

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ

Навчально-науковий інститут інформаційних технологій

та інноваційного підприємництва

(повна назва інституту)

Кафедра технічної кібернетики й інформаційних технологій

ім. професора Р.В. Меркта

(повна назва кафедри)

Пояснювальна записка

до кваліфікаційної роботи

бакалавр

(освітньо-кваліфікаційний рівень)

на тему

Інформаційна система прогнозування фінансових часових рядів

для трейдингу

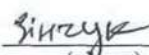
Виконав:

здобувач вищої освіти 4 курсу, групи 2

Спеціальність:

122 «Комп'ютерні науки»

(шифр і назва спеціальності)


(підпис)

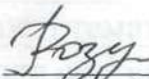
Владислав ЗІНЧУК

(Ім'я, ПРІЗВИЩЕ)

Керівник

к.фіз.-мат.н., доцент

(вчене звання, посада)


(підпис)

Марина РОЗУМ

(Ім'я, ПРІЗВИЩЕ)

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ

Інститут Навчально-науковий інститут інформаційних технологій та інноваційного підприємництва

Кафедра технічної кібернетики й інформаційних технологій ім. професора Р.В. Меркта

Освітньо-кваліфікаційний рівень бакалавр

Напрямок підготовки -

(шифр і назва)

Спеціальність 122 «Комп'ютерні науки»

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

КТКйІТ ім. проф. Р.В. Меркта

Павло НОСО

«18» 06 2026 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Владислав ЗІНЧУК

(Ім'я, ПРІЗВИЩЕ)

1. Тема проекту (роботи) Інформаційна система прогнозування фінансових часових рядів для трейдингу

керівник проекту (роботи) Марина РОЗУМ, к.ф.-м.н., доцент

(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від «18» 05 2026 р. № 124 6.1/89

2. Строк подання проекту (роботи)

Червень 2026

3. Вхідні дані до проекту (роботи) Літературні джерела з аналізу фінансових часових рядів та торговельних бірж

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз основних положень та доцільності тематики трейдингу.

Обґрунтування моделей та проектування системи. Розробка структури проекту.

Розробка програмної реалізації та інтерфейсу модулів системи

Дослідження та аналіз результатів роботи системи

Охорона праці.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

3 діаграми Use case (стор. 48-49),

діаграма загальних компонентів системи (стор. 51),

3 діаграми послідовності дій(стор. 52-54).

6. Консультанти розділів проекту (роботи)

Розділ	Ім'я, ПРІЗВИЩЕ та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці:	доц. Сергій ХОТІН	08.05.26 Хотін	04.06.26 Хотін
Економіко-кадровий	Марина РОЗУМ, к.р.-м.н., доц.	27.04.26 Розум	17.06.26 Розум

7. Дата видачі завдання

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів проекту (роботи)	Примітки
1	Видача завдання	27.04.2026	
2	Переддипломна практика, залік	10.05.2026	
3	Коригування завдання за результатами практики	17.05.2026	
4	Проміжний звіт на кафедрі оцінка готовності	08.06.2026	
5	Попередній захист на кафедрі	15.06.2026	
6	Рецензування	16.06.2026	
7	Захист на засіданні ЕК		

Здобувач вищої освіти

Зінчук
(підпис)

Владислав ЗІНЧУК

(Ім'я, ПРІЗВИЩЕ)

Керівник проекту (роботи)

Розум
(підпис)

Марина РОЗУМ

(Ім'я, ПРІЗВИЩЕ)

АНОТАЦІЯ

Об'єктом дослідження є специфіка побудови та функціонування систем прийняття рішень у сфері фінансового трейдингу на основі методів машинного навчання. Метою роботи є проектування та програмна реалізація автоматизованого інструментарію для аналізу фондового ринку та формування рекомендацій щодо проведення торговельних операцій. Практична значущість полягає у розробці алгоритмічного забезпечення, що інтегрує технічний аналіз та прогнозні моделі Machine Learning для мінімізації суб'єктивних ризиків трейдера. Реалізація програмного комплексу виконана мовою програмування Python у середовищі Jupyter Notebook з використанням бібліотек аналізу даних та машинного навчання (Scikit-learn). Результатом розробки є система, яка в режимі реального часу допомагає користувачу приймати обґрунтовані рішення про купівлю, продаж або утримання активів за допомогою генерованих торгових сигналів. Використання системи дозволяє підвищити точність прогнозування цінової динаміки та оптимізувати стратегію управління інвестиційним портфелем.

ANNOTATION

The object of the study is the specifics of building and functioning of decision-making systems in the field of financial trading based on machine learning methods. The purpose of the work is to design and programmatically implement an automated tool for analyzing the stock market and forming recommendations for conducting trading operations. The practical significance lies in the development of algorithmic support that integrates technical analysis and predictive models to minimize the subjective risks of the trader. The implementation of the software complex is carried out in the Python in the Jupyter Notebook environment using data analysis libraries and machine learning (Scikit-learn, TensorFlow/Keras). The result of the development is a system that in real time helps the user make informed decisions about buying, selling or holding assets using generated trading signals.

ЗМІСТ

ВСТУП.....	7
1 АНАЛІЗ ОСНОВНИХ ПОЛОЖЕНЬ ТА ДОЦІЛЬНОСТІ ТЕМАТИКИ ТРЕЙДИНГУ	10
1.1 Аналіз особливостей біржового трейдингу	10
1.2 Аналіз проблематики оцінки та прийняття рішень у сфері трейдингу	18
1.3 Перспективні шляхи використання машинного навчання та аналізу даних для трейдингу	22
1.4 Аналіз аналогів та обґрунтування вибору мови програмування.....	27
2 ОБГРУНТУВАННЯ МОДЕЛЕЙ ТА ПРОЄКТУВАННЯ СИСТЕМИ	32
2.1 Аналіз перспективних моделей підтримки ухвалення торгових рішень на основі машинного навчання.....	32
2.2 Опис обраного набору даних для аналізу	41
2.3 Розробка структури проєкту.....	44
3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ ДЛЯ ТРЕЙДИНГУ	57
3.1 Розробка програмної реалізації та інтерфейсу модулів системи	57
3.2 Дослідження та аналіз результатів роботи системи.....	71
4 ОХОРОНА ПРАЦІ	81
4.1 Роль і місце служби охорони праці в системі управління підприємством ІТ- галузі.....	81
4.2 Вплив роботизації та автоматизації технологічних процесів на безпеку праці.....	85
ВИСНОВКИ	88
ПЕРЕЛІК ПОСИЛАНЬ	89
ДОДАТОК А	92

ВСТУП

Сучасні товарних біржі та фондові ринки активно розвиваються і надають клієнтам різноманітні можливості для здійснення економічних операцій щодо купівлі та продажу різних активів. Слід вказати на те, що сучасна багатобортова торгівля отримувала новий зміст поряд із класичним способом. Тепер переважанням стали не фізичні товари чи готівка, а контракти на поставку товарів, цінностей (акції, облігації), валюта, боргові зобов'язання і фінансово-похідні – банки спрощені проценти та активи.

Різнманіття доступних біржових опцій свідчить про поступове універсалізацію цих платформи. Біржові майданчики поступово переходять з ролі виключно інструменту товарних взаємин на важливий аспект фінансового ринку.

Таким чином, у сучасному світі товарна біржа виконуватиме важливу роль як універсальна платформа для торгівлі ф'ючерсами та опціонними контрактами на поставку товарів з акцентом на операціях із цінними паперами. Багато угод проводяться через спеціалізовані електронні торговельні системи у форматі онлайн. Розвиток таких платформ є частиною глобального процесу цифровою трансформації економ сфери і може суттєво впливати на ринок працевлаштування та виробничих зв'язків. Тому проблематика прийняття торговельних рішень стала активно дослідженим напрямком, який висвітлено в даній роботі.

Метою роботи є розробка інформаційної системи прогнозування фінансових часових рядів для трейдингу.

Для досягнення цієї мети необхідно вирішити такі завдання:

- аналіз особливостей біржового трейдингу;
- аналіз проблематики оцінки та прийняття рішень у сфері трейдингу;
- перспективні шляхи використання машинного навчання та аналізу даних для трейдингу;
- аналіз аналогів та обґрунтування вибору мови програмування;
- аналіз перспективних моделей підтримки ухвалення торгових рішень на основі машинного навчання;

- опис обраного набору даних для аналізу;
- розробка структури проекту;
- розробка програмної реалізації та інтерфейсу модулів системи;
- дослідження та аналіз результатів роботи системи.

Об'єктом дослідження є моделі аналізу часових рядів.

Предметом дослідження є особливості використання машинного навчання для використання у трейдингу.

Теоретичною основою роботи є сучасні підходи до аналізу та прогнозування фінансових часових рядів із застосуванням методів машинного навчання та технічного аналізу, включаючи моделі рекурентних нейронних мереж, довго-короткочасної пам'яті та оновленого варіанту *gated recurrent unit*, а також класичні статистичні підходи на основі *ARIMA*-моделей. Окрему увагу приділено використанню технічних індикаторів, таких як ковзні середні, індикатор відносної сили, *MACD* та смуги Боллінджера, що дозволяють виявляти трендові закономірності, рівні волатильності та потенційні точки входу/виходу з ринку. Всі зазначені методи ґрунтуються на положеннях сучасної літератури з фінансової аналітики, прогнозування часових рядів та глибинного навчання.

Огляд наукових і технологічних джерел дозволив сформувати методологічну базу, що поєднує підходи до обробки фінансових даних у середовищі *Jupyter Notebook* із використанням *Python*-бібліотек для аналізу даних, побудови моделей машинного навчання та їх оптимізації.

Практична значущість розробленого програмного продукту полягає у створенні інструменту для аналізу біржових даних та підтримки прийняття торгових рішень на основі поєднання технічного аналізу та моделей машинного навчання. Система дозволяє здійснювати прогнозування цінкових рухів, формувати торгові сигнали та оцінювати ефективність різних моделей із можливістю їх порівняння за метриками якості, що знижує суб'єктивність прийняття рішень.

Розроблена система може бути застосована у дослідницьких проєктах з фінансової аналітики, навчальних курсах з машинного навчання та аналізу часових

рядів, а також у прикладних задачах алгоритмічного трейдингу, де необхідне швидке тестування та порівняння різних прогнозних моделей і стратегій.

Структура бакалаврської роботи складає 91 сторінок основного матеріалу, 26 джерел посилань та 1 додаток з лістингом програмного коду (7 сторінок).

1 АНАЛІЗ ОСНОВНИХ ПОЛОЖЕНЬ ТА ДОЦІЛЬНОСТІ ТЕМАТИКИ ТРЕЙДИНГУ

1.1 Аналіз особливостей біржового трейдингу

Сучасна торгова біржа в контексті побудови інформаційних систем прогнозування фінансових часових рядів для трейдингу розглядається як структурований багаторівневий ринок, у межах якого здійснюється укладання широкого спектра угод, пов'язаних із купівлею та продажем товарів, послуг, а також фінансових інструментів. Така інституція функціонує на основі чітко визначених регламентів та технологічних платформ, що забезпечують ефективну взаємодію між учасниками торгів та формують інформаційне середовище для подальшого аналізу і прогнозування ринкової динаміки.

Функціональне призначення торгової біржі охоплює низку ключових аспектів її діяльності [1]. Передусім біржа виступає як інфраструктурна основа організації торговельних процесів, надаючи технічні та комунікаційні засоби для взаємодії продавців і покупців, що забезпечує оперативність укладання угод та підвищує загальний рівень ліквідності фінансового ринку. Водночас важливою характеристикою є забезпечення відкритості механізмів ціноутворення, оскільки інформація щодо обсягів попиту і пропозиції, цінкових параметрів та умов угод є доступною для всіх учасників, що створює передумови для формування достовірних фінансових часових рядів.

Не менш суттєвим є зменшення ризиків невиконання зобов'язань контрагентами, що досягається завдяки впровадженню процедур клірингу, розрахунків і аналітичного супроводу угод, які реалізуються в межах біржової інфраструктури. Крім того, біржа виконує регулятивну функцію, встановлюючи стандарти та правила торгівлі, що регламентують поведінку учасників ринку, забезпечують уніфікацію контрактів і контроль за дотриманням встановлених норм.

Важливою складовою діяльності є сприяння залученню інвестиційних ресурсів, оскільки біржові механізми полегшують процес емісії та розміщення цінних паперів, що стимулює розвиток підприємств і економічне зростання загалом. Одночасно біржа виконує інформаційно-аналітичну функцію, забезпечуючи учасників актуальними даними щодо стану ринку, тенденцій зміни вартості активів та інших економічних індикаторів, що є критично важливим для побудови моделей прогнозування фінансових часових рядів та прийняття обґрунтованих інвестиційних рішень.

Додатково слід відзначити роль біржі у зниженні фінансових ризиків через використання похідних фінансових інструментів, зокрема ф'ючерсних та опціонних контрактів, які дозволяють реалізувати стратегії хеджування та мінімізувати вплив волатильності ринку. Таким чином, торгова біржа формує комплексне середовище, у якому поєднуються функції організації торгів, управління ризиками та генерації даних, що є базисом для функціонування сучасних інформаційних систем прогнозування. Узагальнена схема взаємодії сторін під час укладання угод на торговій біржі наведена на рис. 1.1.



Рисунок 1.1 – Узагальнена схема взаємодії сторін на біржі

У структурі сучасної торгової біржі як основи для формування та обробки фінансових часових рядів доцільно виокремити низку базових компонентів, що забезпечують цілісність її функціонування та надійність інформаційних потоків [2]. Кожен із цих елементів відіграє визначальну роль у забезпеченні коректності даних, які в подальшому використовуються в інформаційних системах прогнозування для задач трейдингу.

Одним із ключових елементів виступає розрахункова палата, яка реалізує клірингові механізми та відповідає за оброблення фінансових транзакцій між учасниками ринку. У межах її функціонування здійснюється списання та зарахування коштів на рахунки сторін угод, а також виконуються міжрахункові перекази, що гарантує завершеність фінансових розрахунків і знижує операційні ризики.

Важливе місце посідає торгова система, яка формує технологічне ядро біржі та забезпечує оброблення заявок на купівлю або продаж активів. Вона здійснює агрегування ордерів, їх узгодження та реєстрацію укладених угод, а також підтримує актуальність інформації у відповідних реєстрах. Особливого значення ця складова набуває в ситуаціях, коли на ринок одночасно надходять заявки з подібними ціновими параметрами, що потребує швидкого та коректного механізму зіставлення.

Не менш значущим компонентом є депозитарний центр, функціональне призначення якого полягає у веденні обліку цінних паперів, що належать учасникам торгів. Він забезпечує зберігання та актуалізацію інформації щодо акцій, облігацій та інших фінансових інструментів, виконує операції списання активів у разі їх продажу, а також здійснює звірку даних і фіксацію результатів проведених операцій. Така діяльність створює надійну інформаційну базу для подальшого аналізу та моделювання ринкових процесів.

Окрему роль у функціонуванні біржі відіграють механізми регулювання, які представлені як внутрішніми, так і зовнішніми інститутами. Внутрішнє регулювання реалізується через систему управлінських рішень і нормативних документів, що приймаються органами управління біржі, тоді як зовнішній

контроль здійснюється державними структурами відповідно до чинної нормативно-правової бази. Така багаторівнева система регулювання сприяє забезпеченню прозорості, стабільності та передбачуваності функціонування ринку.

Таким чином, наведені компоненти формують інтегровану інфраструктуру торгової біржі, яка забезпечує повний цикл оброблення торговельних і фінансових операцій та генерує структуровані дані, необхідні для побудови моделей прогнозування фінансових часових рядів. Склад правління типової торгової біржі наведено на рис. 1.2.



Рисунок 1.2 – Склад правління типової біржі

Сучасні торгові біржі у межах дослідження інформаційних систем прогнозування фінансових часових рядів доцільно розглядати з позиції класифікації за типами активів, що виступають об'єктами біржових операцій. Такий підхід дозволяє систематизувати джерела даних, які формують часові ряди

для подальшого аналізу та моделювання. Зокрема, до об'єктів торгів належать природні ресурси, серед яких нафта, газ, вугілля та руди, фінансові інструменти у вигляді акцій та облігацій, різноманітні боргові зобов'язання, включаючи векселі та чеки, а також валютні активи іноземних держав. Окрім цього, предметом біржових операцій виступають сировинні ресурси та споживчі товари, об'єкти інтелектуальної власності, зокрема результати розробок у сфері інформаційних технологій, а також сучасні цифрові активи, до яких належать електронні гроші та криптовалюти. До цієї сукупності також входять дорогоцінні камені та інші фінансові інструменти, що мають інвестиційну або боргову природу.

З урахуванням специфіки обігових активів біржі поділяються на кілька функціональних типів, що визначають характер формування та поведінки фінансових часових рядів. Зокрема, виокремлюють фондові, товарні, валютні біржі, ринки похідних фінансових інструментів, а також спеціалізовані платформи, орієнтовані на ринок праці. Кожен із цих сегментів має власні особливості ціноутворення, ліквідності та волатильності, що безпосередньо впливає на методи прогнозування.

Валютні ринки характеризуються високим рівнем ліквідності та динамічністю, оскільки операції купівлі та продажу валют здійснюються на основі аналізу їх котирувань, що відображають співвідношення обмінних курсів. Взаємозалежність валютних курсів значною мірою визначається макроекономічними факторами та відображає купівельну спроможність економічних агентів, що робить такі ринки важливим джерелом даних для побудови прогнозних моделей. На грошових ринках додатково реалізуються механізми формування ринкового курсу валют з урахуванням їх правового статусу та регуляторних обмежень.

Товарні біржі функціонують як організовані майданчики для регулярного здійснення оптових операцій із типовими товарами, що забезпечує умови, близькі до досконалої конкуренції, оскільки стандартизовані характеристики продукції мінімізують диференціацію пропозицій. Залежно від номенклатури товарів такі біржі можуть бути вузькоспеціалізованими або універсальними. Прикладами

універсальних торгових платформ виступають Чиказька товарна біржа та Лондонська біржа металів [3], діяльність яких формує значні масиви історичних даних для аналізу ринкових тенденцій.

Фондові біржі забезпечують обіг цінних паперів, включаючи акції, облігації та інші фінансові інструменти, виконуючи роль центрального елемента ринкової інфраструктури. Вони створюють умови для професійної діяльності брокерів і дилерів, а також сприяють ефективному функціонуванню інвестиційних фондів. Завдяки централізованим механізмам обліку та високому рівню прозорості спрощується процес укладання угод і забезпечується доступ до достовірної інформації. Крім того, фондові біржі підтримують як первинне розміщення цінних паперів, так і їх подальший обіг, що формує безперервні фінансові часові ряди. У результаті забезпечується природне регулювання ринку та високий рівень ліквідності інвестиційних активів. Актуальні рейтинги бірж формуються з урахуванням їх ринкової капіталізації, що відображає масштаб і значущість відповідних торгових майданчиків. Управління торговими операціями здійснюється за участю брокерів і дилерів, а типовими прикладами таких бірж є Нью-Йоркська та Лондонська фондові біржі.

Окрему категорію становлять біржі деривативів, які спеціалізуються на торгівлі похідними фінансовими інструментами. У межах таких платформ укладаються угоди щодо ф'ючерсних контрактів і опціонів, що дозволяє фіксувати умови купівлі або продажу базових активів у майбутньому за заздалегідь визначеними параметрами. Основне призначення цих інструментів полягає у хеджуванні ризиків та оптимізації інвестиційних стратегій, що є важливим аспектом у задачах прогнозування. Функціонування таких бірж передбачає наявність клірингових механізмів, які виконують роль посередника та гаранта виконання зобов'язань між сторонами угод, що сприяє зниженню контрагентських ризиків. Стандартизація контрактів за строками, обсягами та умовами виконання спрощує процес торгівлі та підвищує ефективність оброблення даних. Використання маржинальних механізмів як гарантійного забезпечення дозволяє

контролювати рівень ризику та підтримувати стабільність функціонування біржі. Прикладами таких торгових платформ є Eurex, NYSE Euronext та ICE Futures [4].

У контексті цифровізації фінансових ринків та розвитку інформаційних систем прогнозування фінансових часових рядів останніми роками спостерігається формування спеціалізованих криптовалютних бірж, які функціонують як віртуальні торгові платформи. Їх основне призначення полягає у виконанні посередницьких функцій під час обміну електронних грошових ресурсів на різні види як фіатних, так і нефіатних активів. Такі платформи забезпечують інтеграцію різних платіжних систем та дозволяють здійснювати конверсію електронних валют у реальні активи або інші криптоактиви, що значно розширює спектр доступних фінансових інструментів для трейдингу.

Криптовалютні біржі диференціюються залежно від функціональних можливостей і підтримуваних інструментів. Частина з них орієнтована на операції з біткоїном та його базовими похідними, тоді як інші забезпечують обмін криптоактивів на національні валюти різних країн, що підвищує їхню інтегрованість у глобальну фінансову систему. Така багатоваріантність створює додаткові джерела даних для аналізу та формування часових рядів, які характеризуються підвищеною волатильністю та специфічними закономірностями.

Під час проведення торгових сесій на біржах до процесу залучаються не лише професійні посередники у вигляді брокерів і дилерів, але й управлінські структури, які забезпечують організацію та контроль процесів торгівлі фінансовими інструментами. Безпосереднє укладання угод може здійснюватися трейдерами, що діють як принципали та мають відповідні ліцензії, отримані згідно з установленими кваліфікаційними вимогами та чинним законодавством. Окрім цього, важливу роль відіграють сторонні організації, зокрема банківські установи, які забезпечують облік прав власності та контроль за фінансовими операціями, формуючи надійну інфраструктуру для функціонування біржових систем.

Аналіз сучасних тенденцій розвитку біржових ринків свідчить про наявність суттєвих трансформаційних процесів у цій сфері [2]. Зокрема, спостерігається посилення глобалізаційних процесів, що проявляється в інтеграції окремих

торгових майданчиків у великі біржові об'єднання. Водночас відбувається еволюція організаційно-правових моделей, у межах якої провідні біржі трансформуються з некомерційних структур у комерційно орієнтовані інституції. Значного поширення набуває електронна торгівля, що супроводжується активним розвитком біржових інформаційних систем і технологічних рішень, які забезпечують автоматизацію процесів укладання угод і оброблення даних.

Додатково відзначається тенденція до консолідації ринків, що проявляється у поглинанні позабіржових структур організованими біржами, а також зростання універсалізації торгових платформ за рахунок розширення переліку доступних активів. Водночас відбувається трансформація товарних бірж, які дедалі більше інтегруються у фінансові ринки, що супроводжується зростанням ролі похідних інструментів, зокрема товарних ф'ючерсів, у загальному обсязі біржових операцій.

Попри активний розвиток, сучасні торгові біржі стикаються з низкою викликів, що впливають на їхню стабільність та ефективність функціонування. Одним із ключових ризиків є зростання кількості кібератак, спрямованих на фінансові платформи, що підвищує загрози безпеці даних та може негативно позначатися на рівні довіри з боку учасників ринку. Суттєвим фактором є також неоднорідність регуляторного середовища, оскільки відмінності у національних і міжнародних нормах ускладнюють діяльність глобальних бірж і потребують постійного оновлення процедур відповідності законодавству, включаючи механізми верифікації учасників.

Важливим викликом виступає швидкий розвиток інформаційних технологій, який змушує біржі постійно модернізувати технічну інфраструктуру для забезпечення високої швидкості оброблення операцій та їх надійності. Перехід до нових платформ супроводжується ризиками технічних збоїв і можливими порушеннями безперервності торгів. Одночасно ринки можуть зазнавати зниження ліквідності та зростання волатильності, що впливає на стабільність фінансових часових рядів і ускладнює процес їх прогнозування.

Конкурентний тиск між біржами, альтернативними торговими системами та позабіржовими ринками зумовлює необхідність зниження комісійних витрат і

підвищення якості послуг. Додатковий вплив чинить популяризація децентралізованих фінансових платформ і криптовалютних бірж, що частково відволікає потоки ліквідності від традиційних ринків. Суттєвими залишаються й геополітичні фактори, включаючи відмінності у правових системах, торговельні обмеження та санкційні режими, які можуть обмежувати доступ до окремих активів і дестабілізувати роботу біржових інституцій.

Окремо слід враховувати вплив людського чинника, який проявляється у можливості виникнення конфліктів інтересів, маніпуляцій ринком та спекулятивних стратегій, що здатні негативно впливати на прозорість і ефективність торгівлі. Це обумовлює необхідність жорсткого регуляторного контролю та впровадження сучасних інформаційних систем моніторингу, які забезпечують виявлення аномалій у фінансових часових рядах і підвищують надійність функціонування біржового середовища [5].

1.2 Аналіз проблематики оцінки та прийняття рішень у сфері трейдингу

Торгівля на фондових біржах супроводжується значним рівнем складності та транзакційних витрат, що необхідно враховувати під час побудови ефективних стратегій у межах інформаційних систем прогнозування фінансових часових рядів. Сукупність проблем доцільно розглядати крізь призму внутрішніх чинників, пов'язаних з управлінням ризиками та поведінковими аспектами, а також зовнішніх впливів, зумовлених ринковим середовищем і регуляторними вимогами.

Ключовим викликом виступає управління ризиками, яке ускладнюється активним використанням позикового капіталу для підвищення дохідності операцій. Застосування похідних фінансових інструментів у цілях хеджування потребує високого рівня професійної підготовки, оскільки ефективність таких підходів безпосередньо залежить від здатності коректно оцінювати волатильність активів і потенційні втрати, що виникають унаслідок цінових коливань.

Вагомим фактором залишається психологічна складова торгівлі, яка проявляється у впливі емоційних реакцій на процес прийняття рішень в умовах

динамічних змін ринку. Нестабільність котирувань може провокувати імпульсивні дії, що знижує ефективність торгових стратегій, тому дотримання заздалегідь визначеного плану та системного підходу до аналізу є критично важливими.

Ринкове середовище також формує додаткові обмеження, зокрема через високий рівень конкуренції та змінну ліквідність. Необхідність здійснення операцій у періоди підвищеної волатильності вимагає значної концентрації та швидкості прийняття рішень, тоді як недостатня глибина ринку може ускладнювати виконання великих ордерів за бажаними цінами.

Окремий рівень складності пов'язаний із дотриманням регуляторних вимог, що передбачає значні витрати ресурсів на ведення документації, звітність і забезпечення відповідності нормативним стандартам. Ситуація ускладнюється відмінностями у правових підходах різних юрисдикцій, що створює додаткові бар'єри для учасників глобальних ринків [6].

Додаткове навантаження у процесі прийняття рішень формують процедури фундаментального та технічного аналізу, які потребують оброблення значних обсягів даних і перевірки їх достовірності. Оцінювання ринкових тенденцій повинно враховувати вплив макроекономічних показників, інформаційних подій і геополітичних факторів, що безпосередньо відображаються у фінансових часових рядах. У цьому контексті особливого значення набуває інтеграція аналітичних інструментів у інформаційні системи, що дозволяє підвищити обґрунтованість торгових рішень і врахувати ризики, закладені у вибрану стратегію (рис. 1.3).

Процес підтримки ухвалення торгових рішень у межах сучасних біржових систем доцільно формалізувати як послідовну модель оброблення даних і формування дій, що забезпечує підвищення точності прогнозування фінансових часових рядів. У загальному вигляді така модель охоплює етапи акумулювання інформації, її аналітичного опрацювання, побудови стратегії, виконання угод і подальшої оцінки результативності.



Рисунок 1.3 – Схема торгового плану трейдера

Початковий етап передбачає збір історичних і поточних даних, включаючи часові ряди цін активів, обсяги торгів, фінансову звітність емітентів та оперативні ринкові котирування. Додатково інтегруються інформаційні потоки новин, макроекономічні індикатори, аналітичні огляди й експертні оцінки, що формує багатовимірну інформаційну базу для подальшого моделювання. Наступний етап пов'язаний з аналітичною обробкою агрегованих даних, у межах якої здійснюється оцінювання фінансового стану об'єктів інвестування, дослідження ринкових трендів, застосування технічних індикаторів і врахування поведінкових сигналів, зокрема з цифрових інформаційних середовищ.

На основі отриманих результатів формується цільова торгова стратегія, що передбачає визначення інвестиційних цілей, допустимого рівня ризику та горизонту інвестування, а також вибір інструментів і підходів до торгівлі. У цьому контексті відбувається формалізація параметрів угод, встановлення обмежень для контролю збитків і механізмів фіксації прибутку, що є критично важливим для автоматизованих систем трейдингу. Реалізація стратегії здійснюється через торгові платформи, які забезпечують виконання ордерів, моніторинг позицій і динамічну адаптацію до змін ринкового середовища. Завершальним етапом виступає ретроспективний аналіз, у межах якого зіставляються планові та фактичні

результати, ідентифікуються відхилення та коригуються параметри стратегій для підвищення їх ефективності в майбутньому [7].

Важливо враховувати часову структуру біржової активності, яка характеризується поділом на торгові сесії протягом доби, що впливає на інтенсивність операцій і параметри фінансових часових рядів. Додаткову складність формують різні механізми торгівлі, включаючи безперервні та дискретні аукціони, а також спеціалізовані угоди, що зумовлює необхідність адаптації стратегій до конкретних умов ринку. Найбільш поширеними підходами залишаються короткострокові стратегії, зокрема скальпінг, орієнтований на велику кількість швидких операцій із незначним прибутком, і дейтрейдинг, що передбачає закриття позицій у межах однієї торгової сесії [8].

У практиці біржової торгівлі використовуються різні типи заявок, які визначають механіку виконання угод і впливають на характеристики часових рядів. Ринкові заявки забезпечують негайне виконання за поточною ціною, однак можуть супроводжуватися ціновими втратами при великих обсягах. Лімітні заявки дозволяють контролювати ціну виконання, що є доцільним в умовах підвищеної волатильності. Умовні заявки реалізують автоматичне відкриття або закриття позицій за досягнення заданих рівнів, а плаваючі стоп-механізми забезпечують динамічне коригування захисних параметрів із метою збереження прибутку [1].

Суттєвим фактором ризику виступають інфраструктурні обмеження, пов'язані з роботою торгових серверів і мережевих систем. Високе обчислювальне навантаження, характерне для алгоритмічного трейдингу, у поєднанні з можливими збоями обладнання або каналів зв'язку може призводити до затримок виконання операцій чи втрати даних. Подібні технічні порушення періодично виникають як на рівні бірж, так і брокерських платформ, що негативно впливає на надійність торгових процесів [9]. Додаткові ускладнення виникають унаслідок помилок і аномалій у торгових операціях. До них належать некоректні параметри заявок, збої у взаємодії програмних інтерфейсів, порушення специфікацій фінансових інструментів і відхилення від регламенту торгів. Також фіксуються випадки нетипової поведінки заявок, перевищення встановлених лімітів стратегій

або обмежень торгових рахунків, що може призводити до фінансових втрат і спотворення ринкових даних [2].

1.3 Перспективні шляхи використання машинного навчання та аналізу даних для трейдингу

Процеси збору та оброблення даних у межах підтримки ухвалення торгових рішень на біржах відіграють визначальну роль у забезпеченні ефективності трейдингу та якості прогнозування фінансових часових рядів. У практичному вимірі ці задачі реалізуються різними підходами, серед яких особливого значення набуває автоматизація на основі математичних методів, моделей і алгоритмів, що дозволяє підвищити швидкість і обґрунтованість прийнятих рішень.

Серед ключових напрямів виділяється прогнозування часових рядів, яке орієнтоване на передбачення динаміки вартості фінансових активів. Для розв'язання таких задач традиційно застосовуються інструменти математичної статистики, зокрема моделі типу AR, MA, ARMA та ARCH, що забезпечують формалізацію залежностей у часових даних. Додатково використовуються підходи технічного аналізу та теорії прийняття рішень, які в ряді випадків демонструють практичну ефективність. Розвиток отримали також методи статистичної фізики, включаючи концепції стохастичного та детермінованого хаосу у поєднанні з марковськими процесами, що дозволяє враховувати складну нелінійну природу ринкових процесів [8–10].

Водночас зазначені підходи мають обмеження, зумовлені їх відносною статичністю та недостатньою здатністю враховувати багатofакторність і невизначеність фінансових ринків. Це призводить до зниження точності прогнозів, особливо в умовах високої волатильності. Додаткові труднощі виникають на етапі практичної реалізації, оскільки зростає обчислювальна складність, ускладнюється інтеграція різнорідних моделей і виникають проблеми сумісності в програмних системах. Узагальнені підходи до прогнозування для підтримки торгових рішень наведено на рис. 1.4. У зв'язку з цим доцільним є зміщення акценту на

використання методів штучного інтелекту, що обумовлено стрімким розвитком інформаційних технологій та обчислювальних ресурсів. Сучасні підходи ШІ дозволяють ефективно обробляти великі обсяги багатовимірних даних, виявляти приховані закономірності та адаптуватися до змін ринкового середовища. Це відкриває нові можливості для підвищення точності прогнозування фінансових часових рядів і вдосконалення систем підтримки ухвалення рішень у сфері біржового трейдингу [11].



Рисунок 1.4 – Методи та моделі прогнозування для трейдингу

Інтелектуальний аналіз даних як складова сучасних підходів штучного інтелекту являє собою комплексний процес виявлення прихованих закономірностей, трендів і залежностей у великих масивах різномірної інформації, що має критичне значення для побудови інформаційних систем прогнозування фінансових часових рядів. У межах цього підходу реалізуються задачі класифікації, кластеризації, регресійного аналізу, пошуку аномалій і асоціативних залежностей, а також зменшення розмірності даних з метою отримання релевантних знань для підтримки ухвалення рішень. Завдяки адаптивності такі методи активно

застосовуються у фінансовій аналітиці, біржовій торгівлі та суміжних прикладних сферах.

Одним із найбільш перспективних напрямів є використання алгоритмів машинного навчання для автоматизації торгових операцій. У цьому випадку програмні агенти здатні обробляти значні обсяги ринкових даних і формувати обґрунтовані рішення щодо купівлі або продажу фінансових інструментів. Гнучкість програмної реалізації та здатність масштабування в розподілених обчислювальних середовищах забезпечують ефективну підтримку інвесторів у складних умовах біржового середовища. Машинне навчання, будучи підмножиною інтелектуального аналізу даних, орієнтоване на побудову моделей, які навчаються на основі статистичних вибірок і вдосконалюють власну точність у процесі накопичення досвіду, що дозволяє виявляти приховані патерни та формувати прогнози.

Базовими елементами таких підходів є дані, моделі та алгоритмічні процедури їх навчання [12]. Дані можуть мати як структурований, так і неструктурований характер, а їх обсяг і якість безпосередньо впливають на результативність моделей. Моделі виступають формалізованим відображенням залежностей між ознаками та цільовими змінними, тоді як алгоритми визначають механізми їх побудови та оптимізації. У задачах біржового трейдингу ці моделі переважно орієнтовані на прогнозування, зокрема оцінювання майбутніх значень цін або напрямів зміни котирувань.

Серед класичних підходів машинного навчання, що застосовуються у фінансовій аналітиці, виділяються ансамблеві методи, зокрема випадковий ліс, який забезпечує підвищену точність за рахунок комбінування дерев рішень, імовірнісні моделі типу наївного байєсівського класифікатора, а також метод опорних векторів, ефективний для задач із розміченими даними. Окреме місце займає модель ARIMA як інструмент аналізу часових рядів, що дозволяє враховувати тренди та сезонні компоненти. Водночас зазначені підходи мають обмеження, пов'язані зі стохастичною природою фінансових ринків, що може знижувати точність прогнозування в умовах високої невизначеності [13].

У практиці підтримки ухвалення торгових рішень інтелектуальний аналіз даних і машинне навчання забезпечують низку важливих функціональних можливостей [11]. Вони дозволяють здійснювати багатofакторний аналіз історичних фінансових даних, включаючи ціни активів і обсяги торгів, що створює основу для побудови прогнозних моделей. Використання паралельних обчислень забезпечує безперервну обробку поточкових даних у режимі реального часу, що є критично важливим для високочастотного трейдингу. Прогнозна аналітика сприяє формуванню сценаріїв розвитку ринку, тоді як оптимізаційні підходи дозволяють тестувати й адаптувати торгові стратегії на основі історичних даних.

Крім того, сучасні методи обробки природної мови дають змогу аналізувати новинні потоки та інформацію із соціальних мереж для виявлення ринкових настроїв, що розширює спектр факторів, які враховуються під час прогнозування. Інструменти управління ризиками забезпечують оцінювання волатильності та інших критичних параметрів, сприяючи зниженню потенційних втрат. Додатково застосовуються графові моделі, які дозволяють досліджувати складні взаємозв'язки між учасниками ринку та фінансовими інструментами.

Серед сучасних напрямів розвитку штучного інтелекту особливе місце посідає глибоке навчання, яке розглядається як більш складна та функціонально розширена форма машинного навчання. Його основою є використання штучних нейронних мереж, здатних моделювати складні залежності за аналогією з когнітивними процесами, що дозволяє ефективно аналізувати великі обсяги фінансових даних у задачах прогнозування часових рядів [13]. Багатошарова структура таких моделей забезпечує послідовне виділення прихованих ознак і формування інформативних представлень, що підвищує якість аналітики. Класифікацію типових архітектур моделей ШНМ наведено на рис. 1.5.

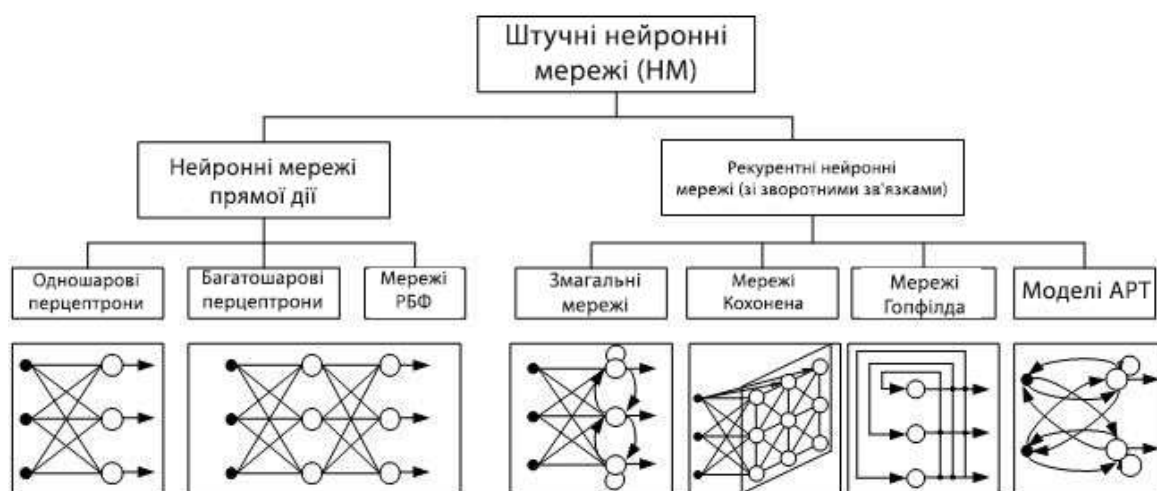


Рисунок 1.5 – Класифікація типових архітектур ШНМ

У практиці біржового трейдингу застосовуються різні архітектури нейронних мереж, орієнтовані на специфіку фінансових даних. Рекурентні мережі демонструють ефективність при роботі з послідовностями, оскільки враховують історичні значення показників. Їх удосконалені варіанти, зокрема LSTM, забезпечують збереження релевантної інформації на тривалих інтервалах, що є важливим для виявлення довгострокових трендів, тоді як моделі типу GRU оптимізують обчислювальні витрати [10].

Згорткові мережі, попри початкову орієнтацію на обробку зображень, адаптовані для аналізу часових рядів і виявлення локальних закономірностей у динаміці цін. Вони ефективно використовуються для ідентифікації графічних патернів, що можуть сигналізувати про зміну ринкових тенденцій. Автокодуювальники застосовуються для зниження розмірності даних і виділення інформативних ознак, що дозволяє виявляти аномалії та фільтрувати шумові сигнали. Генеративно-змагальні мережі забезпечують синтез нових даних на основі наявних вибірок, що є корисним для моделювання рідкісних ринкових сценаріїв і тестування торгових стратегій. Трансформерні архітектури, спочатку орієнтовані на текстову інформацію, ефективно інтегруються у задачі фінансового аналізу, зокрема для оброблення новинних потоків і врахування складних часових залежностей [13].

Вибір конкретної моделі визначається характером даних і поставленими завданнями, однак у задачах прогнозування фінансових часових рядів особливу ефективність демонструють моделі типу LSTM, що поєднують здатність до запам'ятовування довготривалих залежностей із високою точністю прогнозів.

Застосування нейронних мереж у системах підтримки ухвалення торгових рішень забезпечує низку суттєвих переваг [11]. Насамперед підвищується точність прогнозування завдяки здатності моделей виявляти складні нелінійні залежності. Оброблення великих обсягів даних відбувається з високою швидкістю, що дозволяє формувати аналітичні висновки в режимі, наближеному до реального часу. Крім того, забезпечується адаптація торгових стратегій до змін ринкового середовища, формування персоналізованих рекомендацій і зниження витрат за рахунок автоматизації аналітичних процесів.

Отже, інтеграція методів інтелектуального аналізу даних, машинного та глибокого навчання у структуру інформаційних систем прогнозування фінансових часових рядів створює передумови для підвищення ефективності біржової діяльності. Це проявляється у зростанні точності прогнозів, оптимізації торгових стратегій і більш ефективному управлінні ризиками, що в сукупності сприяє прийняттю обґрунтованих рішень у процесі трейдингу.

1.4 Аналіз аналогів та обґрунтування вибору мови програмування

У межах проведення досліджень у сфері аналізу фінансових часових рядів доцільно виокремити два взаємопов'язані етапи: дослідницький аналіз даних та аналітичний етап, що передбачає побудову і тестування прогнозних моделей. Перший етап орієнтований на первинне вивчення структури даних, виявлення аномалій, оцінку статистичних характеристик і формування гіпотез щодо подальшого моделювання. Другий етап пов'язаний із безпосереднім застосуванням методів машинного та глибокого навчання для прогнозування поведінки фінансових інструментів і побудови торгових стратегій.

Для реалізації зазначених етапів використовуються як спеціалізовані аналітичні платформи, так і інструменти програмування загального призначення. Серед них варто виділити середовища RapidMiner і KNIME, які забезпечують швидке прототипування моделей, а також мову програмування Python, що дозволяє реалізовувати гнучкі алгоритмічні рішення. Додатково в прикладних задачах біржової торгівлі можуть застосовуватися торгові платформи, зокрема NinjaTrader, які забезпечують інтеграцію аналітичних моделей із реальними ринковими даними та виконанням торгових операцій.

Платформа RapidMiner є потужним інструментом інтелектуального аналізу даних, який широко використовується для дослідницького аналізу та побудови прототипів прогнозних моделей. Вона забезпечує повний цикл обробки даних, починаючи від підготовки та очищення і завершуючи візуалізацією результатів і оцінюванням моделей [14]. Особливістю системи є візуальний підхід до конструювання аналітичних процесів, що реалізується через блокову (drag-and-drop) структуру побудови моделей.

До основних переваг RapidMiner належать зручний інтерфейс, що спрощує розробку аналітичних процесів без необхідності глибокого програмування; наявність широкого набору інструментів для попередньої обробки даних, включаючи нормалізацію, фільтрацію та виявлення аномалій; підтримка інтеграції з різними джерелами даних, такими як бази даних, файли CSV, Excel та зовнішні API; можливість потокової обробки даних у режимі реального часу, що є особливо важливим для фінансових застосувань; а також розвинені засоби візуалізації та серіалізації результатів [15].

Водночас система має низку обмежень. Зокрема, вона характеризується високими вимогами до обчислювальних ресурсів, особливо до оперативної пам'яті, що може ускладнювати роботу з великими наборами даних. Також відзначається обмежена можливість інтеграції користувацького коду, що знижує гнучкість у складних прикладних сценаріях. Додатковим недоліком є складність експорту навчених моделей для подальшого використання в зовнішніх системах, а також необхідність регулярного оновлення для підтримки актуальності функціоналу [16].

Альтернативною платформою є KNIME, яка також належить до класу інструментів інтелектуального аналізу даних і забезпечує широкий спектр можливостей для дослідницького та прикладного аналізу. Вона дозволяє реалізовувати повний цикл обробки даних із акцентом на модульність і розширюваність, що робить її придатною для складних аналітичних задач.

Функціональні можливості KNIME включають попередню обробку даних, зокрема завантаження з різних форматів (CSV, JSON, XML, JDBC), очищення, обробку пропусків, трансформацію типів даних та фільтрацію; статистичний аналіз із розрахунком ключових характеристик, таких як середнє значення, медіана, стандартне відхилення та кореляційний аналіз; візуальний аналіз даних за допомогою графічних представлень, включаючи гістограми, діаграми розсіювання та теплові карти; а також інструменти агрегації та трансформації даних із можливістю інтерактивного відображення результатів [17].

Серед переваг KNIME слід відзначити високу продуктивність у порівнянні з аналогами, ефективне використання ресурсів, гнучку систему розширення функціоналу через плагіни, відкриту ліцензію та регулярні оновлення, що забезпечують розвиток платформи. Водночас система має певні недоліки, зокрема обмежені можливості інтерактивної візуалізації, необхідність встановлення значної кількості додаткових модулів для роботи з реальними даними, а також складність конфігурації окремих компонентів і відсутність підтримки деяких мов програмування для розширення функціональності [18].

Попри значні можливості сучасних аналітичних платформ і спеціалізованих систем, перспективним напрямом залишається використання мов програмування загального призначення, зокрема Python, яка на сьогодні є де-факто стандартом у сфері Data Science та прикладного машинного навчання. Її популярність зумовлена універсальністю, високим рівнем абстракції та широкою підтримкою бібліотек, що дозволяє ефективно реалізовувати як дослідницькі, так і промислові рішення, включно з алгоритмічною торгівлею та інтеграцією з торговими платформами.

Python забезпечує потужну екосистему інструментів для обробки даних, побудови моделей машинного навчання та автоматизації аналітичних процесів.

Серед найбільш поширених бібліотек слід виділити TensorFlow, Keras та Scikit-learn, які спрощують розробку моделей глибокого та класичного машинного навчання, забезпечуючи при цьому високу гнучкість і масштабованість рішень. Незважаючи на інтерпретований характер мови, продуктивність обчислень значною мірою оптимізується за рахунок використання низькорівневих реалізацій на основі C/C++ та апаратного прискорення через CUDA, що є особливо важливим у задачах роботи з нейронними мережами та великими масивами фінансових даних.

До основних переваг використання Python у системах трейдингу належать гнучкість і простота синтаксису, що значно прискорює розробку та тестування торгових стратегій; розвинена екосистема бібліотек для статистичного аналізу, машинного навчання та фінансових обчислень; активна спільнота розробників, яка забезпечує постійний розвиток інструментів і наявність великої кількості навчальних матеріалів; можливість інтеграції з біржовими API, базами даних і хмарними сервісами; а також підтримка спеціалізованих фреймворків для алгоритмічної торгівлі, таких як Backtrader, Zipline та QuantLib. Додатковою перевагою є можливість роботи з потоковими даними через сучасні технології обміну повідомленнями, зокрема REST API, WebSocket, Kafka та RabbitMQ.

Водночас Python має й певні обмеження. Зокрема, динамічна типізація може ускладнювати контроль за структурою даних у великих проєктах, а інтерпретована природа мови зумовлює нижчу продуктивність у порівнянні з компільованими мовами. Також ускладненим є процес розгортання готових програмних рішень у промисловому середовищі, що потребує додаткових інструментів пакування та контейнеризації. Окрему увагу слід приділяти управлінню залежностями, оскільки стабільність системи значною мірою залежить від коректного контролю версій бібліотек і середовищ виконання.

Завдяки перевагам, використання Python як основного інструменту реалізації систем трейдингу є обґрунтованим і ефективним, оскільки забезпечує баланс між швидкістю розробки, функціональністю та можливістю інтеграції сучасних методів штучного інтелекту.

Висновки. У межах даного розділу здійснено комплексний аналіз сучасних методів і технологій застосування штучного інтелекту та машинного навчання у системах підтримки ухвалення торгових рішень на біржах. Розглянуто специфіку функціонування сучасних торгових платформ, а також основні принципи організації біржової торгівлі. Виявлено ключові проблеми, пов'язані з обробкою фінансових часових рядів, прогнозуванням ринкової динаміки та реалізацією торгових стратегій в умовах високої невизначеності. Окрему увагу приділено особливостям застосування інтелектуального аналізу даних як базового інструменту формування рішень у фінансовій сфері. Показано, що машинне навчання є ефективним інструментом підвищення якості прогнозування, однак потребує подальшого вдосконалення з урахуванням стохастичної природи фінансових ринків і складності багатofакторних залежностей. На основі проведеного аналізу обґрунтовано доцільність подальших досліджень у напрямі практичної реалізації розглянутих підходів із використанням сучасних інструментів програмної розробки та штучного інтелекту..

2 ОБГРУНТУВАННЯ МОДЕЛЕЙ ТА ПРОЄКТУВАННЯ СИСТЕМИ

2.1 Аналіз перспективних моделей підтримки ухвалення торгових рішень на основі машинного навчання

У межах дослідження інформаційної системи прогнозування фінансових часових рядів для трейдингу особливого значення набувають сучасні підходи підтримки ухвалення торгових рішень, що базуються на алгоритмах машинного навчання. Упродовж останніх років такі підходи інтенсивно впроваджуються для вирішення прикладних завдань у сфері фінансового менеджменту та обліку, зокрема у контексті функціонування біржових ринків. Їх використання орієнтоване на автоматизацію процедур аналізу великих масивів даних, реалізацію предиктивних моделей та підвищення обґрунтованості вибору торгових стратегій.

Фінансові дані, що циркулюють на біржах, характеризуються вираженою часовою динамікою, тобто ключові параметри, зокрема ціна активів, рівень попиту чи показники волатильності, змінюються у часі. З огляду на це, доцільним є застосування методів аналізу часових рядів, особливо в задачах короткострокового прогнозування. Збільшення горизонту прогнозу, як правило, супроводжується необхідністю ітеративного обчислення нових значень на основі попередніх оцінок, що допустимо лише в обмежених часових інтервалах, оскільки зі зростанням горизонту прогнозування точність моделей має тенденцію до зниження [12].

Суттєвим аспектом моделювання є наявність обмежень, пов'язаних із необхідністю узгодження умов прогнозування зі стаціонарними властивостями процесів, що передбачає сталість розподілу ймовірностей у часі. Під час вибору та налаштування моделей машинного навчання для розв'язання регресійних задач слід враховувати, що реальні фінансові, економічні та торгові часові ряди зазвичай демонструють нестационарність, однак можуть зберігати однорідність структури. Це означає, що після вилучення детермінованої складової залишковий процес може розглядатися як стаціонарний часовий ряд [10].

Поряд із класичними підходами машинного навчання, доцільно враховувати спеціалізовані статистичні моделі, зокрема ARIMA та GARCH, а також архітектури штучних нейронних мереж із механізмами довготривалої пам'яті, серед яких особливе місце займають LSTM-моделі, що будуть детально розглянуті у подальших розділах роботи [13].

Модель ARIMA являє собою узагальнення підходів авторегресії та ковзного середнього і призначена для формування прогнозів часових рядів різної природи. У практичних задачах вона широко використовується для аналізу динаміки цін фінансових інструментів та інших економічних показників. Як представник авторегресійного класу, ця модель реалізує прогнозування значень у визначені моменти часу з опорою на історичні дані. Важливою особливістю є орієнтація не стільки на абсолютні значення процесу, скільки на їх відносні зміни. Згладжування випадкових коливань у часовому ряді досягається шляхом застосування методу ковзного середнього, відповідно до якого кожне значення замінюється середнім арифметичним сусідніх спостережень.

Ілюстрацію застосування ARIMA для прогнозування фінансового часового ряду наведено на рис. 2.1. На графічному поданні можна простежити окремі часові інтервали, що відображають етап формування моделі на основі фактичних статистичних даних, фазу генерації прогнозних значень із подальшою оцінкою їх відповідності реальним спостереженням, а також інтервал прогнозування, який використовується для інтерпретації поведінки тренду та оцінки його подальшого розвитку [12].

Слід підкреслити, що для досягнення високої точності прогнозування важливим є врахування зовнішніх чинників, які можуть впливати на динаміку фінансових показників і формувати реалістичну поведінку моделей. Не менш значущими залишаються процеси збору, попередньої обробки та підготовки даних, особливо у випадках, коли досліджувані часові ряди характеризуються нестационарністю [14].

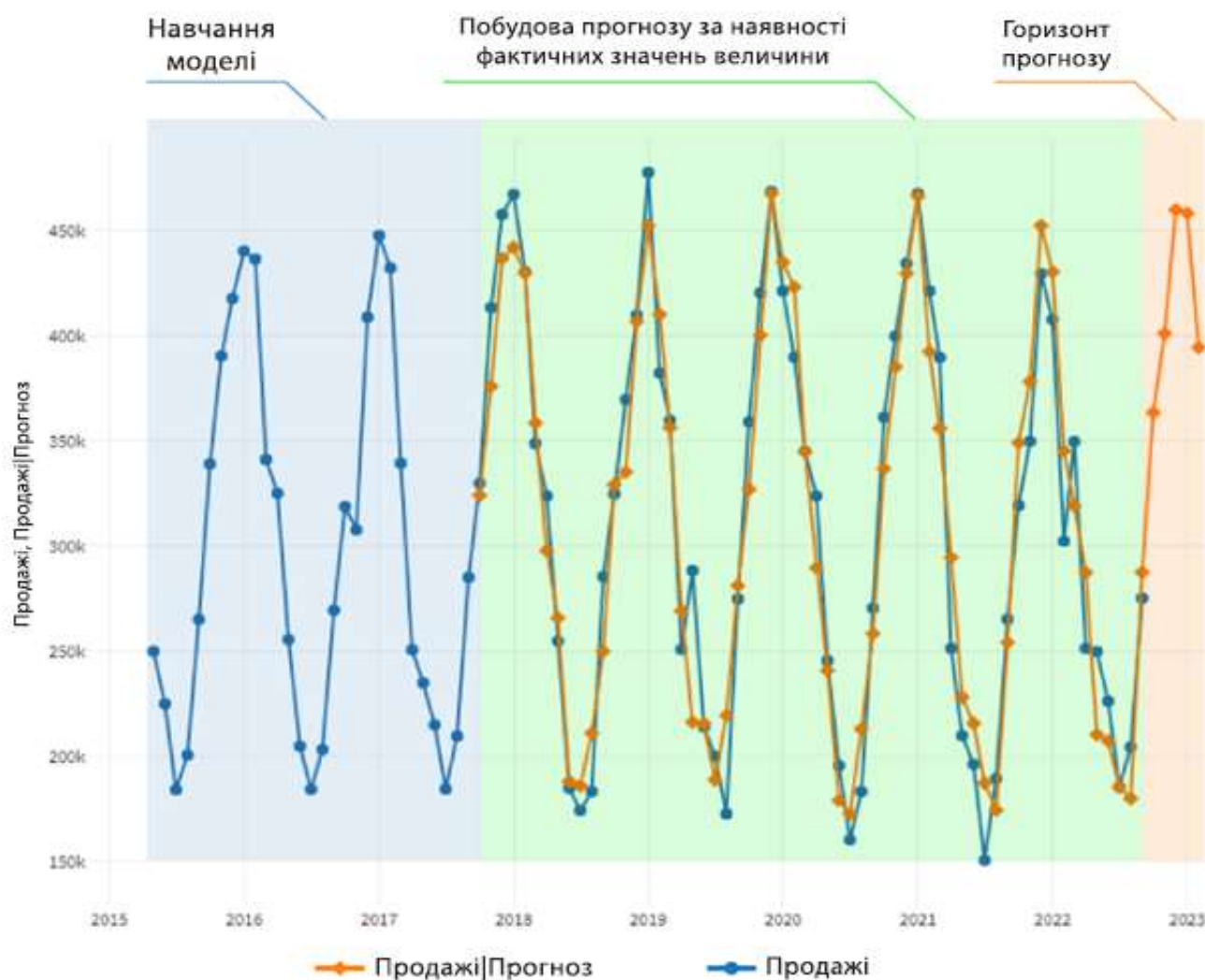


Рисунок 2.1 – Приклад використання моделі ARIMA

Алгоритм підбору параметрів моделі ARIMA представлено на рис. 2.2. У практичній площині побудова цієї моделі для аналізу властивостей фінансового часового ряду реалізується як послідовність взаємопов'язаних процедур, що забезпечують її адаптацію до специфіки вхідних даних [13]. На початковому етапі здійснюється формування тестової конфігурації моделі, яка виступає базовою гіпотезою щодо структури досліджуваного процесу. Далі проводиться оцінювання параметрів моделі та перевірка її адекватності емпіричним даним, що передбачає підбір значень параметрів (p , d , q) і аналіз відповідності отриманих результатів реальній динаміці ряду. Завершальний етап пов'язаний із практичним використанням налаштованої моделі для генерації прогнозних значень, що

дозволяє оцінити її прикладну ефективність у задачах підтримки ухвалення торгових рішень [15].

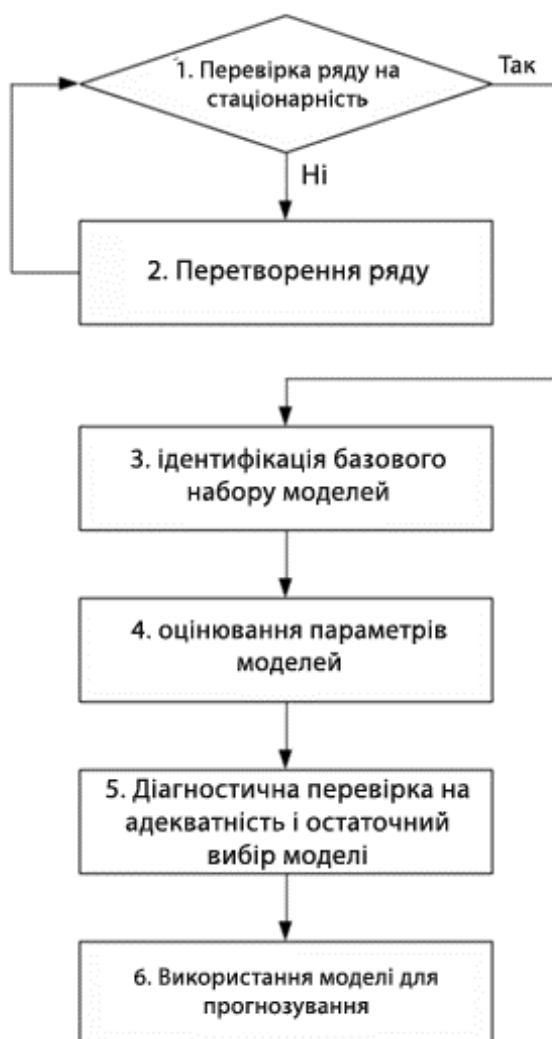


Рисунок 2.2 – Алгоритм підбору моделі ARIMA

До ключових переваг ARIMA слід віднести відносну простоту інтерпретації, яка досягається за рахунок обмеженої кількості параметрів, що формують структуру моделі та забезпечують прозорість отриманих результатів. Висока результативність проявляється насамперед у роботі зі стаціонарними часовими рядами, для яких характерні сталі статистичні характеристики, зокрема середнє значення та дисперсія. Додатковою перевагою є можливість розширення моделі шляхом включення допоміжних компонентів для аналізу трендів і трансформації даних до стаціонарного вигляду. Аналітичний потенціал моделі посилюється завдяки використанню автокореляційних функцій, що дозволяють досліджувати

структуру залежностей у часовому ряді та відстежувати зміну його параметрів. Важливим аспектом також є здатність моделі враховувати механізми зворотного зв'язку, що сприяє її адаптації до оновлених даних у процесі функціонування інформаційної системи. У прикладному вимірі це забезпечує достатній рівень точності прогнозування в межах коротко- та середньострокових часових горизонтів [16].

Разом із тим, використання ARIMA супроводжується рядом обмежень, які необхідно враховувати при розробленні інформаційної системи прогнозування фінансових часових рядів для трейдингу. Зокрема, модель демонструє обмежену ефективність у випадку нелінійних процесів, особливо коли часові ряди містять складні патерни або виражені сезонні компоненти. Суттєвою проблемою є залежність від стаціонарності даних, що зумовлює необхідність їх попередньої обробки, яка може бути трудомісткою та вимагати значних часових витрат. У випадку довготривалих трендів процес диференціювання може виконуватися багаторазово, що підвищує обчислювальне навантаження та потенційно призводить до втрати частини інформативних характеристик ряду. Додаткові труднощі виникають у процесі оптимізації параметрів моделі з використанням критеріїв якості, таких як AIC або BIC, особливо для складних і зашумлених даних. Обмежена адаптивність до динамічних змін ринкового середовища також виступає стримуючим фактором, оскільки модель має відносно статичну природу та потребує достатнього обсягу даних для ефективного налаштування. Високий рівень шумів у фінансових часових рядах додатково ускладнює процес ідентифікації параметрів і може негативно впливати на точність прогнозних оцінок.

Поряд з іншими підходами до розв'язання прикладних задач у системах прогнозування фінансових часових рядів для трейдингу широко застосовуються моделі умовної гетероскедастичності, зокрема узагальнена авторегресійна модель GARCH. Даний клас моделей орієнтований на аналіз і прогнозування волатильності, що робить його особливо корисним при дослідженні фінансових даних, включаючи оцінювання дохідності активів, прогнозування валютних курсів

та аналіз макроекономічних індикаторів. Концептуально GARCH спрямована на моделювання змін дисперсії у часі як випадкового процесу, що забезпечує більш глибоке розуміння ризиків і невизначеності на фінансових ринках [17].

Функціональні можливості цієї моделі полягають у здатності відтворювати майбутню динаміку волатильності на основі історичних спостережень, що дозволяє враховувати ефект кластеризації волатильності, характерний для фінансових часових рядів. Практичне застосування GARCH також пов'язане з оцінюванням і контролем ризиків, зокрема через розрахунок показника Value-at-Risk, що широко використовується у фінансовій аналітиці. Окрім цього, модель може виступати допоміжним інструментом у задачах оптимізації інвестиційних стратегій, забезпечуючи більш точну оцінку очікуваних коливань ринку. Її застосування є доцільним і в задачах оцінювання фінансових деривативів, зокрема при використанні аналітичних моделей на кшталт підходу Блека-Шоулза.

Процес побудови та використання моделі GARCH у складі інформаційної системи прогнозування передбачає послідовне виконання кількох взаємопов'язаних етапів. На початковій стадії здійснюється збір статистичних даних щодо динаміки часових рядів, зокрема розрахунок дохідностей фінансових інструментів на відповідних часових інтервалах. Подальша обробка включає очищення даних від шумових компонентів, усунення аномальних значень та приведення ряду до стану, придатного для моделювання, що часто передбачає досягнення стаціонарності. Наступним кроком є вибір конкретної специфікації моделі $GARCH(p, q)$, де параметр p визначає порядок авторегресійної складової, а параметр q характеризує структуру умовної гетероскедастичності. Базове представлення моделі описує залежність поточної умовної дисперсії від попередніх значень квадратів залишків і власних попередніх значень, що дозволяє ефективно враховувати часову залежність волатильності в межах досліджуваного процесу [18].

Подальший етап застосування моделі GARCH у межах інформаційної системи прогнозування фінансових часових рядів передбачає оцінювання її параметрів із використанням методу максимальної правдоподібності, що дозволяє

отримати числові характеристики моделі та здійснити їх аналітичну інтерпретацію. Після цього виконується перевірка адекватності побудованої моделі шляхом зіставлення її поведінки з реальними даними, зокрема із застосуванням тестів на основі множників Лагранжа для виявлення залишкової гетероскедастичності, аналізу автокореляційних залежностей і контролю стаціонарності процесу. Наступним кроком є формування прогнозів умовної дисперсії, що відображає очікуваний рівень волатильності у майбутніх періодах. Отримані результати використовуються для подальшого аналізу ризиків, оцінювання можливих коливань ринку та обґрунтування інвестиційних рішень у межах підтримки торгової діяльності. Завершальною стадією виступає інтерпретація результатів із подальшим їх впровадженням у практичні сценарії функціонування системи.

Сучасний розвиток підходів до моделювання волатильності зумовив появу модифікованих варіантів GARCH, серед яких особливе місце займають A-GARCH та E-GARCH, що забезпечують кращу відповідність емпіричним даним і підвищену адаптивність до змін ринкового середовища. Зокрема, асиметричні специфікації дозволяють враховувати різну реакцію волатильності на позитивні та негативні шоки, що є характерним для фінансових ринків, де періоди зростання та спаду супроводжуються різною інтенсивністю коливань [19].

До переваг моделей GARCH належить здатність адекватно відтворювати динаміку волатильності активів за рахунок урахування часової мінливості процесів. Важливою характеристикою є можливість прогнозування ризиків, що виникають унаслідок змін рівня волатильності, що безпосередньо пов'язано з прийняттям ефективних торгових рішень. Структурна організація моделі є відносно простою для інтерпретації, що полегшує її інтеграцію у прикладні інформаційні системи. Додатковою перевагою виступає адаптивність до різних типів даних та можливість розширення базової специфікації для врахування асиметричних ефектів. Використання методу максимальної правдоподібності для оцінювання параметрів забезпечує достатній рівень точності, а наявність розвинених засобів діагностики дозволяє здійснювати кількісну оцінку волатильності. Також модель добре поєднується з іншими економетричними

підходами, зокрема з VAR або ARIMA, що розширює можливості комплексного аналізу часових рядів.

Водночас застосування GARCH супроводжується низкою обмежень. Зокрема, процес налаштування параметрів може бути складним і вимагати значних обчислювальних ресурсів, особливо у випадку складних структур даних. Реалізація багатовимірних модифікацій, таких як MGARCH, пов'язана з додатковими технічними труднощами. Припущення щодо нормальності розподілу похибок не завжди відповідає реальним фінансовим даним, які часто характеризуються наявністю важких хвостів. Базова модель має лінійний характер, що обмежує її застосовність для опису складних нелінійних залежностей. Недостатнє врахування асиметричних ефектів також може знижувати точність оцінювання, тоді як використання розширених моделей потребує більших ресурсів і ускладнює інтерпретацію результатів. Додаткові труднощі пов'язані з вибором порядку моделі, що безпосередньо впливає на якість прогнозування, а також із високою чутливістю до процедур діагностики автокореляції. При цьому точність прогнозів має тенденцію до зниження на довгострокових часових інтервалах.

Окремим напрямом розвитку моделей, що використовуються у задачах підтримки ухвалення торгових рішень, є застосування нейронних мереж, зокрема архітектур типу LSTM, які демонструють високу ефективність у прогнозуванні часових рядів завдяки здатності враховувати довготривалі залежності у даних. LSTM належать до класу рекурентних нейронних мереж, орієнтованих на обробку послідовностей, що робить їх придатними для аналізу фінансових часових рядів та інших структурованих у часі даних. Їх ключова перевага полягає у здатності зберігати інформацію протягом тривалих інтервалів часу, що дозволяє уникнути проблеми згасання градієнта, характерної для класичних рекурентних моделей.

Архітектурно LSTM можна представити як послідовність взаємопов'язаних блоків, кожен з яких містить кілька функціональних компонентів, що забезпечують керування потоками інформації. На початковому етапі здійснюється фільтрація надлишкових даних за допомогою механізму «забування», реалізованого через сигмоїдну функцію активації. Далі відбувається відбір нової інформації, що

підлягає збереженню, та оновлення стану комірки відповідно до актуальних значень вхідних даних [20]. Завершальна стадія пов'язана з формуванням вихідного сигналу, який визначається на основі відфільтрованих і трансформованих даних. Схематичне представлення структури LSTM-моделі наведено на рис. 2.3.

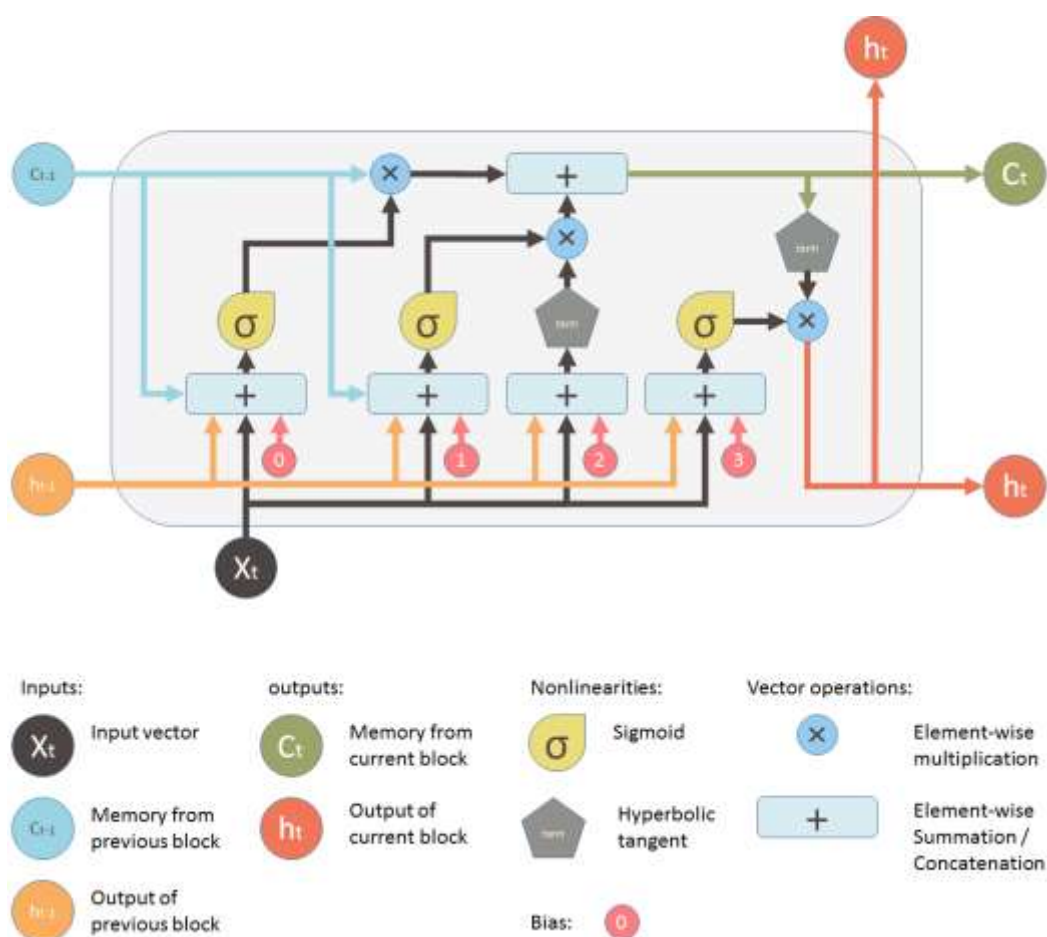


Рисунок 2.3 - Схема складу LSTM моделі ШНМ

З позиції структурної організації модель LSTM формується як сукупність спеціалізованих функціональних блоків, кожен із яких виконує визначену роль у процесі обробки послідовних даних. Центральним елементом виступає комірка стану, яка забезпечує передавання інформації крізь усю часову послідовність із мінімальними змінами, що дозволяє зберігати релевантні залежності на значних часових інтервалах. Такий механізм фактично реалізує безперервний інформаційний потік, що підтримує цілісність накопичених знань моделі.

Важливою складовою є також затвор забування, який визначає, які частини інформації втрачають актуальність і мають бути вилучені зі стану комірки. Затвор введення виконує функцію селекції нових даних, що підлягають інтеграції до внутрішнього стану моделі, тоді як затвор виведення відповідає за формування вихідного сигналу на основі актуалізованої інформації.

До ключових переваг LSTM належить здатність ефективно утримувати інформацію протягом тривалих часових інтервалів, що є критично важливим для аналізу фінансових часових рядів. Завдяки використанню механізмів керування потоками даних через систему затворів модель демонструє високу ефективність як на коротких, так і на довгих послідовностях. Це забезпечує її універсальність і практичну доцільність у задачах прогнозування, зокрема у межах інформаційних систем підтримки ухвалення торгових рішень.

Разом із тим, застосування LSTM пов'язане з певними обмеженнями. Зокрема, модель характеризується значними обчислювальними витратами та потребує тривалого часу для навчання і тестування, що ускладнює її використання в умовах обмежених ресурсів. Додатковою складністю є процес налаштування архітектури та підбору гіперпараметрів, який часто має експериментальний характер і вимагає значного досвіду. Крім того, для досягнення високої якості прогнозування необхідні великі обсяги навчальних даних, що не завжди доступні в реальних умовах. Таким чином, поряд із класичними методами машинного навчання існує широкий спектр альтернативних підходів до моделювання фінансових часових рядів. Їх комбінування, порівняльний аналіз та інтеграція в єдині гібридні рішення відкривають перспективи для підвищення точності прогнозування та узагальнювальної здатності моделей [21].

2.2 Опис обраного набору даних для аналізу

З урахуванням особливостей предметної області та вимог до побудови інформаційної системи прогнозування фінансових часових рядів для трейдингу, обґрунтованим є використання платформи Yahoo Finance як основного джерела

даних [22]. Зазначений ресурс забезпечує надійний доступ до актуальної інформації про стан фінансових ринків і надає розвинений інструментарій для аналітичної обробки даних, прогнозування динаміки цін, роботи з фінансовою звітністю, а також врахування макроекономічних подій і новинного фону.

У задачах проптрейдингу функціональність платформи орієнтована на формування якісної інформаційної бази для прийняття рішень. Зокрема, вона забезпечує отримання історичних даних, що є необхідними для тестування та оптимізації торгових стратегій, виявлення прихованих закономірностей і побудови статистичних та машинних моделей. Додатково реалізується можливість аналізу поточного стану ринку на основі фінансової звітності компаній, включаючи показники доходів, прибутків, дивідендних виплат і ключових мультиплікаторів, що дозволяє здійснювати оцінювання фундаментальної вартості активів. Важливим компонентом є також інтеграція новинного аналізу, який враховує вплив зовнішніх факторів на динаміку ринку.

Функціональні характеристики платформи включають підтримку різних часових інтервалів представлення даних, починаючи від високочастотних значень із мінімальним кроком і завершуючи агрегованими показниками на довгих інтервалах. Система надає широкий спектр фінансової інформації, включаючи звіти про прибутки, рух грошових коштів і балансові показники, а також ключові коефіцієнти, зокрема P/E, P/B і рівень дивідендної дохідності. Окрему цінність становить доступ до даних щодо дивідендних виплат і сплітів акцій, що є необхідним для коректного аналізу довгострокових інвестиційних стратегій. Крім того, платформа дозволяє отримувати інформацію про опціонні ланцюжки, що розширює можливості оцінювання волатильності та формування складніших торгових стратегій. Ілюстративний приклад інтерфейсу представлення даних у межах платформи наведено на рис. 2.4, що відображає структуровану форму подання фінансової інформації, придатної для подальшої інтеграції в модулі аналізу та прогнозування інформаційної системи.

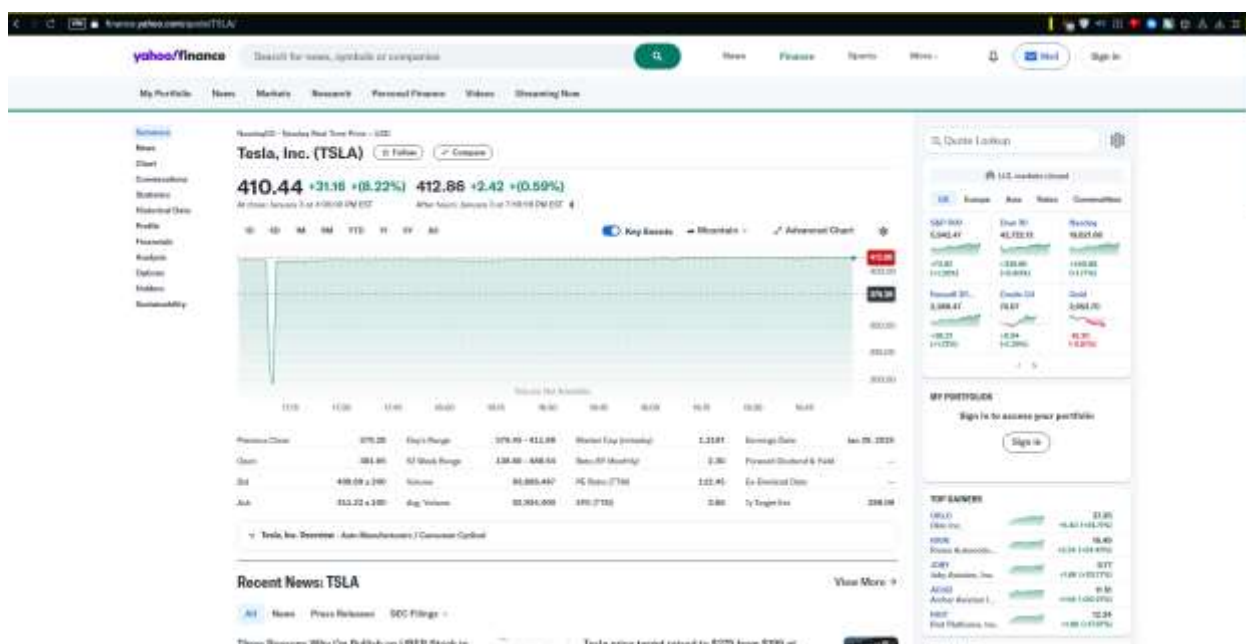


Рисунок 2.4 – Інтерфейс YF

Доцільність використання платформи Yahoo Finance у межах інформаційної системи прогнозування фінансових часових рядів для трейдингу також підтверджується її функціональними перевагами. Насамперед варто відзначити відкритий доступ до значного обсягу фінансових даних, що забезпечує можливість формування повноцінної інформаційної бази без додаткових витрат. Ресурс надає широкий спектр показників, включаючи історичні часові ряди значної тривалості, що є критично важливим для побудови та навчання моделей машинного навчання. Додатковою перевагою виступає зручність інтеграції у програмні рішення, що дозволяє ефективно використовувати ці дані в автоматизованих системах аналізу та підтримки ухвалення торгових рішень.

Разом із тим, використання цієї платформи супроводжується певними обмеженнями, які необхідно враховувати при проектуванні системи. Зокрема, безкоштовний доступ до даних передбачає їх оновлення із затримкою, що знижує ефективність застосування у задачах високочастотного трейдингу. Також можливі неточності у даних, особливо для менш ліквідних або маловідомих фінансових інструментів, що може впливати на якість побудованих моделей. Додатковим

обмеженням є наявність лімітів API, які ускладнюють масове завантаження великих обсягів інформації.

Для інтеграції даних у програмну частину системи доцільно застосовувати бібліотеку `yfinance`, яка забезпечує зручний інтерфейс доступу до ресурсів Yahoo Finance. Використання цього інструменту дозволяє автоматизувати процес отримання історичних і поточних ринкових даних, а також супутньої фінансової інформації. Зокрема, бібліотека підтримує завантаження котирувань фінансових інструментів за обрані періоди із деталізацією за показниками відкриття, максимуму, мінімуму, закриття, скоригованої ціни та обсягів торгів. Окрім цього, реалізовано доступ до даних, наближених до реального часу, за умови їх доступності, а також до фінансової звітності компаній, інформації про дивідендні виплати та спліти акцій. Додатково підтримується отримання метаданих щодо секторів і галузей, аналіз опціонних ланцюжків, а також робота з валютними курсами та біржовими фондами, що розширює можливості комплексного аналізу в межах інформаційної системи.

2.3 Розробка структури проєкту

Архітектура розроблюваної інформаційної системи прогнозування фінансових часових рядів для трейдингу передбачає наявність трьох функціонально відокремлених модулів аналізу даних, які у взаємодії забезпечують формування обґрунтованих торгових рішень. Перший модуль орієнтований на оцінювання та інтерпретацію торгових індикаторів і реалізує підхід, заснований на використанні ковзних середніх. У межах цього компонента генеруються сигнали купівлі та продажу шляхом аналізу перетину різних типів ковзних середніх, зокрема SMA та EMA, із додатковим урахуванням таких індикаторів, як RSI, MACD і смуги Боллінджера, що дозволяє підвищити точність і надійність сигналів.

Другий модуль призначений для аналітичної обробки часових рядів із використанням моделей ARIMA різних конфігурацій, включаючи базові, автоматизовано параметризовані та сезонні варіанти. Його функціонування

спрямоване на виявлення закономірностей у динаміці фінансових показників і формування коротко- та середньострокових прогнозів, що враховують трендові та сезонні компоненти.

Третій модуль реалізує підходи машинного навчання, орієнтовані на побудову прогностичних моделей для аналізу фінансових часових рядів. У його межах здійснюється навчання моделей на історичних даних з метою передбачення майбутньої динаміки цін активів на заданому часовому горизонті, що розширює можливості системи у задачах підтримки прийняття рішень.

Схематичне представлення взаємодії зазначених модулів наведено на рис. 2.5. Формування вхідних даних для системи здійснюється через програмний інтерфейс доступу до платформи Yahoo Finance, що забезпечує вибірку необхідної інформації відповідно до заданих параметрів, зокрема часових інтервалів та типів фінансових інструментів. Отримані дані зберігаються у структурованому вигляді в окремих колекціях, після чого підлягають попередній обробці, яка включає аналіз трендів, розрахунок значень цін закриття за визначені періоди, а також урахування подій, пов'язаних зі сплітами акцій і змінами капіталізації.

На основі сформованого масиву даних здійснюється первинний аналітичний етап, у межах якого створюються відповідні об'єкти та моделі із застосуванням різних методів обробки. Застосування ковзних середніх дозволяє зменшити вплив випадкових коливань і виявити домінуючі тренди для різних часових інтервалів, що сприяє визначенню потенційних точок входу та виходу з позицій. Використання моделей ARIMA забезпечує ефективний аналіз часових рядів із вираженими сезонними компонентами та дозволяє враховувати трендові зміни у процесі прогнозування вартості активів і обсягів торгів. Додатково залучаються методи глибокого навчання, які підвищують здатність системи до моделювання складних нелінійних залежностей і формування альтернативних сценаріїв розвитку ринкової ситуації, що є важливим для підвищення якості прийняття торгових рішень.

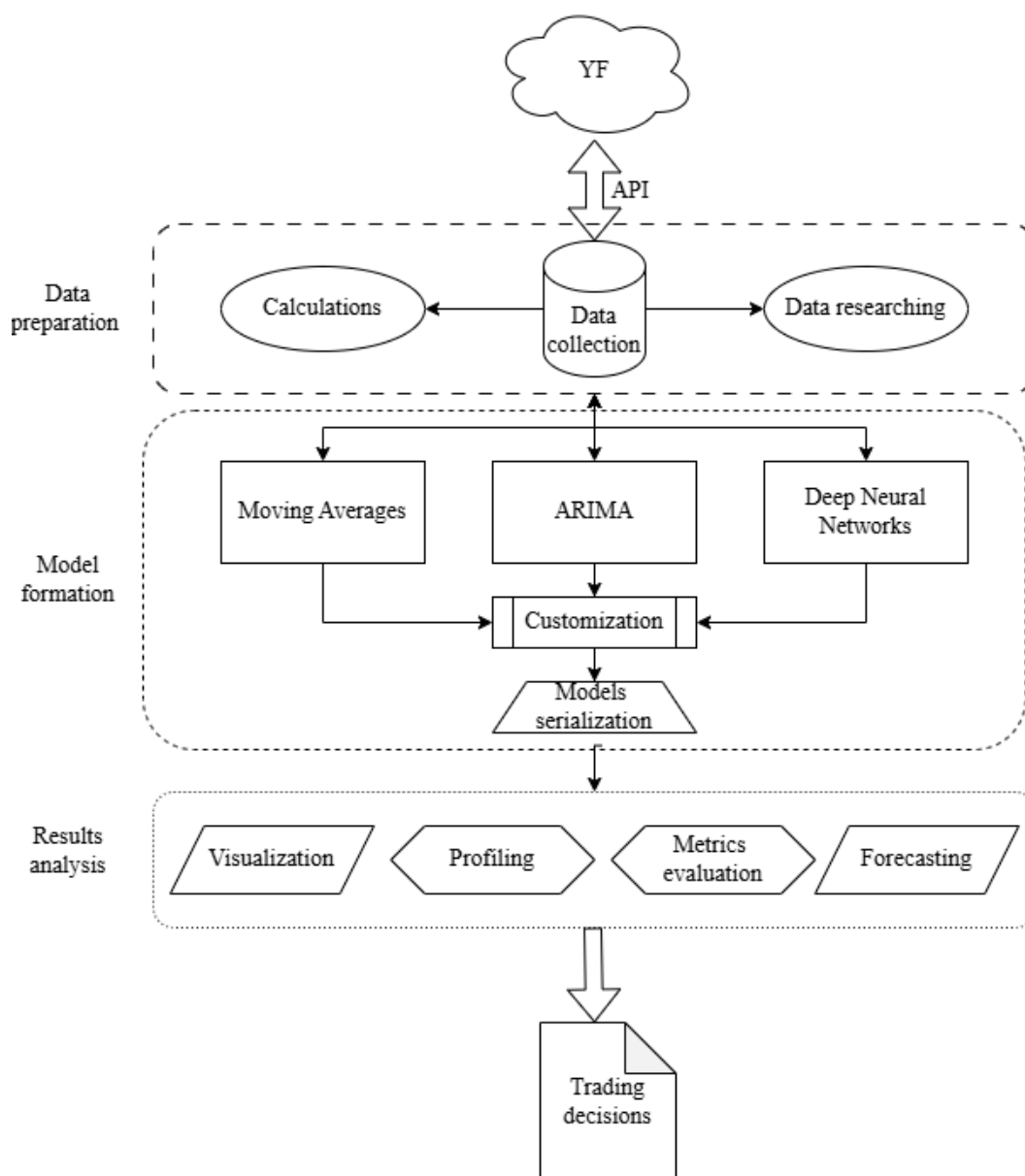


Рисунок 2.5 – Схематична структура взаємодії модулів системи

Після етапу побудови моделі виконуються процедури її параметризації відповідно до обраних часових інтервалів та конфігурацій вхідних змінних, що безпосередньо впливають на якість прогнозування. Надалі сформовані моделі можуть бути збережені як автономні програмні об'єкти із застосуванням форматів серіалізації, зокрема `pickle` або `keras`, що забезпечує їх подальше повторне використання без прив'язки до середовища розробки.

Збережені моделі можуть бути завантажені для виконання аналітичних процедур у різних програмних середовищах, що підвищує гнучкість і масштабованість інформаційної системи прогнозування фінансових часових рядів

для трейдингу. Наступним кроком є комплексна оцінка результатів функціонування моделей, яка передбачає візуальне представлення даних у вигляді двовимірних графіків для виокремлення характерних закономірностей, аналіз ефективності процесів тестування, розрахунок метрик якості та виконання прогнозних обчислень. Отримані результати використовуються як основа для підтримки ухвалення торгових рішень у межах стратегій типу купівля, продаж або утримання позиції.

Оцінювання простого ковзного середнього (2.1) базується на визначенні середнього арифметичного значень цін за заданий часовий інтервал

$$SMA = \frac{\sum_{i=1}^n P_i}{n}, \quad (2.1)$$

де P_i інтерпретується як ціна закриття, а n характеризує кількість періодів спостереження.

Експоненційне ковзне середнє (2.2), на відміну від простого, надає більшу вагу останнім спостереженням, що дозволяє оперативніше реагувати на зміни ринкової динаміки за рахунок використання коефіцієнта згладжування α .

$$EMA_t = \alpha \cdot P_t + (1 - \alpha) \cdot EMA_{t-1}. \quad (2.2)$$

У межах задачі ідентифікації ринкових трендів здійснюється класифікація динаміки цін активів залежно від їх співвідношення з ковзними середніми. Зростаюча тенденція фіксується у випадку перевищення поточних значень над середнім рівнем, тоді як спадна визначається за протилежної умови. Перетин цінового графіка з ковзним середнім у напрямку знизу вгору інтерпретується як потенційний сигнал до відкриття довгої позиції, тоді як зворотний перетин може свідчити про доцільність продажу активу з урахуванням прогнозного горизонту, що відповідає відомим сценаріям типу Golden Cross або Death Cross.

Моделі класу ARIMA з параметрами (p, d, q) , де p визначає порядок авторегресійної складової, d відповідає за кількість операцій диференціювання для досягнення стаціонарності, а q характеризує порядок ковзного середнього, інтегруються в систему для формування прогнозів майбутніх значень фінансових показників на основі статистичних вибірок. Використання такого підходу дозволяє враховувати ключові ринкові рівні, зокрема зони підтримки та опору, а також підвищує обґрунтованість визначення моментів входу та виходу з торгових позицій.

Для узагальненого відображення логіки функціонування системи застосовано моделювання засобами UML. Діаграма варіантів використання для модуля аналізу торгових індикаторів наведена на рис. 2.6. У межах цього функціонального компонента користувач отримує можливість виконувати аналіз завантажених даних, здійснювати обчислення значень індикаторів, зокрема SMA, EMA, RSI та MACD, інтерпретувати результати у графічному вигляді, а також зберігати сформовані аналітичні висновки і візуалізації.

Функціональність модуля побудови моделей ARIMA є подібною за загальною логікою, проте розширена додатковими можливостями. Користувач задає тікер фінансового інструменту, часовий діапазон і параметри моделі, після чого ініціює процес її формування. Передбачено інструменти для оцінювання точності за допомогою метрики RMSE, виконання прогнозування, відображення аналітичних показників і рекомендацій у межах інтерфейсу системи. Окрім цього, реалізовано механізми серіалізації моделей, результатів аналізу та графічних матеріалів у відповідні файлові формати. Відповідна діаграма варіантів використання представлена на рис. 2.7. Аналогічно, для модуля побудови моделей глибокого навчання сформовано окрему діаграму варіантів використання, що наведена на рис. 2.8.

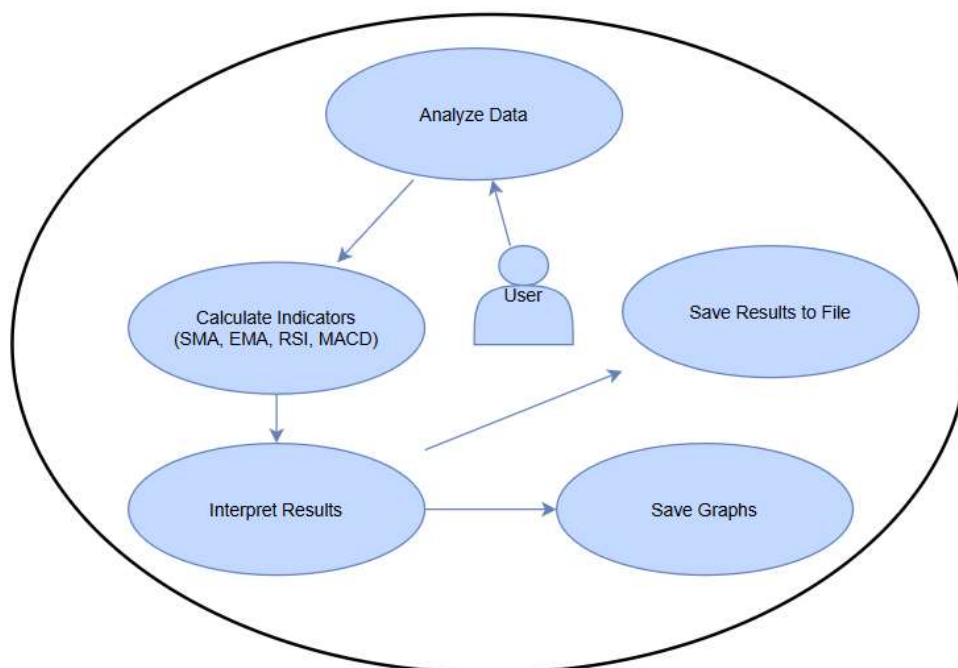


Рисунок 2.6 – Use case модуля аналізу торгових індикаторів

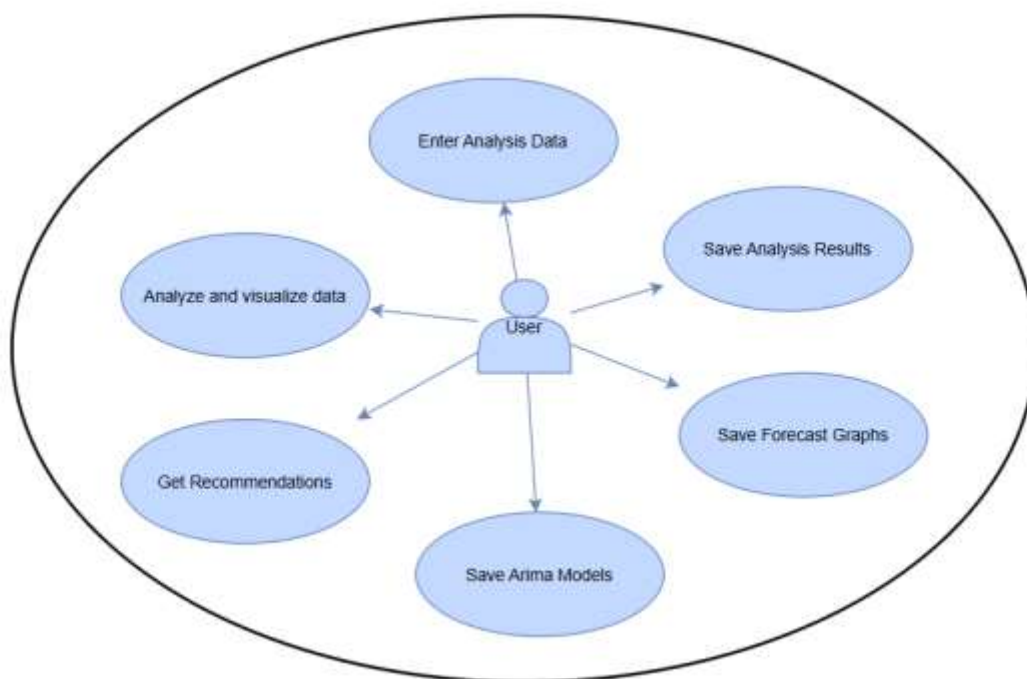


Рисунок 2.7 – Use case модуля аналізу даних на основі моделей ARIMA

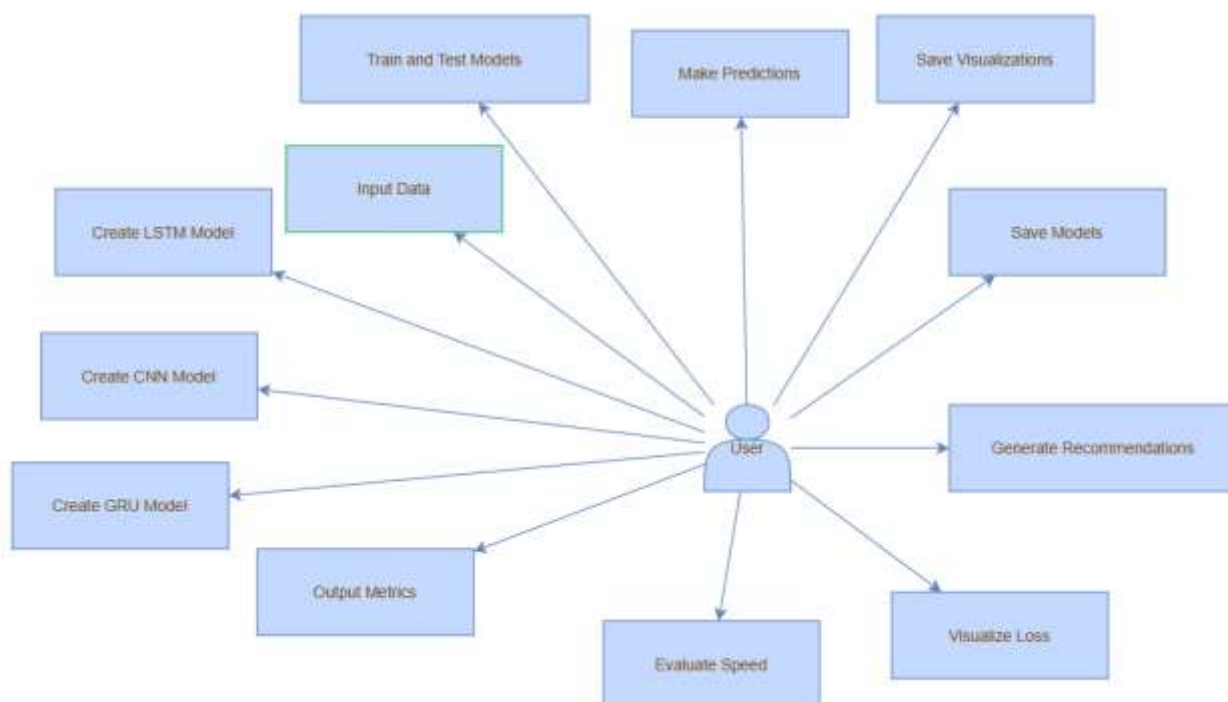


Рисунок 2.8 – Use case модуля побудови моделей ГН

У межах розроблюваної інформаційної системи користувачу надається розширений набір функціональних можливостей, спрямованих на організацію повного циклу роботи з фінансовими часовими рядами. Зокрема, передбачено введення початкових даних, формування навчальної та тестової вибірок, побудову моделей машинного навчання на основі архітектур LSTM, CNN та GRU, а також виконання процедур їх навчання і подальшої валідації [23-25]. Крім того, реалізовано механізми відображення метрик точності, візуалізації значень функції втрат, аналізу швидкодії процесу побудови моделей, здійснення прогнозування та генерації рекомендацій для підтримки ухвалення торгових рішень. Результати обробки можуть бути збережені у вигляді графічних матеріалів, а самі моделі — у форматі keras для подальшого використання.

Архітектура реалізована у вигляді сукупності взаємопов'язаних функціональних блоків, що інкапсулюють окремі аспекти обробки даних і побудови моделей у середовищі програмування Python.

Компонент GUI відповідає за організацію користувацького інтерфейсу та забезпечує взаємодію з системою через набір візуальних елементів керування. У

його межах реалізовано обробку подій, пов'язаних із діями користувача, зокрема натисканням кнопок, введенням даних у текстові поля та заповненням табличних структур.

Модуль `DataProcessing` виконує функції завантаження, попередньої обробки та трансформації даних, необхідних для подальшого використання у процесах навчання та тестування моделей. Його роль полягає у забезпеченні коректності та узгодженості вхідних даних, що є критично важливим для досягнення високої точності прогнозування. Компонент `Model` реалізує побудову моделей глибокого навчання, зокрема із застосуванням архітектур LSTM, CNN і GRU, що дозволяє враховувати як часові залежності, так і складні нелінійні взаємозв'язки у фінансових даних. Модуль `MetricsEvaluation` забезпечує обчислення показників якості функціонування моделей і виконує їх порівняльний аналіз, що дозволяє обрати найбільш ефективну конфігурацію для конкретної задачі прогнозування.

Компонент `Prediction` відповідає за запуск процесів прогнозування та формування вихідних результатів на основі навченої моделі, що безпосередньо використовується для підтримки ухвалення торгових рішень.

Модуль `Visualization` реалізує побудову графічних представлень результатів, включаючи динаміку змін показників і значення метрик, що сприяє більш глибокому аналізу ефективності моделей.

Компонент `FileOperations` забезпечує виконання операцій серіалізації, збереження та завантаження моделей машинного навчання і результатів їх функціонування, що підвищує відтворюваність експериментів і зручність подальшого використання системи.

Відповідна діаграма компонентів системи наведена на рис.2.9, що ілюструє взаємозв'язки між зазначеними функціональними блоками та їх роль у загальній архітектурі інформаційної системи прогнозування фінансових часових рядів для задач трейдингу.

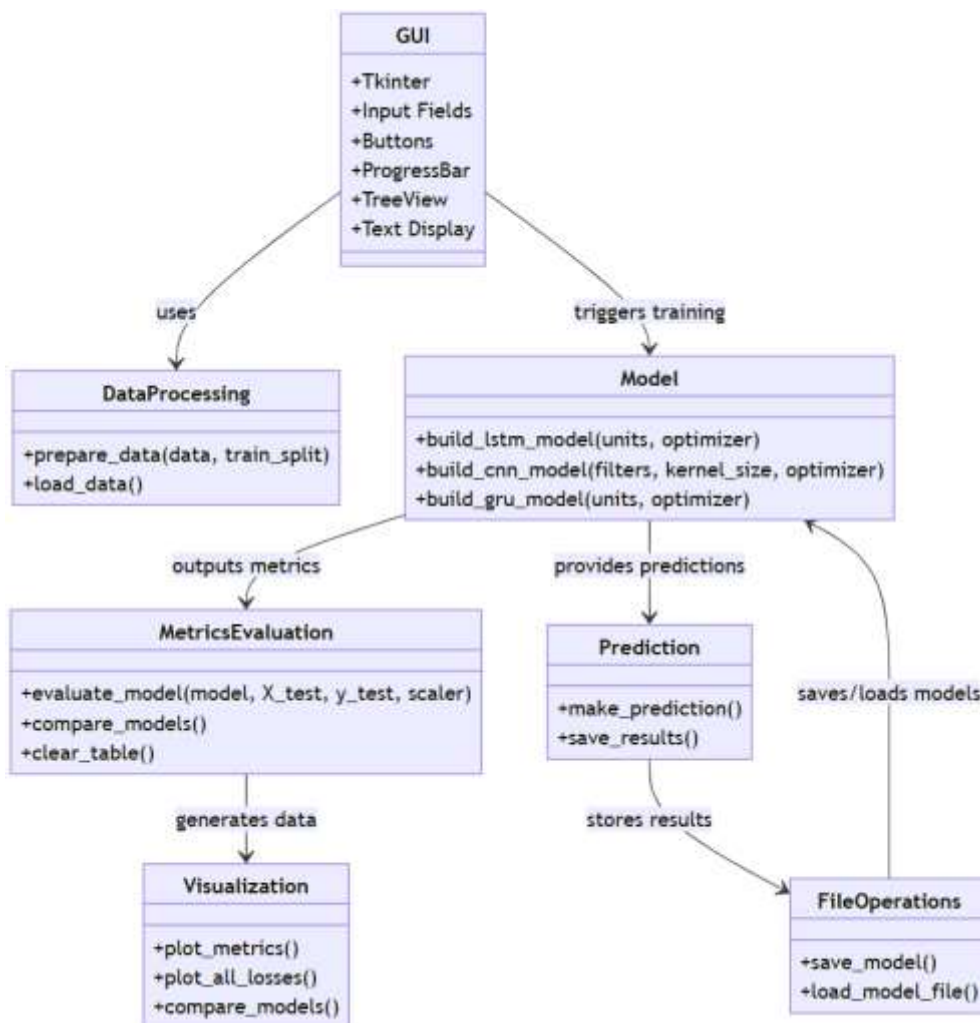


Рисунок 2.9 – Діаграма загальних компонентів системи

Для демонстрації процесу обробки даних та взаємодії між ключовими компонентами під час роботи системи створено діаграму послідовності модуля оцінки та аналізу торгових індикаторів, наведеної на рис. 2.10. Після запуску програми ініціюється створення екземпляра класу Tkinter, який відповідає за формування графічного інтерфейсу користувача. Далі здійснюється послідовний виклик функцій системи, що включає отримання необхідних даних через зовнішнє API, їх попередню обробку та обчислення торгових індикаторів. Після цього виконуються операції збереження результатів та побудови візуальних представлень даних у вигляді графіків, які відображаються безпосередньо в межах інтерфейсу користувача. Такий підхід дозволяє забезпечити інтегрований аналіз

ринкових показників і зручну взаємодію з користувачем під час оцінки сигналів купівлі та продажу.

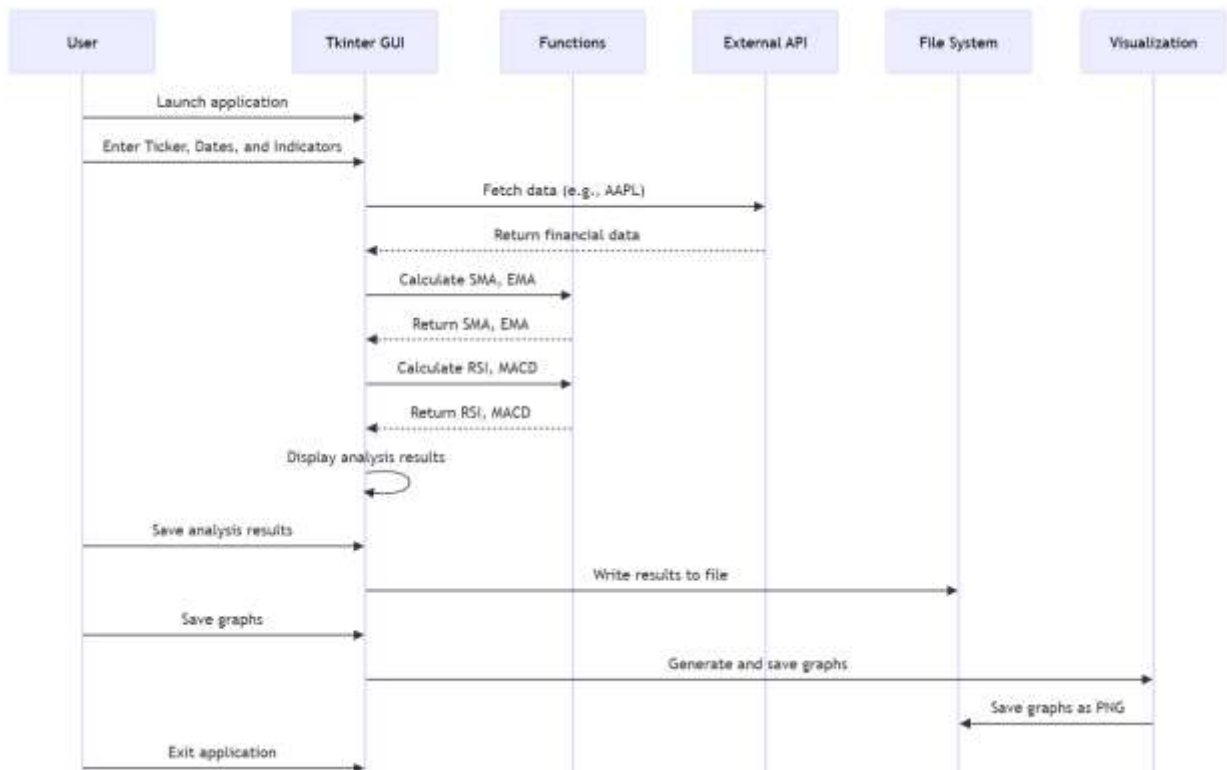


Рисунок 2.10 – Діаграма послідовності дій модуля торгових індикаторів

Діаграма послідовності роботи модуля аналізу даних на основі моделей ARIMA наведена на рис. 2.11. Вона повторює загальну логіку попереднього модуля, проте містить додаткові кроки, пов'язані зі створенням моделей ARIMA, автоматизованим підбором гіперпараметрів для моделі з автоналаштуванням, а також з операціями збереження та завантаження моделей у форматі pickle через компонент FileSystem.

Діаграма послідовності модуля створення моделей машинного навчання представлена на рис. 2.12. Процес формування моделей відбувається послідовно: спочатку користувач обирає потрібну архітектуру з переліку підтримуваних типів (LSTM, CNN, GRU), після чого задає гіперпараметри та виконує процедури навчання і тестування моделей. Логи, що містять інформацію про помилки та результати навчання, виводяться у консоль для налагодження та контролю процесу

в середовищі розробки. Такий підхід забезпечує прозору взаємодію користувача з системою та контроль точності виконання кожного етапу моделювання.

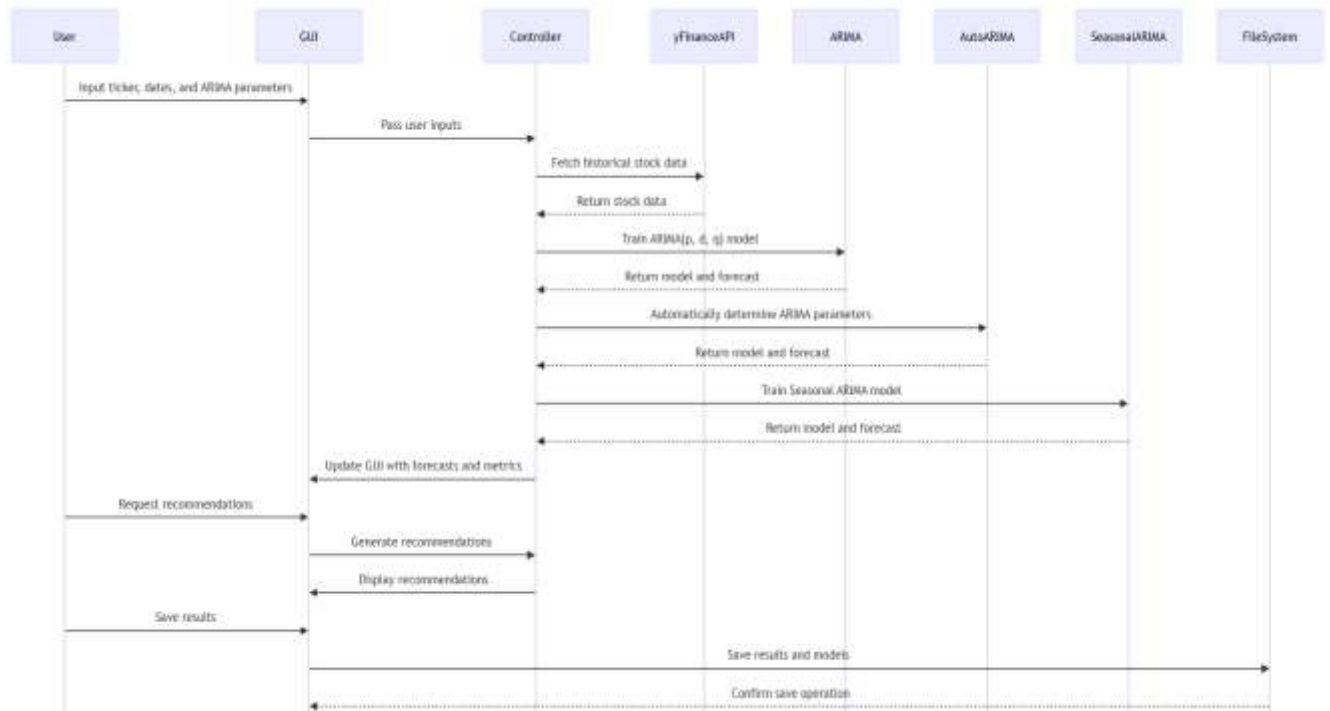


Рисунок 2.11 – Діаграма послідовності операцій модуля ARIMA

Структура проєкту в середовищі розробки наведена на рис.2.13. Каталог `ready_models` призначений для зберігання серіалізованих файлів об'єктів навчених моделей машинного навчання. У папці `results` зберігаються текстові документи з оцінками результатів роботи системи, включаючи аналітику торгових рішень, прогнозування ринкових трендів та рекомендації щодо застосування торгових стратегій. Каталоги `visualization`, `visualization_arima` та `visualization_dnn` використовуються для архівування графічних матеріалів, створених у процесі роботи відповідних модулів, у форматі PNG. Головний каталог проєкту містить програмні файли модулів, серед яких `signals_analysis`, `model_arima` та `model_dnn`, що реалізують відповідну функціональність для обробки даних, побудови моделей та візуалізації результатів. Така структура забезпечує зручне зберігання, доступ і повторне використання компонентів системи.

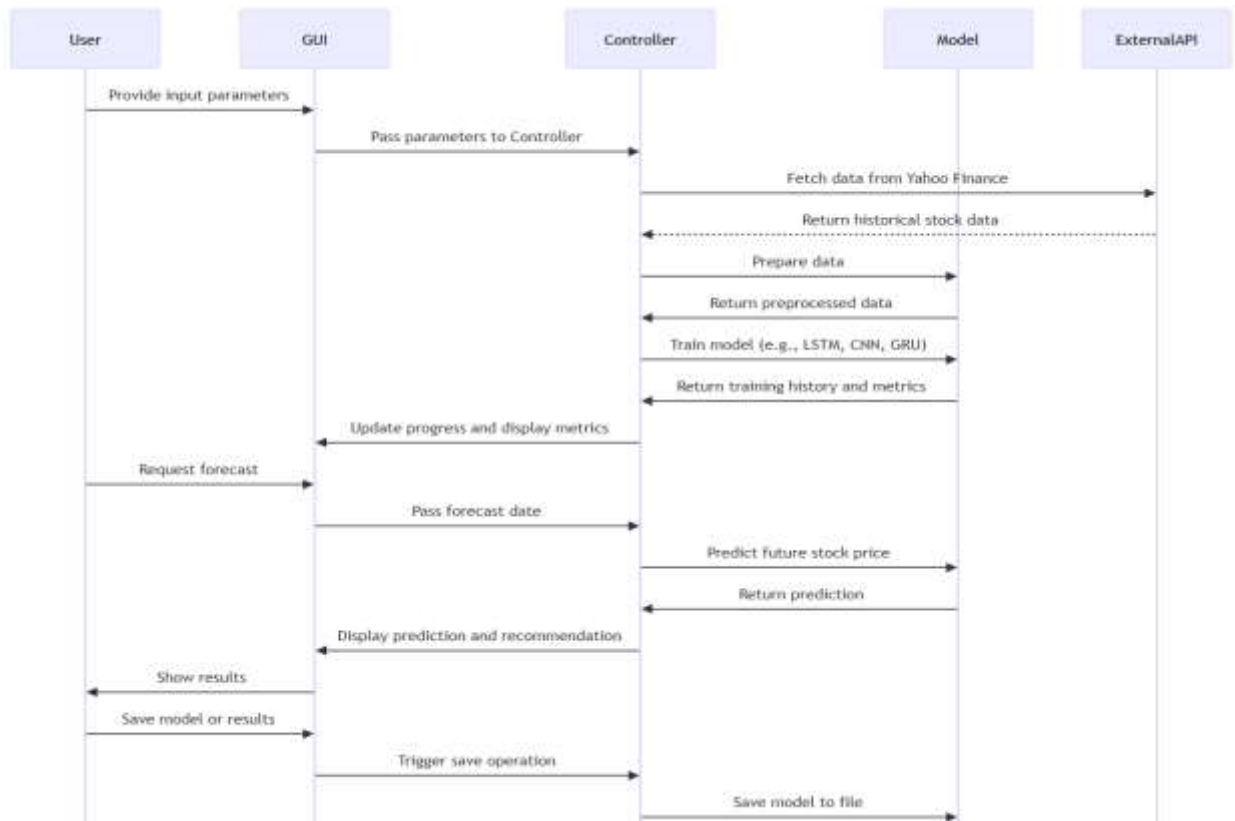


Рисунок 2.12 – Діаграма послідовності операцій модуля створення моделей машинного навчання

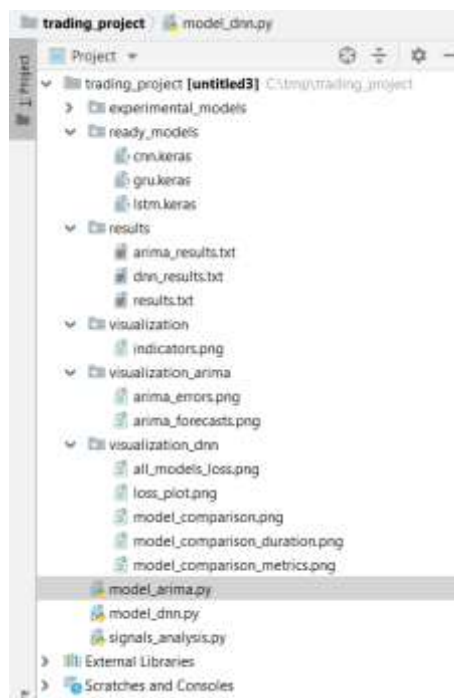


Рисунок 2.13 – Структура проєкту в середовищі розробки

Висновки. Проведений аналіз показав ефективність використання класичних та сучасних методів прогнозування фінансових часових рядів для підтримки трейдингових рішень. Досліджено особливості моделей ARIMA, GARCH та LSTM, їх переваги й обмеження, що дозволяє вибирати оптимальні підходи для різних типів даних і часових горизонтів. Розроблений проект системи інтегрує модулі аналізу торгових індикаторів, статистичного прогнозування та глибинного навчання, забезпечуючи комплексну оцінку ринку. Структуроване зберігання даних, моделей і візуалізацій підвищує повторне використання результатів і полегшує аналітичні процеси. Використання платформи Yahoo Finance та бібліотек Python дозволяє автоматизувати збір і обробку даних, що підвищує точність прогнозів і підтримує прийняття обґрунтованих торгових рішень.

3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ ДЛЯ ТРЕЙДИНГУ

3.1 Розробка програмної реалізації та інтерфейсу модулів системи

У межах реалізації інформаційної системи прогнозування фінансових часових рядів для задач трейдингу виконано розробку програмного модуля з описом його функціональних можливостей та користувацького інтерфейсу. Програмний продукт реалізовано у форматі настільного застосунку із застосуванням графічних компонентів бібліотеки Tkinter, що забезпечує інтерактивну взаємодію користувача з аналітичними інструментами.

Ядро бізнес-логіки підсистеми аналізу технічних індикаторів орієнтоване на оброблення часових рядів фінансових даних та формування відповідних аналітичних оцінок. Зокрема, функціональна процедура `calculate_sma(data, window)` забезпечує визначення простого ковзного середнього для заданого інтервалу спостереження шляхом застосування механізму ковзного агрегування `rolling(window=window).mean()`, у результаті чого формується новий часовий ряд, кожен елемент якого відображає усереднене значення цін за попередній період довжиною n .

Функція `calculate_ema(data, window)` реалізує обчислення експоненційного ковзного середнього із використанням методу `ewm(span=window, adjust=False).mean()`, що дозволяє надавати більшої ваги останнім спостереженням, підвищуючи чутливість індикатора до короткострокових змін ринку.

У свою чергу, процедура `calculate_rsi(data, window=14)` виконує розрахунок індексу відносної сили, який застосовується для ідентифікації станів перекупленості або перепроданості фінансового інструмента. Алгоритм передбачає визначення різниці між послідовними значеннями цінового ряду, подальше розділення на позитивні прирости та негативні зміни, їх усереднення в межах заданого періоду та обчислення відношення середнього приросту до середнього збитку.

Функціональний модуль `calculate_macd(data, fast_period, slow_period, signal_period)` забезпечує побудову індикатора MACD як різниці між швидким та повільним експоненційними ковзними середніми. Додатково формується сигнальна лінія як згладжене значення MACD, а також гістограма, що відображає розбіжність між основною та сигнальною кривими, що є важливим для виявлення трендових змін.

Оцінювання результативності торгової стратегії здійснюється за допомогою функції `calculate_metrics(data, signal_column)`, яка базується на аналізі сформованих торгових сигналів. У межах цього підходу визначається позиція на кожному часовому кроці, обчислюється відносна зміна ціни активу, після чого розраховується дохідність стратегії як функція позиції та ринкової динаміки. Подальше агрегування дозволяє отримати накопичену дохідність, а також ключові показники ефективності, серед яких загальна дохідність, максимальна просадка, коефіцієнти Шарпа та Сортіно, що характеризують співвідношення ризику та прибутковості. Інтегральна процедура `analyze_data()` забезпечує комплексне виконання аналітичного процесу відповідно до параметрів, заданих користувачем, із подальшим відображенням результатів у графічному інтерфейсі та формуванням текстової інтерпретації значень RSI.

Функція `save_to_file()` реалізує механізм збереження результатів аналітичних розрахунків у текстовому файлі `signals_results.txt`, що дозволяє забезпечити подальше використання або документування отриманих даних.

Побудова візуального представлення результатів покладена на функцію `save_graphs()`, яка формує графіки цінового ряду разом із ковзними середніми, індикатором MACD, його сигнальною складовою та гістограмою, а також індексом RSI із позначенням критичних рівнів. У разі відсутності відповідної директорії `visualization` вона створюється автоматично, після чого результати зберігаються у графічному файлі `indicators.png`.

Структурну організацію програмного модуля аналізу індикаторів, включаючи ключові функціональні компоненти, змінні та структури даних для збереження результатів обчислень, наведено на рис. 3.1. Інтерфейс головного вікна

програмної реалізації підсистеми відображено на рис. 3.2. Він передбачає введення користувачем текстових і числових параметрів, зокрема ідентифікатора фінансового інструмента, часових меж вибірки даних із джерела YF, параметрів згладжування для ковзних середніх, а також конфігураційних значень для розрахунку індикатора MACD.

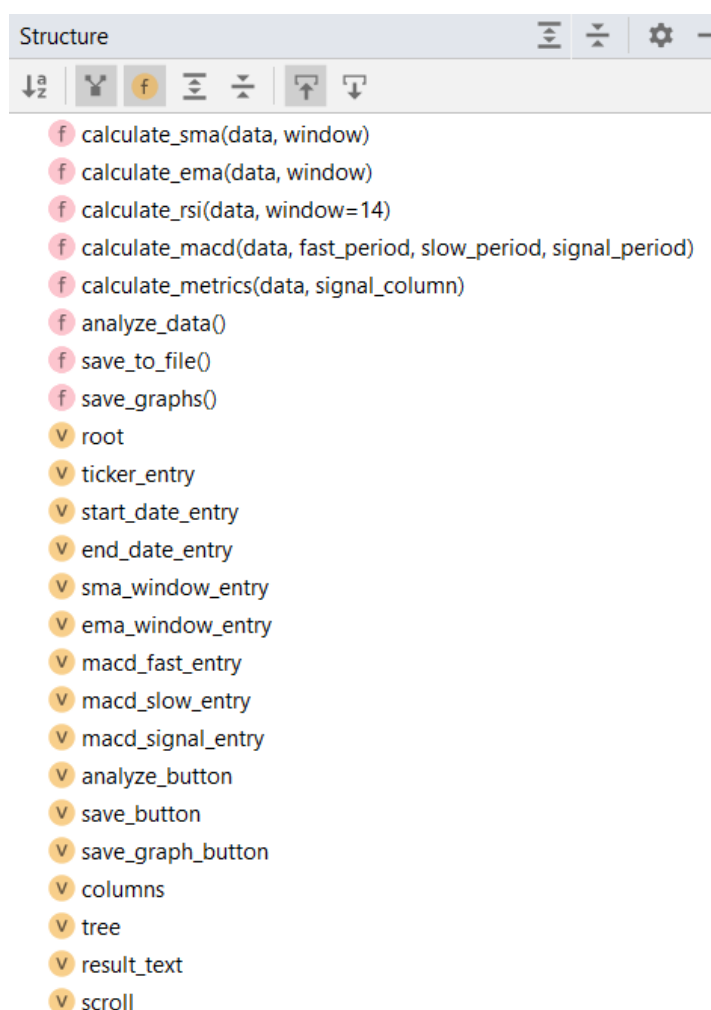


Рисунок 3.1 – Програмний склад модуля оцінки індикаторів

У структурі графічного інтерфейсу передбачено реалізацію трьох основних елементів керування, які забезпечують взаємодію користувача з функціоналом системи. Перша кнопка ініціює процес аналітичної обробки фінансових часових рядів після введення та валідації вхідних параметрів, друга відповідає за збереження отриманих результатів у файл, тоді як третя забезпечує експорт сформованих графічних візуалізацій індикаторів. З метою підвищення зручності

інтерпретації результатів у програмному застосунку реалізовано табличне подання даних, що містить п'ять інформативних стовпців. Узагальнені результати прогнозування, обчислені метрики ефективності, а також текстове пояснення сформованих торгових сигналів відображаються у спеціалізованій текстовій області, розташованій під таблицею. Для забезпечення коректної роботи з великими обсягами інформації дана область доповнена вертикальним механізмом прокручування. Візуальне представлення результатів аналізу фінансових даних, отриманих на основі оцінювання технічних індикаторів, наведено на рис. 3.3.

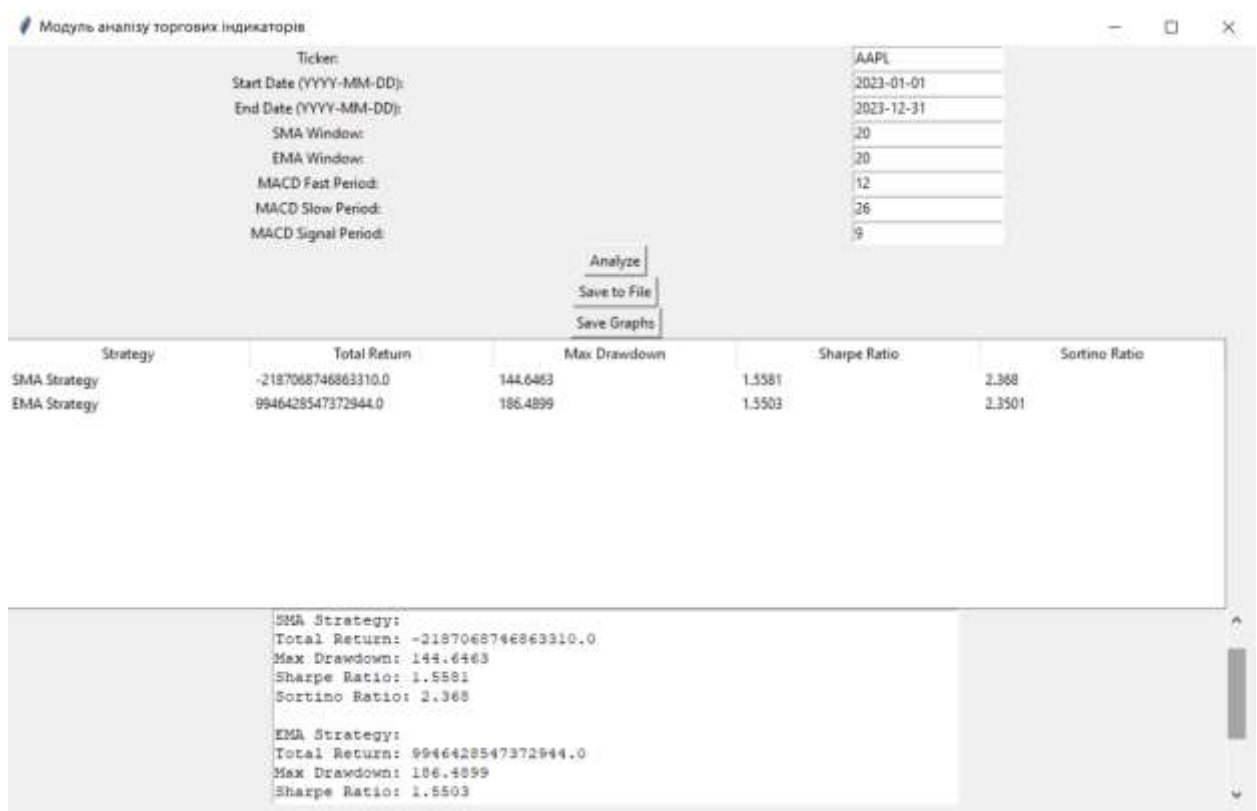


Рисунок 3.2– Інтерфейс користувача головної форми програмного модуля оцінки індикаторів

Порівняльний аналіз динаміки простого та експоненційного ковзних середніх свідчить про їх загальну узгодженість, однак у межах окремих часових інтервалів спостерігається перетин цінового ряду з відповідними кривими знизу вгору, що інтерпретується як потенційний сигнал до формування висхідного тренду. Водночас на завершальному етапі досліджуваного періоду фіксується зменшення

відстані між SMA та EMA, що може вказувати на ймовірну зміну поточного трендового напрямку.

Розбіжність між основною лінією MACD та її сигнальною складовою виступає індикатором інтенсивності ринкового сигналу. У переважній більшості зафіксованих випадків відбувається перетин MACD із сигнальною лінією у напрямку знизу вгору, що інтерпретується як передумова до відкриття довгої позиції. Додатково спостерігається тенденція до зростання амплітуди гістограми, що свідчить про підсилення поточної трендової динаміки та підтверджує імпульс руху ціни. Аналіз поведінки індексу відносної сили виявляє розширення зони перекупленості, що опосередковано вказує на підвищену ймовірність короткострокових корекційних знижень ціни. В окремих часових інтервалах простежуються ознаки слабо вираженої дивергенції між рухом ціни та значеннями RSI, що може сигналізувати про потенційне послаблення поточного тренду.

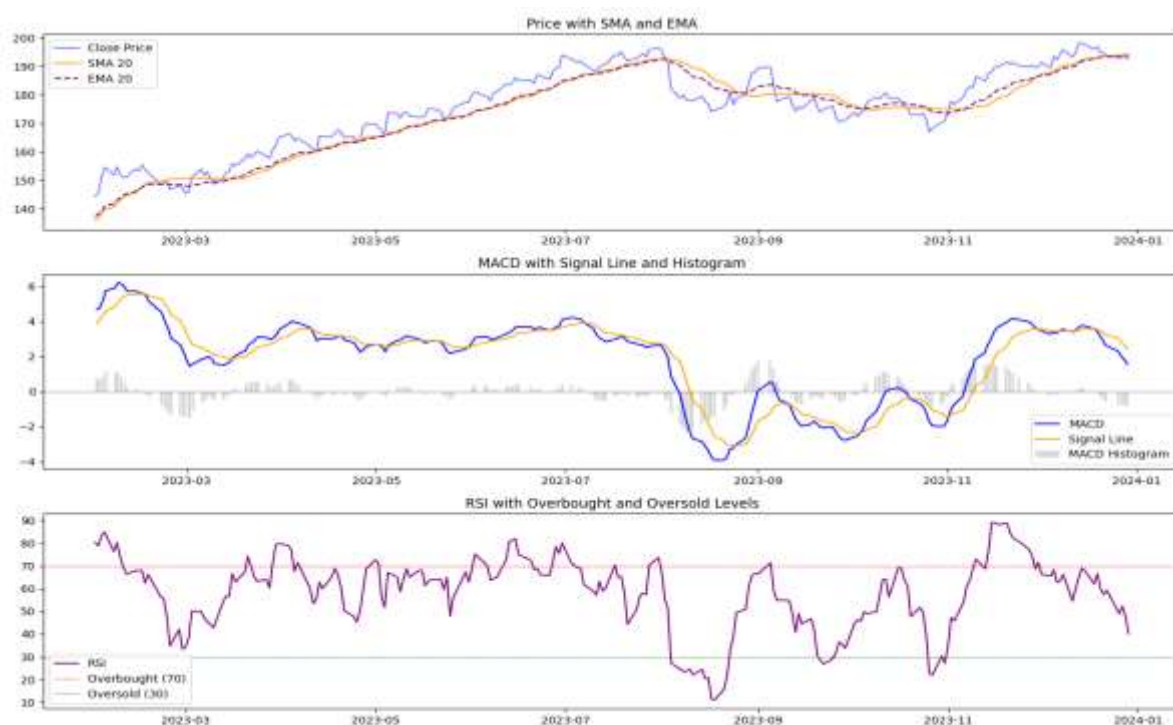


Рисунок 3.3 – Склад вихідної візуалізації результатів аналізу даних

Отже, узагальнення результатів оцінювання сукупності технічних індикаторів дозволяє обґрунтувати доцільність застосування стратегії,

орієнтованої на довгострокове утримання фінансових активів із урахуванням періодичних короткострокових коливань ринку. Структурну організацію програмного модуля, призначеного для побудови та використання моделей класу ARIMA у задачах прогнозування фінансових часових рядів, включаючи ключові функціональні компоненти, змінні та колекції для акумулювання даних, відображено на рис. 3.4. У межах реалізації також передбачено допоміжні змінні, що використовуються для збереження текстових підписів елементів графічного інтерфейсу, забезпечуючи узгодженість та зручність взаємодії користувача із системою. Головне вікно програмного модуля, що відповідає за налаштування параметрів та ініціалізацію процесу побудови ARIMA-моделі, представлено на рис. 3.5. Воно забезпечує введення необхідних параметрів моделі та організацію взаємодії користувача з підсистемою прогнозування.

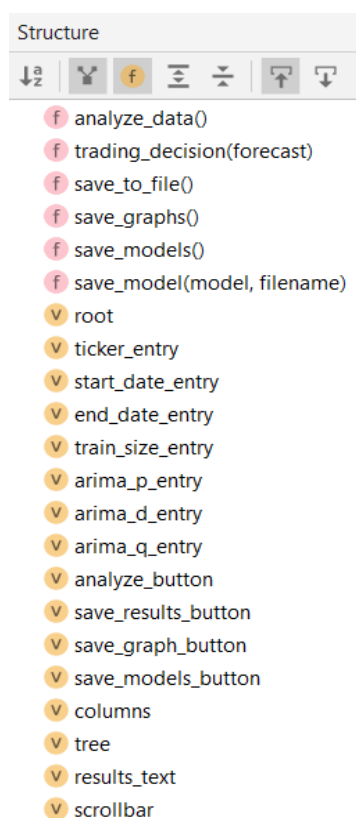


Рисунок 3.4 – Програмний склад модуля побудови моделей ARIMA

Ключові аспекти бізнес-логіки програмного модуля прогнозування на основі моделей ARIMA орієнтовані на комплексну обробку фінансових часових рядів та

формування обґрунтованих торгових рішень. Центральне місце в архітектурі займає функція `analyze_data()`, яка реалізує інтегрований сценарій обробки даних, що включає отримання та валідацію параметрів, введених користувачем через інтерфейс, завантаження історичних фінансових даних, побудову моделей ARIMA, а також подальшу оцінку їхньої якості за допомогою показника RMSE. На основі отриманих прогнозних значень додатково здійснюється генерація сигналів, що відображають доцільність відкриття довгих або коротких позицій.

Формування безпосередніх торгових рекомендацій покладено на функцію `trading_decision(forecast)`, яка інтерпретує результати прогнозування та трансформує їх у прикладні рішення щодо купівлі або продажу фінансового активу залежно від очікуваної динаміки ринку.

Збереження результатів аналітичної обробки реалізовано у функції `save_to_file()`, яка забезпечує запис підсумкових показників у текстовий файл `arima_results.txt`. У цьому файлі фіксуються значення середньоквадратичної помилки для кожної побудованої моделі, що дає змогу користувачу здійснювати подальший аналіз ефективності та порівняння альтернативних підходів.

Візуалізація результатів моделювання здійснюється засобами функції `save_graphs()`, яка формує графічні представлення для оцінювання точності та адекватності прогнозів. Зокрема, будується гістограма значень RMSE для кількох моделей, що дозволяє наочно оцінити їхню якість, а також графіки, які відображають фактичні значення часових рядів разом із прогнозованими траєкторіями. Згенеровані зображення зберігаються у директорії `visualization_arima`, де файл `arima_errors.png` містить інформацію про похибки моделей, тоді як `arima_forecasts.png` відображає співставлення реальних і прогнозних даних. Окрему функціональну роль виконує процедура `save_models()`, яка забезпечує збереження навчених моделей ARIMA у серіалізованому вигляді із застосуванням механізмів `pickle` та бібліотеки `joblib`. Такий підхід дозволяє повторно використовувати отримані моделі без необхідності їх повторного навчання, що підвищує ефективність роботи інформаційної системи прогнозування у практичних умовах трейдингу.

Модуль створення та аналіз моделей ARIMA

Ticker: AAPL
 Start Date (YYYY-MM-DD): 2020-01-01
 End Date (YYYY-MM-DD): 2023-01-01
 Train Size (0-1): 0.8
 ARIMA p: 2
 ARIMA d: 3
 ARIMA q: 1

Analyze
 Save to File
 Save Graphs
 Save Models

Model	RMSE
ARIMA(2,3,1)	58.002
Auto ARIMA	13.2653
Seasonal ARIMA	36.5106

Результати аналізу моделей ARIMA для AAPL

ARIMA (2, 3, 1) помилка RMSE: 58.00200012867189
 Auto ARIMA для RMSE: 13.265258228004212
 Seasonal ARIMA помилка RMSE: 36.51061577789545

Торгові рекомендації:
 ARIMA (2, 3, 1): Рекомендується купівля.
 Рівень впевненості: 55.00%
 Auto ARIMA: Рекомендується очікування.
 Рівень впевненості: 15.00%
 Seasonal ARIMA: Рекомендується продаж.
 Рівень впевненості: 20.00%

Рисунок 3.5 – Інтерфейс користувача форми побудови моделей ARIMA

Архітектура графічного інтерфейсу даної форми за своєю структурною організацією та принципами компоновання елементів є узгодженою з попередньо реалізованим модулем. У процесі розробки використано аналогічний набір віджетів, зокрема поля введення параметрів, елементи керування у вигляді кнопок, а також табличний компонент для відображення результатів обробки даних. Візуалізацію підсумкових результатів прогнозування та сформованих торгових рішень на основі моделей ARIMA представлено на рис. 3.6, що ілюструє ефективність застосування відповідного підходу в межах інформаційної системи підтримки трейдингових рішень.

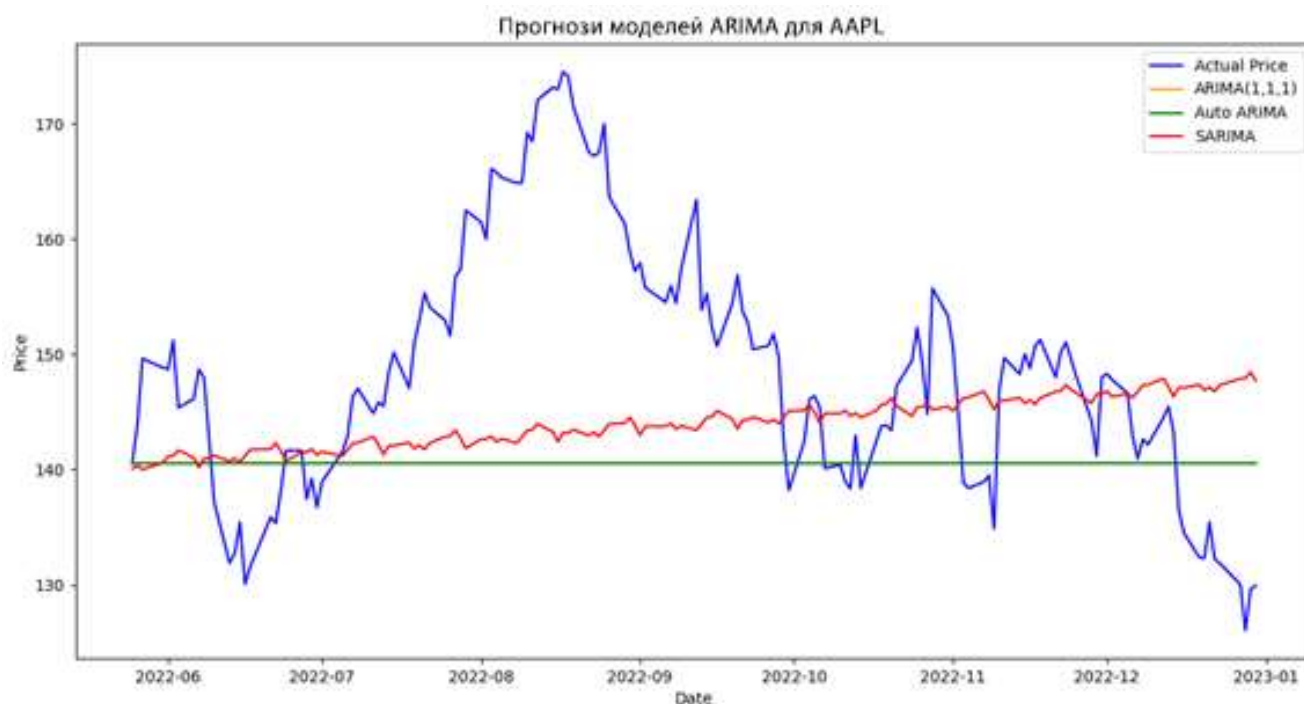


Рисунок 3.6 – Приклад візуалізації результатів прогнозу на базі ARIMA

Як свідчать результати аналізу, прогноз, сформований із використанням сезонної моделі, демонструє дещо вищий рівень відповідності фактичним даним порівняно з іншими розглянутими підходами. Водночас загальна точність прогнозування залишається обмеженою, що зумовлено використанням згладжених, усереднених значень у часовому ряді. Така особливість призводить до втрати частини короткострокових коливань, унаслідок чого застосування отриманих прогнозів без додаткового врахування технічних індикаторів не забезпечує належної ефективності при прийнятті торгових рішень.

Структурну організацію програмного модуля, орієнтованого на розробку та використання глибоких нейронних мереж для прогнозування фінансових часових рядів, включаючи основні функціональні компоненти, змінні та структури даних для збереження результатів обробки, наведено на рис. 3.7.

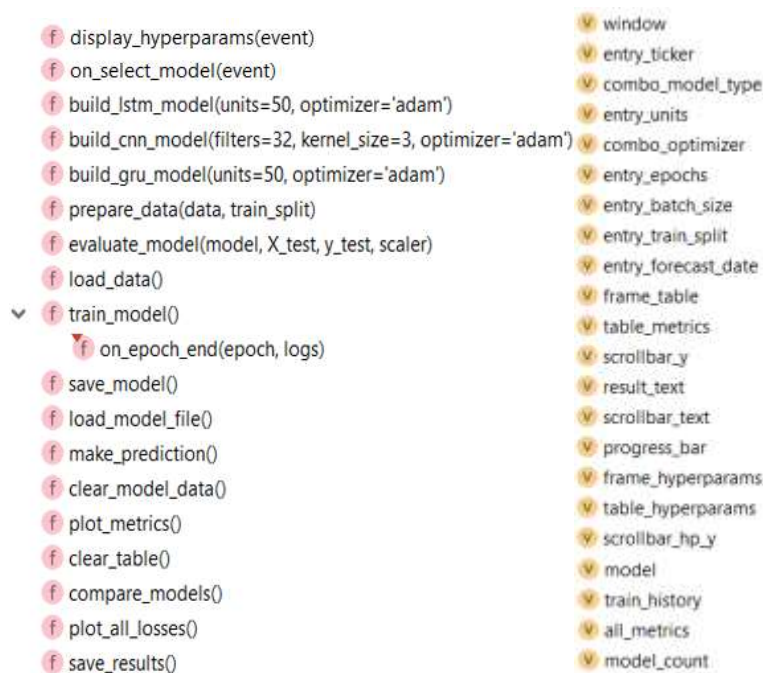


Рисунок 3.7 – Програмний склад модуля побудови моделей нейронних мереж

Функціональне ядро програмного модуля, орієнтованого на побудову та застосування глибоких нейронних мереж у задачах прогнозування фінансових часових рядів, реалізовано через сукупність взаємопов'язаних процедур, що забезпечують повний цикл обробки даних, навчання моделей та формування прогнозів. Зокрема, функція `build_lstm_model(units, optimizer)` формує архітектуру рекурентної нейронної мережі типу LSTM, яка включає два послідовні LSTM-шари, де кількість нейронів у другому шарі зменшена відносно першого, а також вихідний повнозв'язний шар з одним нейроном. Після побудови структура моделі компілюється із заданим оптимізатором та функцією втрат типу `mse`.

Аналогічно, процедура `build_cnn_model(filters, kernel_size, optimizer)` реалізує створення згорткової нейронної мережі, що містить згортковий шар із визначеними параметрами, шар підвибірки типу `max pooling`, а також два повнозв'язні шари, завершені вихідним нейроном. Параметри оптимізації та функція втрат встановлюються за тим самим принципом, що й для LSTM.

Функція `build_gru_model(units, optimizer)` забезпечує побудову моделі на основі GRU-архітектури, яка за логікою організації є близькою до LSTM, однак

відрізняється внутрішніми механізмами керування станами, що дозволяє ефективніше працювати з певними типами часових залежностей.

Підготовка вхідних даних виконується у функції `prepare_data(data, train_split)`, де часовий ряд трансформується у формат, придатний для навчання нейронних мереж. У межах цієї процедури здійснюється нормалізація значень за допомогою `MinMaxScaler`, формування вхідних послідовностей фіксованої довжини та відповідних цільових значень, після чого дані розподіляються на навчальну та тестову вибірки відповідно до заданої пропорції.

Ключовим етапом є функція `train_model()`, яка координує процес навчання. Вона виконує завантаження даних, ініціалізацію обраної моделі з урахуванням заданих гіперпараметрів, організовує процес навчання з відображенням прогресу через відповідний інтерфейсний елемент, а також здійснює оцінювання якості моделі на тестових даних за допомогою процедури `evaluate_model()`. Отримані метрики та параметри зберігаються для подальшого аналізу та порівняння.

Функція `evaluate_model(model, X_test, y_test, scaler)` реалізує обчислення основних показників якості, зокрема середньоквадратичної помилки, середньої абсолютної помилки, коефіцієнта детермінації та відносної похибки прогнозу. Перед розрахунком здійснюється зворотне масштабування прогнозованих і фактичних значень до початкового діапазону. Завантаження вхідних даних організовано у функції `load_data()`, яка отримує фінансові часові ряди за заданим тикером із джерела YF та передає їх на подальшу обробку. Процедура `make_prediction()` забезпечує формування прогнозу на основі попередньо навченої моделі. У разі відсутності активної моделі виконується її завантаження з файлу. Прогноз генерується на основі останніх спостережень тестової вибірки, після чого формуються відповідні торгові рекомендації з урахуванням отриманих метрик.

Функція `save_model()` реалізує збереження навченої моделі у файлову систему в межах директорії `ready_models`, що дозволяє повторно використовувати результати навчання. Завантаження моделей здійснюється за допомогою `load_model_file()`, яка перевіряє наявність відповідного файлу та забезпечує обробку помилок у разі його відсутності. Візуалізація процесу навчання

реалізована у функції `plot_metrics()`, що формує графіки зміни функції втрат для навчальної та тестової вибірок і зберігає їх у каталозі `visualization_dnn`.

Для аналітичного порівняння моделей використовується функція `compare_models()`, яка виконує зіставлення за ключовими метриками якості та тривалістю навчання, формуючи відповідні графічні представлення у вигляді гістограм. Додатково процедура `plot_all_losses()` дозволяє отримати узагальнений графік функції втрат для всіх досліджуваних моделей, що спрощує їх порівняльний аналіз. Функція `save_results()` забезпечує збереження підсумкових результатів прогнозування у текстовому вигляді в директорії `results`, що сприяє подальшому використанню отриманих даних. Очищення проміжних результатів та службової інформації реалізовано через процедури `clear_model_data()` та `clear_table()`, які відповідають за скидання метрик, історії навчання та табличних даних відповідно.

Графічний інтерфейс головного вікна програмного модуля, призначеного для роботи з глибокими нейронними мережами, наведено на рис. 3.8, де відображено засоби налаштування параметрів моделей, запуску процесів навчання та аналізу результатів.



Рисунок 3.8 – Інтерфейс користувача головної форми

Графічний інтерфейс програмного модуля передбачає наявність елементів керування, що забезпечують повний цикл роботи з моделями глибокого навчання, включаючи їх ініціалізацію, навчання, збереження та подальше використання. Зокрема, реалізовано кнопки для запуску процесу навчання моделей, виконання прогнозування, завантаження та збереження моделей у файлову систему, побудови графічних залежностей функції втрат, фіксації результатів прогнозування, проведення порівняльного аналізу моделей, а також очищення табличних структур даних.

Для забезпечення детального аналізу параметрів налаштування кожної побудованої моделі передбачено окремий табличний компонент, який містить значення гіперпараметрів і оснащений механізмом прокручування, що дозволяє ефективно працювати з великими обсягами інформації.

Порівняльну характеристику значень функції втрат для розроблених моделей машинного навчання наведено на рис. 3.9, що дозволяє оцінити їхню збіжність та якість навчання в контексті задач прогнозування фінансових часових рядів.

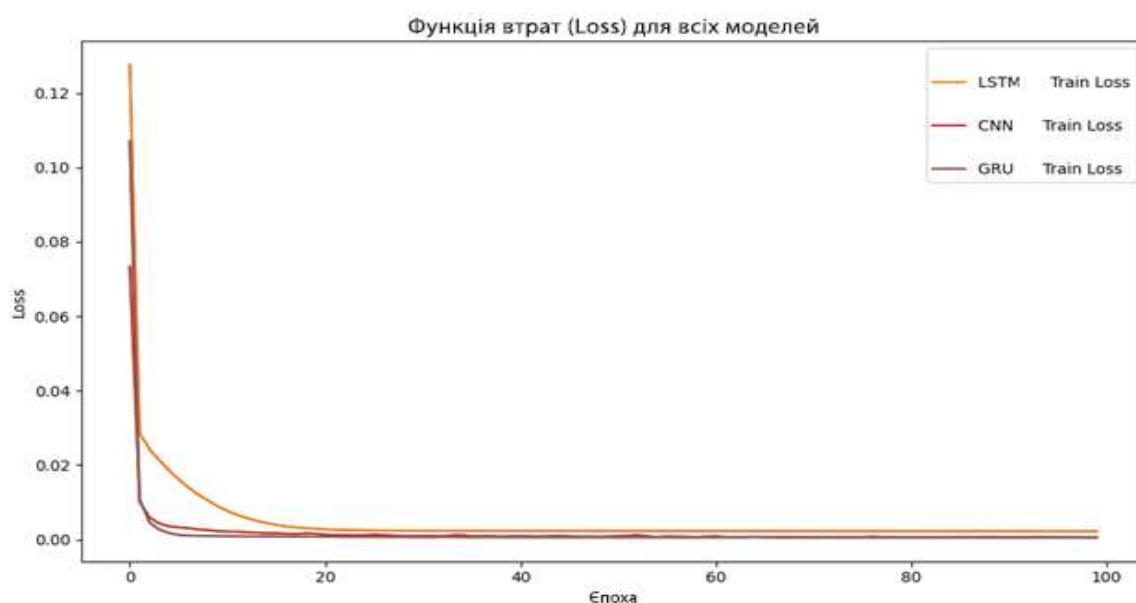


Рисунок 3.9 – Порівняння метрик loss по моделях

Візуалізація порівняння створених моделей машинного навчання за тривалістю навчання представлена на рис. 3.10.

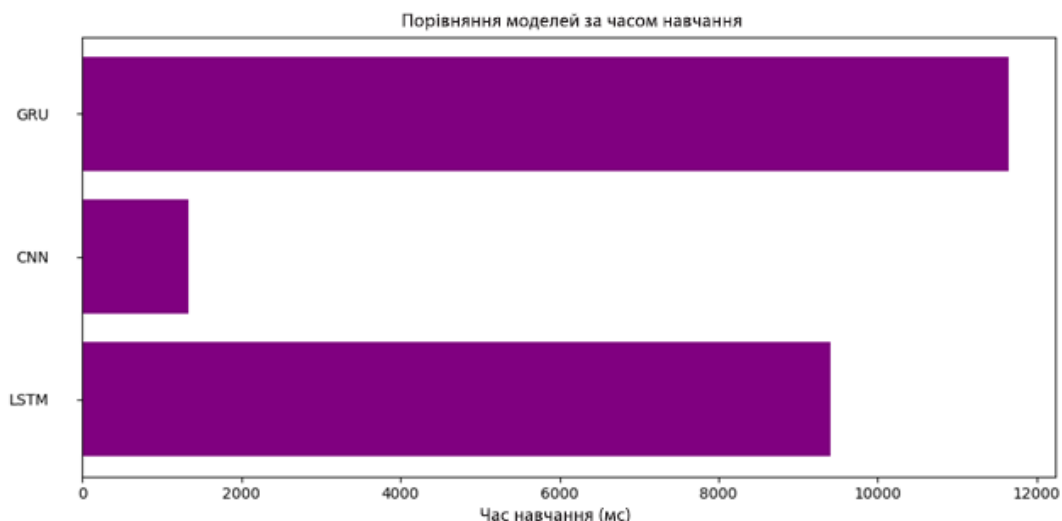


Рисунок 3.10 – Порівняння моделей за тривалістю навчання

Аналіз динаміки навчання свідчить про відсутність виражених ознак перенавчання протягом 100 епох, що підтверджує адекватність обраних гіперпараметрів та стабільність процесу оптимізації. Найбільш інтенсивне зниження значень функції втрат спостерігається для моделі GRU, причому суттєве покращення досягається вже на початкових ітераціях навчання. Водночас для моделі LSTM характерна більш поступова збіжність, і стабілізація значень loss відбувається орієнтовно після 20-ї епохи, що вказує на повільніший процес адаптації вагових коефіцієнтів. Порівняльний аналіз розроблених моделей за значеннями основних метрик якості представлено на рис. 3.11.

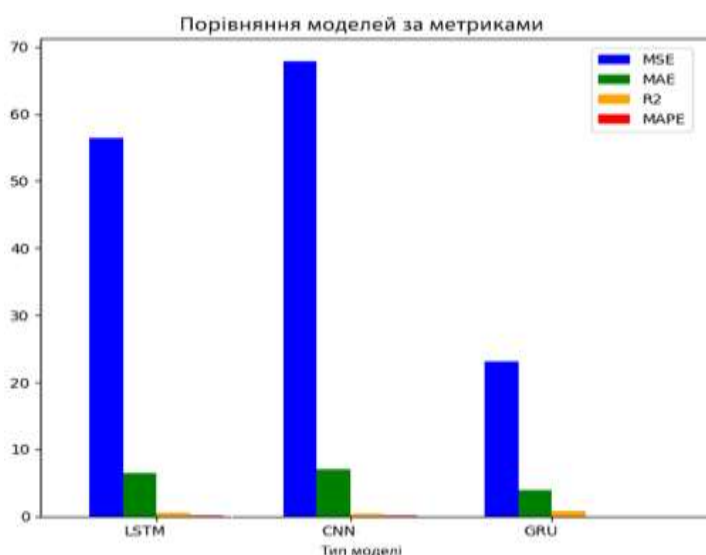


Рисунок 3.11 – Порівняння створених моделей за значеннями метрик

Отримані результати свідчать, що згортова нейронна мережа характеризується найменшою тривалістю навчання, проте поступається іншим моделям за точністю прогнозування. Натомість модель LSTM потребує значно більших обчислювальних витрат, оскільки тривалість її навчання перевищує аналогічний показник для CNN більш ніж у шість разів. Найбільш ресурсоемною виявилася модель GRU, для якої зафіксовано максимальний час навчання. Водночас саме вона продемонструвала найкращі результати за коефіцієнтом детермінації R^2 , що свідчить про вищу якість апроксимації залежностей у часовому ряді. Важливо зазначити, що для забезпечення коректності порівняння у всіх експериментах використовувався однаковий алгоритм оптимізації Adam.

Після апробації базового функціоналу розробленої інформаційної системи та підтвердження її працездатності в умовах практичного застосування доцільним є проведення подальшого дослідження окремих програмних модулів з метою поглибленого аналізу їх ефективності та можливостей удосконалення..

3.2 Дослідження та аналіз результатів роботи системи

Для проведення експериментальних досліджень розроблено окремий програмний скрипт у середовищі Jupyter Notebook, у межах якого послідовно реалізовано етапи попереднього аналізу фінансових даних, побудови моделей прогнозування та оцінювання їхньої ефективності. Така організація дозволяє забезпечити поетапну відтворюваність обчислювального процесу та контроль кожної стадії обробки часових рядів.

З метою автоматизації процесу підбору оптимальних значень гіперпараметрів для моделей машинного навчання інтегровано підхід grid search, що забезпечує систематичний перебір комбінацій параметрів із подальшим вибором конфігурації, яка демонструє найкращі показники якості.

У процесі виконання скрипту в консольному виводі відображається узагальнена інформація та метадані щодо доступних фінансових інструментів, зокрема тікери компаній Apple, Microsoft та Google. Додатково подається

інформація про середній рівень капіталізації на момент аналізу, а також дані квартальної звітності щодо дивідендних виплат за активами в межах поточного року. Для виявлення закономірностей у динаміці фінансових часових рядів доцільним є застосування графічного аналізу, який дозволяє відобразити зміну показників у безперервному часовому вимірі. Візуалізація трендів вартості закриття акцій різних компаній за останній рік наведена на рис. 3.12.



Рисунок 3.12 – Тренди вартості закриття акцій різних компаній

Як випливає з результатів візуального аналізу, максимальні значення вартості активів характерні для компанії Microsoft, тоді як найнижчі рівні спостерігаються у Google, що відображає загальні відмінності у ринковій капіталізації та динаміці цінових рядів досліджуваних інструментів.

Подальший етап дослідження спрямовано на детальніше вивчення поведінки фінансових часових рядів та формування потенційних торгових рішень на основі технічних індикаторів, зокрема на прикладі акцій компанії Apple (тікер AAPL), які характеризуються відносно стабільною структурою цінових змін та зручністю для аналізу усереднених тенденцій. Візуалізацію динаміки цін із накладенням ковзних середніх SMA та ЕМА наведено на рис. 3.13.

Аналіз отриманих графічних даних свідчить, що в період із серпня по листопад попереднього року спостерігалось зниження рівня ринкової

волатильності та переважання коливальних рухів у межах сформованого тренду. Така поведінка цінового ряду характеризується відносною нестабільністю із амплітудою коливань на рівні приблизно 10–15%, що може бути інтерпретовано як період зниження інтенсивності купівельної активності.

Ймовірною причиною зазначених змін можуть виступати як внутрішні корпоративні фактори, так і загальноекономічні процеси, що впливають на поведінку учасників ринку. Водночас після листопада спостерігається поступова стабілізація значень SMA та ЕМА, що супроводжується формуванням стійкішого висхідного тренду, який за своєю структурою наближається до динаміки, характерної для початку літнього періоду.

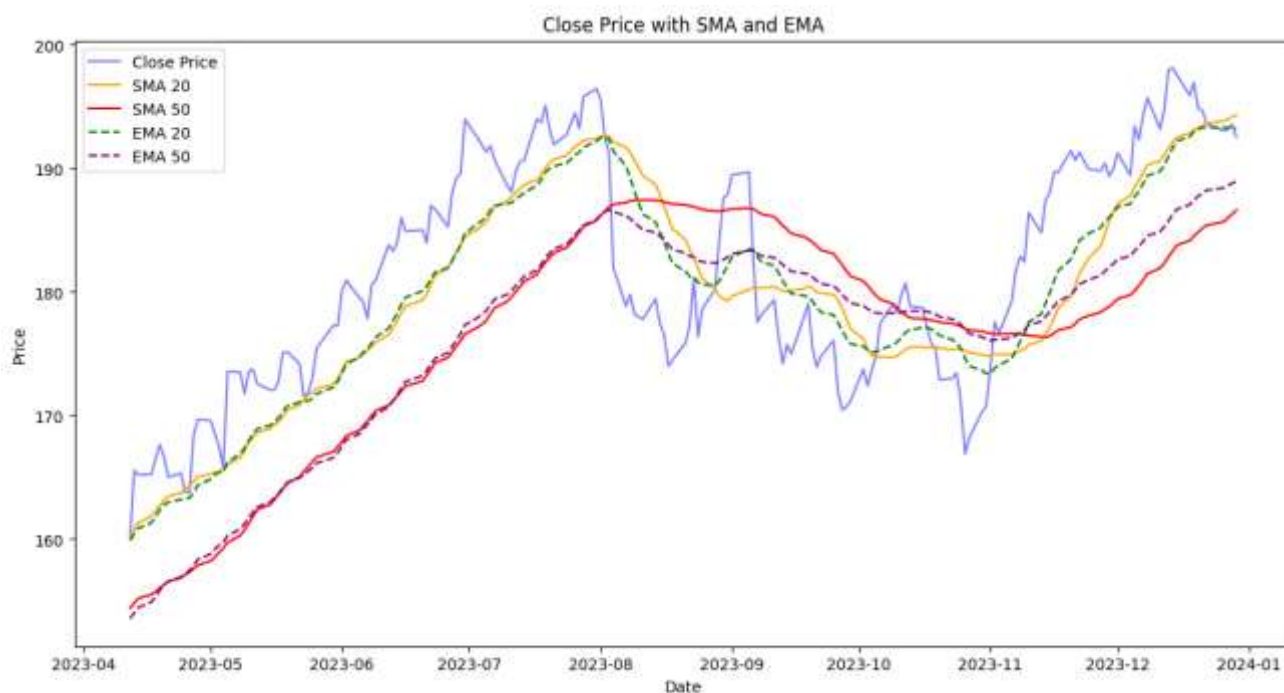


Рисунок 3.13 – Графік цін акцій з трендами

Графічне представлення торгових сигналів, сформованих на основі взаємодії індикаторів SMA та ЕМА, наведено на рис. 3.14. Загальна структура сигналів демонструє переважання рекомендацій щодо купівлі активу, що узгоджується з виявленою висхідною динамікою цінового ряду. Водночас у межах окремих локальних корекційних фаз спостерігаються короткочасні спадні рухи, які, однак, також здебільшого інтерпретуються алгоритмом як потенційні точки входу в довгу

позицію. Це свідчить про те, що виявлена система сигналів орієнтована переважно на фіксацію висхідного тренду та реагує на короткострокові просідання як на можливості для подальшого придбання активу.

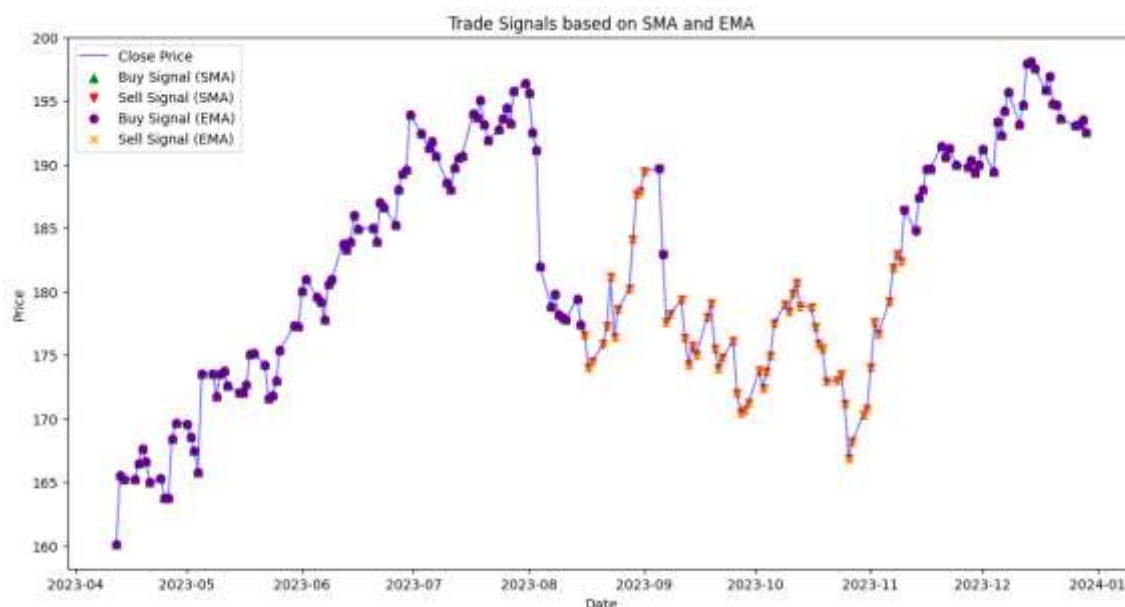


Рисунок 3.14 – Графік торгових сигналів на основі SMA та EMA

Графічну динаміку різниці між швидкою та повільною ковзними середніми представлено на рис. 3.15. Загалом значення відповідних індикаторів характеризуються високим ступенем наближеності, однак у часовому інтервалі з середини серпня відбувається зміщення співвідношення між SMA та EMA, що супроводжується переходом у негативну область. Такий характер поведінки може інтерпретуватися як посилення очікувань щодо зниження ринкової активності та зростання ймовірності корекційного руху вниз. Водночас уже з початку вересня спостерігається відновлення висхідної тенденції SMA, яка набуває більш стабільного характеру та зберігає поступову позитивну динаміку в подальших періодах. Розподіл цін закриття у поєднанні з відповідними торговими сигналами наведено на рис. 3.16.

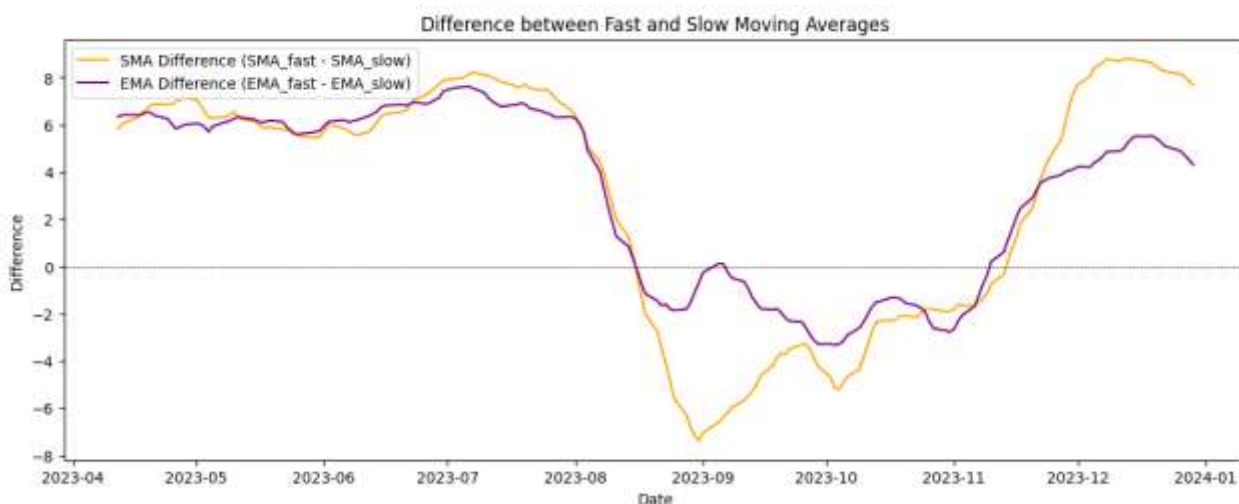


Рисунок 3.15 – Графік різниці між швидким і повільним ковзними середніми

Отримані результати узгоджуються з попередньо встановленими закономірностями, демонструючи наявність виражених хвилеподібних коливань вартості активу. Зокрема, характер динаміки свідчить про періодичні фази підвищення та зниження цінового рівня, які найбільш виразно проявляються в осінній часовий інтервал.

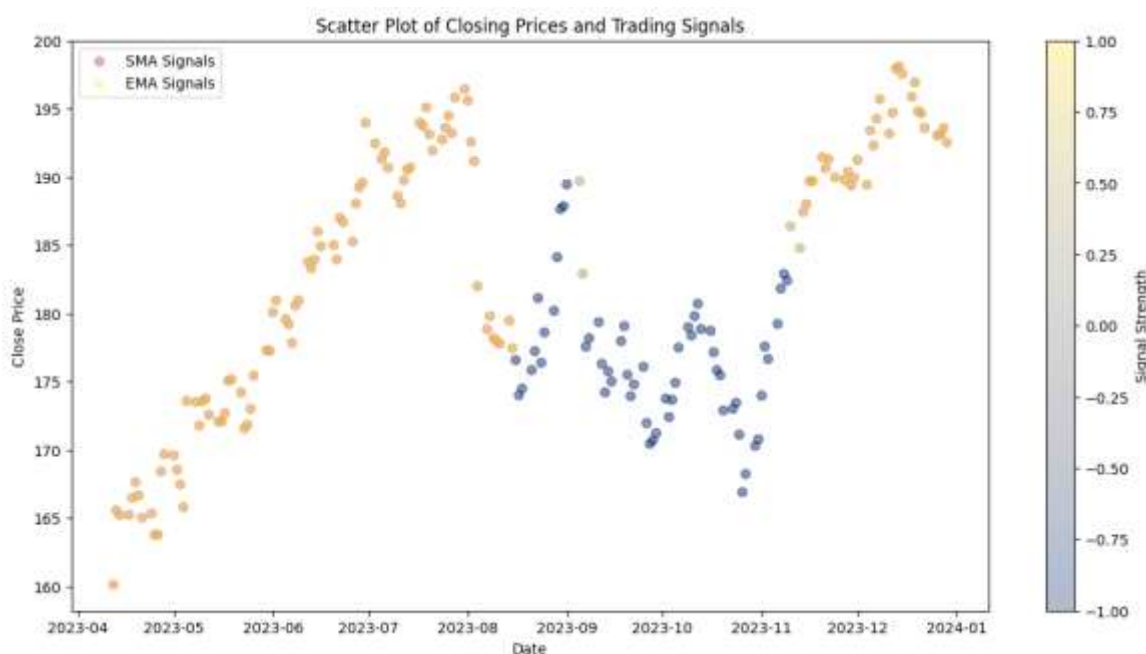


Рисунок 3.16 – Графік розподілу цін закриття та сигналів

Графік RSI із зонами перекупленості та перепроданості наведено на рис. 3.17.



Рисунок 3.17 – Графік RSI

Аналіз значень індикатора RSI демонструє домінування зони перекупленості над зоною перепроданості, що вказує на переважання висхідних ринкових настроїв у межах досліджуваного періоду. Така поведінка індикатора може бути інтерпретована як ознака загального зростання вартості фінансових активів та підвищення ринкової капіталізації відповідних компаній. Графічне представлення індикатора MACD у поєднанні з сигнальною лінією та гістограмою наведено на рис. 3.18, що дозволяє додатково оцінити силу та напрямок трендових рухів у часовому ряді. Додатковим аналітичним інструментом, який використовується для поглибленого аналізу ринкової динаміки, є смуги Боллінджера. Вони формуються навколо ковзного середнього значення ціни та включають верхню і нижню межі, які визначають статистично обґрунтований діапазон волатильності, що дозволяє оцінювати ступінь відхилення ціни від її середнього рівня.

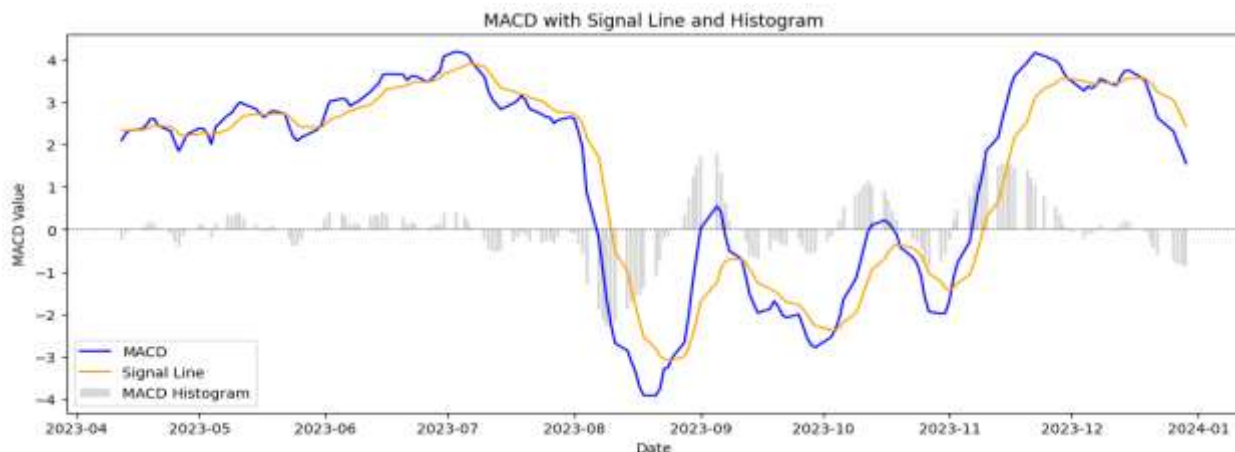


Рисунок 3.18 – Графік MACD із сигнальною лінією та гістограмою

Середня лінія смуг Боллінджера, що відповідає 20-періодній SMA, виконує роль індикатора узагальненого напрямку руху ціни, формалізуючи середнє значення вартості активу за заданий часовий інтервал та дозволяючи ідентифікувати домінуючий тренд. У межах досліджуваного періоду спостерігається чітко виражена висхідна тенденція, що підтверджує переважання позитивної динаміки ринку. Верхня межа смуг формується на відстані двох стандартних відхилень від центральної лінії, відображаючи потенційні зони підвищених цінових значень та екстремумів ринку. Нижня смуга, відповідно, розташовується на аналогічній відстані нижче середнього значення та характеризує діапазон відносно занижених цінових рівнів. Така побудова дозволяє оцінювати статистично обґрунтований інтервал волатильності та ступінь відхилення поточної ціни від її середнього значення. Звуження смуг, яке інтерпретується як період зниження ринкової волатильності, фіксується наприкінці липня та грудня, що може розглядатися як потенційний індикатор майбутньої зміни трендового напрямку.. Візуальне представлення смуг Боллінджера у поєднанні з ціновою динамікою наведено на рис. 3.19, тоді як порівняльна гістограма значень RMSE для побудованих моделей ARIMA представлена на рис. 3.20.

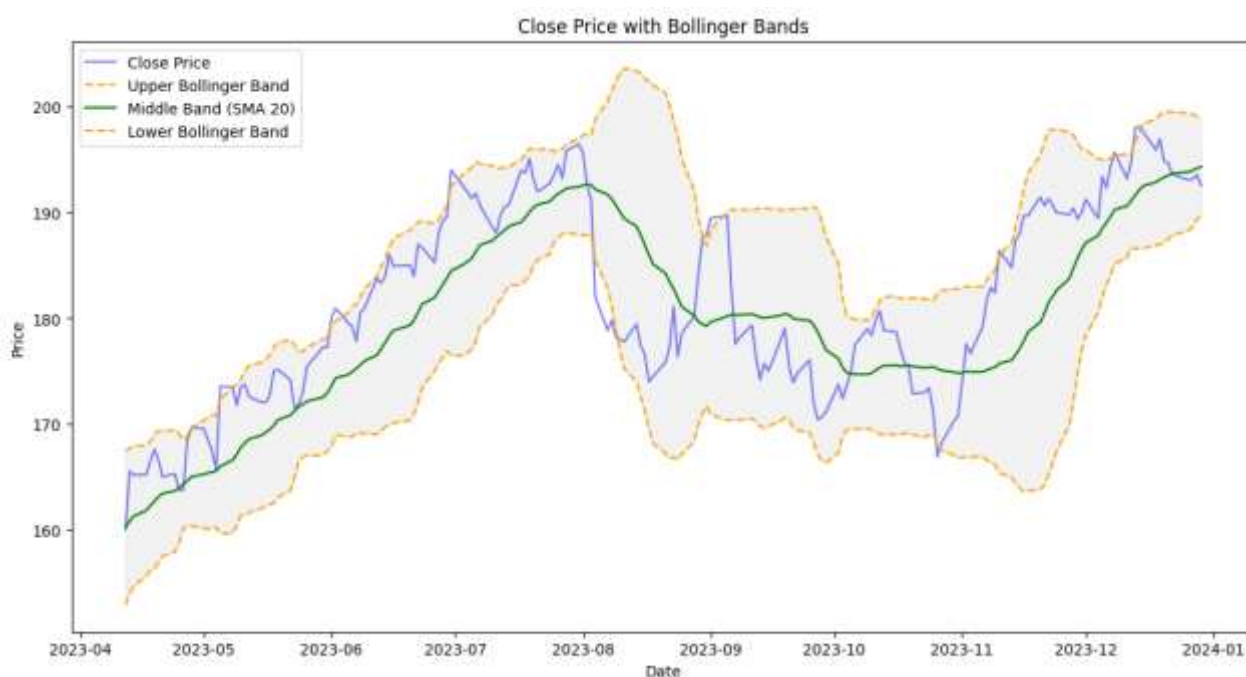


Рисунок 3.19 – Візуалізація смуг Боллінджера з графіком цін

Натомість максимальне розширення діапазону спостерігається в інтервалі з кінця серпня до початку жовтня, що свідчить про підвищену ринкову нестабільність у цей період. З точки зору торгової стратегії, найбільш доцільними є сигнали на придбання активу в умовах висхідного тренду при наближенні ціни до нижньої межі смуг Боллінджера, що у межах досліджуваних даних переважно відповідає періоду кінця жовтня

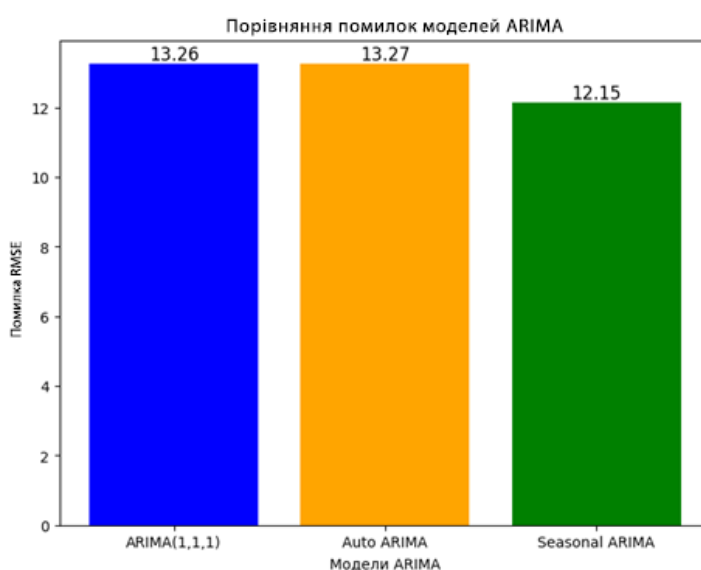


Рисунок 3.20 – Гістограма порівняння значень RMSE моделей ARIMA

Отримані трейдингові рішення, сформовані різними моделями прогнозування, демонструють певну неоднозначність та варіативність інтерпретацій. Автоматизована модель характеризується переважно песимістичним сценарієм поведінки, оскільки у більшості випадків генерує сигнали, орієнтовані на продаж активу. Натомість перша розглянута модель відображає більш збалансований характер прийняття рішень, за якого сигнали на продаж фіксуються переважно на початкових етапах та наприкінці досліджуваного тренду.

Найбільш змістовні та структуровано узгоджені результати демонструє сезонна модель, яка формує торгові сигнали відповідно до виявленої циклічності даних. Зокрема, придбання активу здійснюється на етапах стабільного висхідного тренду, тоді як продажі передують періодам сезонного зниження. У подальшому, у

другій половині року, знову фіксуються сигнали на купівлю в умовах відновлення нестабільного, але стійкого трендового руху. Така поведінка моделі може бути інтерпретована як більш адаптивна стратегія, орієнтована на довгострокове утримання позицій із урахуванням циклічних коливань ринку.

Для забезпечення системного аналізу та порівняння результатів виконано узагальнення всіх розрахованих метрик у структурованому вигляді, що дозволяє оцінити їхню ефективність у єдиному аналітичному просторі. Порівняльні значення показників якості моделей машинного навчання наведено в таблиці 3.1.

Таблиця 3.1 – Результати порівняння метрик моделей

Назва метрики	Модель машинного навчання			
	LSTM	CNN	GRU	CNN+LSTM
MSE	40,423	36,875	15,357	41,134
MAE	5,425	5,244	3,164	5,337
RMSE	6,357	6,072	3,918	6,413
R ²	0,665	0,695	0,873	0,66
MAPE	0,037	0,035	0,022	0,037
sMAPE	3,662	3,543	2,145	3,616

Таким чином, остаточне торгове рішення доцільно формувати не на основі окремого індикатора чи моделі, а шляхом узгодженого врахування результатів усіх використаних підходів. Прийняття рішень у такій системі має здійснюватися користувачем спільно з аналітиком даних із використанням комплексної інтерпретації отриманих сигналів та оцінок торгових індикаторів. У випадку повної узгодженості прогнозів різних моделей формується найбільш надійний та однозначний сценарій поведінки ринку. Якщо ж спостерігаються суперечності між результатами моделей, доцільним є або вибір домінуючої стратегії на основі більш точних метрик, або тимчасове утримання від активних дій до появи стабільного тренду з чітко вираженою динамікою.

Висновки. У межах даного розділу здійснено розробку програмного забезпечення, орієнтованого на використання в системах підтримки прийняття торгових рішень. Описано структуру та особливості сформованого набору даних, а

також розроблено архітектуру програмного проєкту, що реалізує основні етапи аналітичного процесу. Програмна реалізація виконана у вигляді десктопного застосунку, який структурно складається з трьох функціональних модулів, що забезпечують обробку даних, побудову моделей та візуалізацію результатів.

Проведено експериментальні дослідження розроблених моделей машинного навчання з подальшим аналізом та інтерпретацією отриманих результатів. Встановлено, що комплексне застосування методів технічного аналізу та прогнозних моделей дозволяє підвищити якість торгових рішень та зменшити ризики помилкових інтерпретацій ринкових сигналів порівняно з використанням окремих підходів.

Найбільш перспективним напрямом подальшого розвитку є застосування моделей глибокого навчання, здатних здійснювати багатогоризонтне прогнозування часових рядів. Серед розглянутих підходів найвищу прогнозну ефективність демонструють моделі типу GRU.

4 ОХОРОНА ПРАЦІ

В даний час в ІТ-галузі питання охорони праці мають велике значення. Хоча галузь і не передбачає роботи з небезпечним обладнанням чи фізичними ризиками у традиційному розумінні, проте характер праці фахівців-комп'ютерників висуває свої, не менш серйозні проблеми. Високе інформаційне навантаження, тривала робота за монітором, щільна взаємодія в командах та специфіка дистанційного формату - усе це потребує ретельного підходу до організації безпечних умов праці. Тому, в цьому розділі будуть детально описані роль і місце служби охорони праці в системі управління підприємством ІТ- галузі та вплив роботизації та автоматизації технологічних процесів на рівень безпеки праці.

4.1 Роль і місце служби охорони праці в системі управління підприємством ІТ- галузі

На ІТ-підприємствах служба охорони праці відіграє ключову роль у запобіганні травматизму та створенні безпечного робочого середовища. Її завдання включають профілактику небезпечних факторів, покращення умов праці та забезпечення відповідності нормам охорони праці у всіх підрозділах.

Фахівці з охорони праці в ІТ-компанії виконує низку ключових функцій, що охоплюють як фізичні, так і психологічні аспекти безпеки. Їхні основні функції наведемо далі.

Оцінка ризиків та їх мінімізація. Проведення регулярних оцінок робочих місць, виявлення потенційних небезпек, пов'язаних з тривалою роботою за комп'ютером (наприклад, синдром карпального каналу, проблеми із зором, болі в спині), неправильним облаштуванням робочих місць, освітленням, вентиляцією. Розробка та впровадження заходів для покращення умов праці.

Розробка та впровадження нормативних документів. Створення внутрішніх інструкцій, положень та стандартів з охорони праці, що відповідають вимогам чинного законодавства України та специфіці ІТ-діяльності. Це включає інструкції

з безпечної експлуатації комп'ютерної техніки, правила пожежної безпеки, порядок дій у надзвичайних ситуаціях тощо.

Навчання та інструктажі: Організація та проведення вступних, первинних, повторних та позапланових інструктажів для всіх співробітників, а також спеціального навчання з питань охорони праці для керівного складу та осіб, відповідальних за окремі ділянки.

Контроль за дотриманням вимог з охорони праці. Регулярний моніторинг дотримання співробітниками правил та інструкцій з охорони праці, використання засобів індивідуального захисту (наприклад, при роботі з електрообладнанням, у серверних приміщеннях), стану робочих місць.

Облік і аналіз нещасних випадків та професійних захворювань.

Розслідування нещасних випадків та професійних захворювань. Участь у розслідуванні причин нещасних випадків на виробництві та випадків професійних захворювань, розробка заходів щодо запобігання їх повторенню.

Взаємодія з контролюючими органами. Представлення інтересів підприємства при перевірках органами державного нагляду за охороною праці, надання необхідної документації.

Психологічна підтримка та профілактика стресу. Враховуючи високі навантаження та інтелектуальну працю в ІТ, фахівець може ініціювати заходи з профілактики емоційного вигорання та психологічного стресу, наприклад, через організацію перерв, зон відпочинку, інформаційних кампаній щодо здорового способу життя.

Ведення документації: Систематизація та зберігання всієї необхідної документації з охорони праці (журнали інструктажів, протоколи перевірок, акти розслідувань тощо).

Підготовка наказів і розпоряджень з питань охорони праці;

Консультавання працівників щодо безпечних методів роботи та інформування про нові норми.

Місце служби охорони праці в системі управління ІТ-підприємством наступне.

Фахівці з охорони праці, як правило, підпорядковуються безпосередньо керівнику підприємства або одному з його заступників [26]. Це забезпечує необхідний рівень незалежності та повноважень для ефективного виконання своїх функцій. На великих підприємствах може існувати окремий відділ охорони праці.

Інтеграція фахівців з охорони праці в систему управління ІТ-підприємством є ключовою для ряду завдань, наведемо їх далі.

Зниження ризиків. Запобігання виробничому травматизму та професійним захворюванням, що призводить до зменшення витрат на лікування, виплати компенсацій та штрафи.

Підвищення продуктивності праці. Безпечні та комфортні умови праці сприяють кращому самопочуттю співробітників, знижують рівень стресу та підвищують їхню працездатність та мотивацію.

Дотримання законодавства. Уникнення правових наслідків, пов'язаних з порушенням норм охорони праці, включаючи штрафи та судові розгляди.

Формування позитивного іміджу компанії як роботодавця. Демонстрація турботи про співробітників покращує репутацію компанії як роботодавця, що є важливим для залучення та утримання кваліфікованих ІТ-фахівців.

Згідно із Законом України «Про охорону праці», фахівці охорони праці можуть видавати обов'язкові приписи щодо усунення порушень, відсторонювати працівників, які не пройшли навчання, та зупиняти роботу небезпечного обладнання. Скасувати припис може лише керівник підприємства [26]. Фахівець також відповідає за дотримання законодавства, ведення звітності та документації.

У контексті цифрової трансформації роль служби охорони праці є стратегічною. Вона не лише забезпечує виконання нормативів, а й формує безпечну цифрову робочу культуру, інтегруючи технології, такі як ІІІ, VR/AR, для вирішення викликів, пов'язаних з ергономікою, ментальним здоров'ям і віддаленою роботою. Це вимагає технічних навичок і вміння розв'язувати складні задачі.

Фахівці з охорони праці вирішують також специфічні проблеми в ІТ-галузі:

- обробка великих обсягів даних про інциденти та відповідність вимогам;

- впровадження нових комп'ютерних технологій;
- подолання опору змінам з боку працівників.
- впровадження ергономічних рішень;
- впровадження програми підтримки психічного здоров'я працівників;
- інтеграція сучасних ІТ технологій у навчання та контроль знань.

При реалізації заходів з охорони праці для працівників працюючи віддалено, фахівці з охорони праці виконує наступну роботу.

Розробка політики безпеки для віддаленої роботи, що відповідає законодавчим вимогам.

Проведення віртуальних перевірок робочих місць (наприклад, через відеозв'язок) для оцінки ергономіки та виявлення потенційних ризиків у домашніх офісах.

Надання працівникам чек-листів для самостійної оцінки їхнього робочого місця.

Створення інтерактивних навчальних модулів і регулярне проведення вебінарів, адаптованих до умов віддаленої роботи.

Ефективне навчання та контроль знань з охорони праці відіграють ключову роль у формуванні культури безпеки. Сучасні методи, такі як онлайн-платформи (наприклад, ЄСОН), гейміфікація та технології VR/AR, дозволяють підвищити результативність навчання, забезпечуючи доступність, практичну спрямованість та залученість працівників.

Тому, для вдосконалення системи охорони праці в ІТ-секторі службі охорони праці рекомендовано регулярно оновлювати навчальний контент, впроваджувати системи зворотного зв'язку, проводити імітаційні аудити та інтегрувати принципи безпеки в корпоративну культуру.

Таким чином, діяльність служби охорони праці стає не лише формальним обов'язком, а й основою для створення здорового та продуктивного робочого середовища.

4.2 Вплив роботизації та автоматизації технологічних процесів на безпеку праці

Впровадження роботизованих систем та автоматизації технологічних процесів має значний та багатогранний вплив на безпеку праці. Цей вплив може бути як позитивним, так і створювати нові виклики.

Позитивні чинники впливу на безпеку праці наведемо далі.

Усунення небезпечних робіт. Роботи та автоматизовані технологічні системи можуть виконувати завдання, які є надто небезпечними, монотонними, фізично важкими або шкідливими для людини. Це включає роботу в умовах високих температур, впливу хімічних речовин, радіації, підняття важких вантажів, виконання одноманітних рухів, що призводять до травм від повторюваних навантажень. Наприклад, в атомній енергетиці роботи незамінні для виконання завдань у зонах з підвищеним рівнем радіації, де перебування людини є неприпустимим.

Зменшення людських помилок. Автоматизовані системи працюють за чітко визначеними сценаріями, що мінімізує вплив людського фактору на виникнення аварій та травм. Це особливо важливо у складних і високоточних операціях.

Підвищення точності та контролю. Роботи забезпечують високу точність виконуваних операцій, що знижує ризик пошкодження обладнання або продукції, а отже, і потенційних небезпек для працівників. Системи моніторингу можуть фіксувати рухи співробітників і техніки, сигналізувати про небезпечні зони або потенційні зіткнення.

Покращення ергономіки. Автоматизація може звільнити працівників від вимушених поз, надмірних фізичних навантажень та повторюваних рухів, що сприяє зменшенню м'язово-скелетних розладів.

Швидке реагування на надзвичайні ситуації. У деяких випадках роботи можуть використовуватися для гасіння пожеж, розгрібання завалів або виконання інших рятувальних робіт у небезпечних умовах, мінімізуючи ризик для рятувальників.

Проте, автоматизація та роботизація створює нові проблеми та ризики для безпеки праці, наведемо їх опис далі.

Ризик зіткнень. Співіснування роботів та людей на виробництві може призвести до зіткнень, якщо не впроваджено належних систем безпеки, таких як захисні огорожі, сенсори та системи зупинки.

Складність обслуговування. Обслуговування, налагодження та ремонт роботизованих систем вимагають високої кваліфікації та можуть бути небезпечними, якщо не дотримуватися чітких протоколів безпеки (наприклад, повна зупинка всіх агрегатів перед регулюванням або ремонтом).

Кібербезпека. Зростає ризик кібератак на автоматизовані системи, що може призвести до збоїв у їх роботі, неконтрольованих рухів або інших небезпечних ситуацій.

Соціально-психологічний вплив полягає у ряді аспектів, наведемо їх опис далі.

Втрата робочих місць. Роботизація та автоматизація може призвести до скорочення кількості низькокваліфікованих робочих місць, що створює стрес та невпевненість у працівників, що, своєю чергою, може мати негативний вплив на їхнє психічне здоров'я.

Перекваліфікація та навчання. Працівникам, що залишаються, часто потрібна перекваліфікація для роботи з новими технологіями, що може бути джерелом стресу.

Соціальна ізоляція. Взаємодія з роботами може зменшити соціальну взаємодію між працівниками, що може негативно вплинути на психологічний комфорт.

"Ефект надмірної довіри". Працівники можуть надмірно довіряти автоматизованим системам, що може призвести до зниження пильності та ігнорування потенційних небезпек.

Заходи для забезпечення безпеки праці в умовах автоматизації та роботизації.

Для мінімізації ризиків та максимального використання переваг автоматизації та роботизації для безпеки праці необхідно ряд процедур, наведемо їх опис далі.

Комплексний підхід до проектування. Системи автоматизації повинні проектуватися з урахуванням принципів "безпека за замовчуванням" (safety by design), що включає вбудовані захисні механізми.

Навчання та підвищення кваліфікації. Регулярне навчання персоналу з безпечної експлуатації, обслуговування та взаємодії з роботизованими системами є критично важливим.

Чіткі протоколи безпеки. Розробка та суворе дотримання інструкцій з безпеки, процедур блокування та розмітки небезпечних зон.

Моніторинг та аналіз ризиків. Постійний моніторинг функціонування роботизованих систем та оцінка потенційних ризиків для здоров'я та безпеки працівників.

Ергономічна інтеграція. Проектування робочих місць з урахуванням взаємодії людини та машини, забезпечення комфортних та безпечних умов праці.

Психологічна підтримка. Надання психологічної підтримки працівникам, які стикаються з викликами, пов'язаними зі зміною умов праці.

Загалом, роботизація та автоматизація мають потенціал значно покращити безпеку праці, усуваючи працівників від найнебезпечніших та найважчих завдань. Однак, успішна інтеграція цих технологій вимагає всебічного підходу до питань безпеки, постійного аналізу ризиків та інвестицій у навчання та адаптацію персоналу. Таким чином, аналізуючи зміст даного розділу, можна зробити висновок про те, що підвищення ролі і місця служби охорони праці в системі управління підприємством ІТ- галузі та масове впровадження роботизації та автоматизації технологічних процесів на виробництві сприятимуть підвищенню рівня комфорту та безпеки праці фахівців-комп'ютерників.

ВИСНОВКИ

В процесі виконання даної роботи досягнута її мета, виконані наступні сформовані завдання роботи:

- проведено аналіз особливостей біржового трейдингу;
- проведено аналіз проблематики оцінки та прийняття рішень у сфері трейдингу;
- розглянуто перспективні шляхи використання машинного навчання та аналізу даних для трейдингу;
- виконано аналіз аналогів та обґрунтування вибору мови програмування;
- виконано аналіз перспективних моделей підтримки ухвалення торгових рішень на основі машинного навчання;
- проведено опис обраного набору даних для аналізу;
- здійснена розробка структури проєкту;
- виконана розробка програмної реалізації та інтерфейсу модулів системи;
- проведено дослідження та аналіз результатів роботи системи.

Подальший розвиток системи доцільно спрямувати на впровадження більш складних архітектур глибокого навчання, зокрема трансформерів, для підвищення точності багатокрокового прогнозування фінансових часових рядів. Перспективним є також розширення функціоналу за рахунок інтеграції поточкових (real-time) ринкових даних і автоматизації оновлення моделей у режимі безперервного навчання. Окремий напрям розвитку полягає у створенні інтелектуального модуля прийняття рішень, який поєднуватиме результати різних моделей та технічних індикаторів для формування адаптивних торгових стратегій.

ПЕРЕЛІК ПОСИЛАНЬ

1. Аврамчиков В.М. Система підтримки прийняття рішень щодо торгів на фондовій біржі / Фінансовий аналіз: проблеми та рішення. – 2012. – №19. – С. 27-33.
2. Кривокоченко Л.В. Сучасні особливості товарної біржі / Український зовнішньоекономічний вісник. – 2019. – №5. – С. 41-53.
3. Нікульников Н.В., Іваєв М.І., Шевирєва А.Д. Використання систем підтримки прийняття рішень у торговельній компанії / Економіка, підприємництво і право. – 2022. – Том 12. – №11. – С. 3173-3184.
4. Shukla A., Dadhich M., Vaya D., Goel A. Impact of Behavioral Biases on Investors' Stock Trading Decisions: A Comprehensive Quantitative Analysis / Indian Journal Of Science And Technology. – 2024. – №17. – PP. 670-678.
5. Hou X., Luo C., Gao B. An intelligent stock trading decision system based on ensemble classifier through multimodal perturbation / Journal of Intelligent & Fuzzy Systems. – 2024. – PP. 1-19.
6. Rajeswari A.M., Bhatia P., Anushiya S. Fuzzy Logic-Based Outlier Detection Technique for Supporting Stock Market Trading Decision. – 2023.
7. Aramrueng T. & Tangtammaruk P. An Experimental Economic Study of Loss Aversion in Stock Trading Decisions / Humanities and Social Sciences Letters. – 2021. – №9. – PP. 417-429.
8. Gopal V., Veeramani C., Muruganandan S. An effective approach for predicting daily stock trading decisions using fuzzy inference systems / Soft Computing. – 2022. – №28. – PP. 3301–3319.
9. Синицин Є.В., Толмачов А.В. Модель системи підтримки прийняття рішень на фінансових ринках для підприємств на основі ймовірнісного аналізу та машинного навчання / Вісник ОНМУ. – 2019. – Том 18. – №2. – С. 378–393.
10. Cao J., Muhl S., Rieger M., Chen H. Sign Matters: Stock-Movement-Based Trading Decisions of Individual Investors / Journal of Banking & Finance. – 2022. – №148.

11. Aloud M. An Intelligent Stock Trading Decision Support System Using the Genetic Algorithm / International Journal of Decision Support System Technology. – 2020. – №12. – PP. 36-50.
12. Lin Y., Liu S., Yang H., Wu H., Jiang B. Improving stock trading decisions based on pattern recognition using machine learning technology / PLOS ONE. – 2021. – №16.
13. Pathak S., Pawar A., Taware S., Kulkarni S., Akkalkot A. A Survey on Machine Learning Algorithms for Risk-Controlled Algorithmic Trading / International Journal of Scientific Research in Science and Technology. – 2023. – PP. 1069-1089.
14. Ramjan S., Sunkpho J. Principles and Theories of Data Mining With RapidMiner. – 2023. – 516 p.
15. Istiqomah R., Ambarwati R. Optimizing Iron Price Forecasting with Linear Regression Analysis and RapidMiner / Sinkron. – 2024. – №8. – PP. 964-973.
16. Javadpour L. Using RapidMiner for executing queries and visualization in a traditional database course / Journal of Education for Business. – 2021. – №97. – PP. 1-6.
17. Di Martino S., Mazzocca N., Rocco T., Starace L. Mobility Data Analytics with KNOT: The KNime mObility Toolkit. – 2023.
18. Tripathi N., Bhardwaj N., Kumar S., Jain S. A machine learning-based KNIME workflow to predict VEGFR-2 inhibitors / Chemical biology & drug design. – 2023. – №102.
19. Saabith S., Thangarajah V., Fareez M. A Review on Python Libraries and IDEs for Data Science. – 2021.
20. Siva P., Yamaganti Dr., Rohita Dr. A Review on Python for Data Science, Machine Learning and IOT. – 2023.
21. Ranjan M., Barot K., Khairnar V., Rawal V., Pimpalgaonkar A., Saxena S., Sattar A. Python: Empowering Data Science Applications and Research. Journal of Operating Systems Development & Trends. – 2023. – №10. – PP.27-33.
22. Carriage Services, Inc. (CSV) Stock Price, News, Quote & History - Yahoo Finance. URL: <https://finance.yahoo.com/quote/CSV/history/>

23. Jain R., Vanzara R. Emerging Trends in AI-Based Stock Market Prediction: A Comprehensive and Systematic Review / Eng. Proc. – 2023. – №56. – PP. 254.
24. Cui K., Lu X. R-DDQN: Optimizing Algorithmic Trading Strategies Using a Reward Network in a Double DQN / Mathematics. – 2024. – №12(11). – PP. 1621.
25. Ding X., Du Z., Zhang X. Stock Price Forecasting Using Deep Learning Models: A Comparative Study / Expert Systems with Applications. – 2022. – №199. – PP. 117163.
26. Закон України “Про охорону праці” , в редакції Закону № 229-IV від 21.11.2002, ВВР, 2003, № 2, ст.10.

ДОДАТОК А

Лістинг програмного коду

```

#import all libraries
import numpy as np
import pandas as pd
import math
import sklearn
import sklearn.preprocessing
import datetime
import os
import matplotlib.pyplot as plt
import tensorflow as tf

# import dataset
dataset = pd.read_csv('E:\EPAT\RELIANCE.NS.csv', index_col = 0)
df_stock = dataset.copy()
df_stock = df_stock.dropna()
df_stock = df_stock[['Open', 'High', 'Low', 'Close']]

# data scaling (normalizing)
def normalize_data(df):
    min_max_scaler = sklearn.preprocessing.MinMaxScaler()
    df['Open'] =
min_max_scaler.fit_transform(df.Open.values.reshape(-1,1))
    df['High'] =
min_max_scaler.fit_transform(df.High.values.reshape(-1,1))
    df['Low'] = min_max_scaler.fit_transform(df.Low.values.reshape(-
1,1))
    df['Close'] =
min_max_scaler.fit_transform(df['Close'].values.reshape(-1,1))
    return df
df_stock_norm = df_stock.copy()
df_stock_norm = normalize_data(df_stock_norm)

# Splitting the dataset into Train, Valid & test data
valid_set_size_percentage = 10
test_set_size_percentage = 10
seq_len = 20 # taken sequence length as 20
def load_data(stock, seq_len):
    data_raw = stock.as_matrix()
    data = []
    for index in range(len(data_raw) - seq_len):
        data.append(data_raw[index: index + seq_len])
    data = np.array(data);
    valid_set_size =
int(np.round(valid_set_size_percentage/100*data.shape[0]));
    test_set_size =
int(np.round(test_set_size_percentage/100*data.shape[0]));

```

```

    train_set_size = data.shape[0] - (valid_set_size +
test_set_size);
    x_train = data[:train_set_size, :-1, :]
    y_train = data[:train_set_size, -1, :]
    x_valid = data[train_set_size:train_set_size+valid_set_size, :-
1, :]
    y_valid = data[train_set_size:train_set_size+valid_set_size, -
1, :]
    x_test = data[train_set_size+valid_set_size:, :-1, :]
    y_test = data[train_set_size+valid_set_size:, -1, :]
    return [x_train, y_train, x_valid, y_valid, x_test, y_test]

```

```

x_train, y_train, x_valid, y_valid, x_test, y_test =
load_data(df_stock_norm, seq_len)
print('x_train.shape = ', x_train.shape)
print('y_train.shape = ', y_train.shape)
print('x_valid.shape = ', x_valid.shape)
print('y_valid.shape = ', y_valid.shape)
print('x_test.shape = ', x_test.shape)
print('y_test.shape = ', y_test.shape)

```

"""Building the Model"""

```

# parameters & Placeholders
n_steps = seq_len-1
n_inputs = 4
n_neurons = 200
n_outputs = 4
n_layers = 2
learning_rate = 0.001
batch_size = 50
n_epochs = 100
train_set_size = x_train.shape[0]
test_set_size = x_test.shape[0]
tf.reset_default_graph()
X = tf.placeholder(tf.float32, [None, n_steps, n_inputs])
y = tf.placeholder(tf.float32, [None, n_outputs])

# function to get the next batch
index_in_epoch = 0;
perm_array = np.arange(x_train.shape[0])
np.random.shuffle(perm_array)

def get_next_batch(batch_size):
    global index_in_epoch, x_train, perm_array
    start = index_in_epoch
    index_in_epoch += batch_size
    if index_in_epoch > x_train.shape[0]:
        np.random.shuffle(perm_array) # shuffle permutation array
        start = 0 # start next epoch
        index_in_epoch = batch_size
    end = index_in_epoch

```

```

    return x_train[perm_array[start:end]],
    y_train[perm_array[start:end]]

#RNN
layers = [tf.contrib.rnn.BasicRNNCell(num_units=n_neurons,
activation=tf.nn.elu)
          for layer in range(n_layers)]

# LSTM
#layers = [tf.contrib.rnn.BasicLSTMCell(num_units=n_neurons,
activation=tf.nn.elu)
#          for layer in range(n_layers)]

#LSTM with peephole connections
#layers = [tf.contrib.rnn.LSTMCell(num_units=n_neurons,
#                                  activation=tf.nn.leaky_relu,
use_peepholes = True)
#          for layer in range(n_layers)]

#GRU
#layers = [tf.contrib.rnn.GRUCell(num_units=n_neurons,
activation=tf.nn.leaky_relu)
#          for layer in range(n_layers)]

multi_layer_cell = tf.contrib.rnn.MultiRNNCell(layers)
rnn_outputs, states = tf.nn.dynamic_rnn(multi_layer_cell, X,
dtype=tf.float32)
stacked_rnn_outputs = tf.reshape(rnn_outputs, [-1, n_neurons])
stacked_outputs = tf.layers.dense(stacked_rnn_outputs, n_outputs)
outputs = tf.reshape(stacked_outputs, [-1, n_steps, n_outputs])
outputs = outputs[:,n_steps-1,:] # keep only last output of sequence

# Cost function
loss = tf.reduce_mean(tf.square(outputs - y))

#optimizer
optimizer = tf.train.AdamOptimizer(learning_rate=learning_rate)
training_op = optimizer.minimize(loss)

# Fitting the model
with tf.Session() as sess:
    sess.run(tf.global_variables_initializer())
    for iteration in range(int(n_epochs*train_set_size/batch_size)):
        x_batch, y_batch = get_next_batch(batch_size) # fetch the
next training batch
        sess.run(training_op, feed_dict={X: x_batch, y: y_batch})
        if iteration % int(5*train_set_size/batch_size) == 0:
            mse_train = loss.eval(feed_dict={X: x_train, y:
y_train})
            mse_valid = loss.eval(feed_dict={X: x_valid, y:
y_valid})
            print('%.2f epochs: MSE train/valid = %.6f/%.6f'%(
iteration*batch_size/train_set_size, mse_train,
mse_valid))

```

```

# Predictions
y_test_pred = sess.run(outputs, feed_dict={X: x_test})

#checking prediction output nos
y_test_pred.shape

# plotting the graph
comp =
pd.DataFrame({'Column1':y_test[:,3],'Column2':y_test_pred[:,3]})
plt.figure(figsize=(10,5))
plt.plot(comp['Column1'], color='blue', label='Target')
plt.plot(comp['Column2'], color='black', label='Prediction')
plt.legend()
plt.show()

# -*- coding: utf-8 -*-
"""
Created on Sat Sep  1 20:22:56 2018

@author: Umesh
"""

# Importing Libraries
import tensorflow as tf
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
import matplotlib.pyplot as plt
import talib
import random

# Importing dataset
dataset = pd.read_csv('TATAMOTORS.NS.csv')
dataset = dataset.dropna()
dataset = dataset[['Open', 'High', 'Low', 'Close']]

#Preparing the dataset
dataset['H-L'] = dataset['High'] - dataset['Low']
dataset['O-C'] = dataset['Close'] - dataset['Open']
dataset['3day MA'] = dataset['Close'].shift(1).rolling(window =
3).mean()
dataset['10day MA'] = dataset['Close'].shift(1).rolling(window =
10).mean()
dataset['30day MA'] = dataset['Close'].shift(1).rolling(window =
30).mean()
dataset['Std_dev'] = dataset['Close'].rolling(5).std()
dataset['RSI'] = talib.RSI(dataset['Close'].values, timeperiod = 9)
dataset['Williams %R'] = talib.WILLR(dataset['High'].values,
dataset['Low'].values, dataset['Close'].values, 7)

dataset['Price_Rise'] = np.where(dataset['Close'].shift(-1) >
dataset['Close'], 1, 0)

```

```

dataset = dataset.dropna()

data = dataset.iloc[:, 4:]

# Dimensions of data
n = data.shape[0]
p = data.shape[1]
# Make data a np.array
data = data.values

# Splitting the dataset- Training and test data
train_start = 0
train_end = int(np.floor(0.8*n))
test_start = train_end + 1
test_end = n
data_train = data[np.arange(train_start, train_end), :]
data_test = data[np.arange(test_start, test_end), :]

# Scale data
scaler = MinMaxScaler(feature_range=(-1, 1))
scaler.fit(data_train)
data_train = scaler.transform(data_train)
data_test = scaler.transform(data_test)

# Build X and y
X_train = data_train[:, 0:-1]
y_train = data_train[:, -1]
X_test = data_test[:, 0:-1]
y_test = data_test[:, -1]

# Building the Artificial Neural Network:
# Number of features in training data
n_features = X_train.shape[1]

# Neurons
n_neurons_1 = 512
n_neurons_2 = 256
n_neurons_3 = 128

# Session
net = tf.InteractiveSession()

# Placeholder
X = tf.placeholder(dtype=tf.float32, shape=[None, n_features])
Y = tf.placeholder(dtype=tf.float32, shape=[None])

# Initializers
sigma = 1
weight_initializer = tf.variance_scaling_initializer(mode="fan_avg",
distribution="uniform", scale=sigma)
bias_initializer = tf.zeros_initializer()

# Hidden weights:

```

```

#Layer 1: Variables for hidden weights and biases
W_hidden_1 = tf.Variable(weight_initializer([n_features,
n_neurons_1]))
bias_hidden_1 = tf.Variable(bias_initializer([n_neurons_1]))
#Layer 2: Variables for hidden weights and biases
W_hidden_2 = tf.Variable(weight_initializer([n_neurons_1,
n_neurons_2]))
bias_hidden_2 = tf.Variable(bias_initializer([n_neurons_2]))
#Layer 3: Variables for hidden weights and biases
W_hidden_3 = tf.Variable(weight_initializer([n_neurons_2,
n_neurons_3]))
bias_hidden_3 = tf.Variable(bias_initializer([n_neurons_3]))

# Output weights:
#Output layer: Variables for output weights and biases
W_out = tf.Variable(weight_initializer([n_neurons_3, 1]))
bias_out = tf.Variable(bias_initializer([1]))

# Hidden layer
hidden_1 = tf.nn.relu(tf.add(tf.matmul(X, W_hidden_1),
bias_hidden_1))
hidden_2 = tf.nn.relu(tf.add(tf.matmul(hidden_1, W_hidden_2),
bias_hidden_2))
hidden_3 = tf.nn.relu(tf.add(tf.matmul(hidden_2, W_hidden_3),
bias_hidden_3))

# Output layer (transpose!)
out = tf.transpose(tf.add(tf.matmul(hidden_3, W_out), bias_out))

# Cost function
mse = tf.reduce_mean(tf.squared_difference(out, Y))

# Optimizer
opt = tf.train.AdamOptimizer().minimize(mse)

# Run initializer
net.run(tf.global_variables_initializer())

# Fitting the neural network
batch_size = 100
epochs = 10
# Run
for e in range(epochs):
    # Shuffle training data
    shuffle_data = np.random.permutation(np.arange(len(y_train)))
    X_train = X_train[shuffle_data]
    y_train = y_train[shuffle_data]
    # Minibatch training
    for i in range(0, len(y_train) // batch_size):
        start = i * batch_size
        batch_x = X_train[start:start + batch_size]
        batch_y = y_train[start:start + batch_size]
        # Run optimizer with batch

```

```

net.run(opt, feed_dict={X: batch_x, Y: batch_y})

#Predicting the movement of the stock
pred = net.run(out, feed_dict={X: X_test})
y_pred = pred[0]
y_pred = pred[0] > 0.5

dataset['y_pred'] = np.NaN
dataset.iloc[(len(dataset) - len(y_pred)):-1:] = y_pred
trade_dataset = dataset.dropna()

#Computing Strategy Returns
trade_dataset['Tomorrows Returns'] = 0.
trade_dataset['Tomorrows Returns'] =
np.log(trade_dataset['Close']/trade_dataset['Close'].shift(1))
trade_dataset['Tomorrows Returns'] = trade_dataset['Tomorrows
Returns'].shift(-1)

trade_dataset['Strategy Returns'] = 0.
trade_dataset['Strategy Returns'] = np.where(trade_dataset['y_pred']
== True,
        trade_dataset['Tomorrows Returns'], -
trade_dataset['Tomorrows Returns'])

trade_dataset['Cumulative Market Returns'] =
np.cumsum(trade_dataset['Tomorrows Returns'])
trade_dataset['Cumulative Strategy Returns'] =
np.cumsum(trade_dataset['Strategy Returns'])

#Plotting the graph of returns
plt.figure(figsize=(10,5))
plt.plot(trade_dataset['Cumulative Market Returns'], color='r',
label='Market Returns')
plt.plot(trade_dataset['Cumulative Strategy Returns'], color='g',
label='Strategy Returns')
plt.legend()
plt.show()

def img_to_encoding(image, model):
    image = cv2.resize(image, (96, 96))
    img = image[...,::-1]
    img = np.around(np.transpose(img, (2,0,1))/255.0, decimals=12)
    x_train = np.array([img])
    embedding = model.predict_on_batch(x_train)
    return embedding

```

StrikePlagiarism.com

Звіт не був оцінений

Метадані

ДОКУМЕНТ

Заголовок

2026 КвРБак - Зінчук

Автор

Зінчук Владислав Валентинович

Науковий керівник / Експерт

С.М. Коновалов

ІД документа

334346312

ОРГАНІЗАЦІЯ

Назва організації

Odesa National Maritime University

підрозділ

Технічна кібернетика й інформаційні технології ім. професора Р.В. Мерктя

ЗВІТ

Дата звіту

6/17/2026

Дата редагування

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до заголовного обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

10.95%

10.95%

КП 1

25

Діагональ фраз для коефіцієнта подібності 1

4.03%

4.03%

КП 2

17946

Кількість слів

0.33%

0.33%

КЦ

142950

Кількість словосполучень

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		0
Інтервали		0
Мікропробіли		0
Білі знаки		0
Парафрази (SmartMarks)		106

Джерела

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Копії тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз

Копії тексту

#	НАЗВА ТА АДРЕСА ДЖЕРЕЛА (URL, НАЗВА БАЗИ)	кількість дублюваних слів (фрагментів)
---	---	--