial-intelligence-predicts-financial-markets/> (дата звернення: 11.11.2023)

30. Атаманчук П.С. Безпека життєдіяльності та охорона праці (Практичний курс): Навчальний посібник. Кам'янець-Подільський: "Думка". 2010. 152 с.

31. Стручок В.С. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання / В.С.Стручок. - Тернопіль: ФОП Паляниця В. А., 2022. - 156 с. Available at: <http://elartu.tntu.edu.ua/handle/lib/39196> (accessed 18.12.2023)

ДОДАТКИ

ДОДАТОК А
Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

МАТЕРІАЛИ

ХІ НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ
«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»



13-14 грудня 2023 року

ТЕРНОПІЛЬ
2023

Д.Р. Карабан; Р.О. Жаровський МЕТОДИ ЗАБЕЗПЕЧЕННЯ АНОНІМНОСТІ В ІНТЕРНЕТІ D.R. Karaban; R.O. Zharovskyi METHODS OF PROVIDING ANONYMITY IN THE INTERNET	237
Корнєв О., Пастух О. РОЗРОБКА МУЛЬТИАГЕНТНОЇ ПРОГРАМНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ КУРСУ КРИПТОВАЛЮТ НА ОСНОВІ НЕЙРОМЕРЕЖЕВИХ ТЕХНОЛОГІЙ Korniev O., Pastukh O. DEVELOPMENT OF A MULTI-AGENT SOFTWARE SYSTEM FOR FORECASTING THE CRYPTOCURRENCY COURSE BASED ON NEURAL NETWORK TECHNOLOGIES	238
Р. Войтович, М. Петрик ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ ВИРІШЕННЯ ЗАДАЧ КЛАСИФІКАЦІЇ БІОБ'ЄКТІВ НА ЗОБРАЖЕННЯХ R. Voytovych, M. Petryk APPLICATION OF NEURAL NETWORKS TO SOLVE THE PROBLEMS OF CLASSIFICATION OF BIOOBJECTS IN IMAGES	239
О.Р. Оробчук, Р. В. Гарматій АКТУАЛЬНІСТЬ ІНФОРМАЦІЙНИХ СИСТЕМ ДЛЯ АВТОМАТИЗОВАНОГО ВИЯВЛЕННЯ ВРАЗЛИВОСТЕЙ У ВЕБДОДАТКАХ O.R. Orobchuk, R. V. Harmatii INFORMATION SYSTEM FOR AUTOMATED VULNORABILITY DETECTION IN WEB APPLICATION	240
Олег Пастух, Назар Гушпіт ГЛОБАЛЬНА СИСТЕМА ВЗАЄМОПОВ'ЯЗАНИХ КОМП'ЮТЕРНИХ МЕРЕЖ INTERNET COMPUTER Oleh Pastukh, Nazarii Hushpit GLOBAL SYSTEM OF INTERCONNECTED COMPUTER NETWORKS INTERNET COMPUTER	241

УДК 004.8

Корнєв О., Пастух О.

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

**РОЗРОБКА МУЛЬТИАГЕНТНОЇ ПРОГРАМНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ
КУРСУ КРИПТОВАЛЮТ НА ОСНОВІ НЕЙРОМЕРЕЖЕВИХ ТЕХНОЛОГІЙ**

Korniev O., Pastukh O.

**DEVELOPMENT OF A MULTI-AGENT SOFTWARE SYSTEM FOR
FORECASTING THE CRYPTOCURRENCY COURSE BASED ON NEURAL
NETWORK TECHNOLOGIES**

Проведене наукове дослідження спрямоване на розробку інноваційної мультиагентної програмної системи для прогнозування курсу криптовалют, з використанням передових методів нейромережєвих технологій. Використання мультиагентного підходу у системі дозволило ефективно моделювати та аналізувати взаємодію різних агентів на криптовалютних ринках, що є ключовим чинником для точного прогнозування.

Технічний стек проекту включає в себе використання передових нейромережєвих архітектур для аналізу та прогнозу курсів криптовалют. Використання мереж з глибоким навчанням дозволяє системі автоматично адаптуватися до змінних умов ринку та виявляти складні закономірності у динаміці курсів.

Щоб забезпечити надійність та ефективність системи, були використані передові методи тестування програмного забезпечення, такі як JUnit 5. Це дозволило проводити вичерпні тести окремих компонентів системи та забезпечити стабільність та високу якість прогнозів.

Однією з високопріоритетних задач у розробці системи є забезпечення безпеки та конфіденційності, зокрема шляхом використання передових механізмів шифрування та захисту даних користувачів. Для досягнення цих цілей, в системі було використано сучасні алгоритми шифрування з високим ступенем надійності, такі як AES (Advanced Encryption Standard) та RSA (Rivest–Shamir–Adleman). Ці алгоритми гарантують ефективне захоплення та захист конфіденційної інформації, забезпечуючи високий рівень стійкості до різноманітних кібератак і зловживань.

Висновки отримані в результаті дослідження свідчать про великий потенціал застосування нейромережєвих технологій у сфері прогнозування курсу криптовалют. Далі розвиток системи може охоплювати подальше вдосконалення алгоритмів нейромережєвого аналізу, що спрямоване на підвищення точності та швидкодії прогнозів. Паралельно із цим, розширення функціоналу системи може включати у себе нові можливості, такі як аналіз факторів впливу на курс, що дозволить отримати ще більш точні та обґрунтовані прогнози в умовах постійної зміни криптовалютних ринків.

Література

1. Посвятенко Ю. "Інноваційні технології в сучасній розробці програмного забезпечення." / Ю. Посвятенко, М. Іванов, П. Васнецов – К.: Техніка, 2020. – 300 с.
2. AWS Documentation. [Ел. ресурс]. – Режим доступу: <https://docs.aws.amazon.com> (дата звернення: 01.01.2023).
3. What Is The Future Of Artificial Intelligence AI In Transportation? – [Електронний ресурс]: [<https://www.modeshift.com/what-is-the-future-of-artificial-intelligence-ai-in-transportation/>]

ДОДАТОК Б

Диск із кваліфікаційною роботою магістра