

Київський національний університет імені Тараса Шевченка

Економічний факультет

Кафедра економічної кібернетики

**КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА
РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПРОГНОЗУВАННЯ ТА
АНАЛІЗУ РИНКУ FOREX**

студента 2 курсу магістратури
спеціальності 051 «Економіка»
ОНП «Економічна кібернетика»
денної форми навчання
Тоцького Владислава Віталійовича

Науковий керівник:

к.е.н., доцент

Шпирко Віктор Васильович

Засвідчую, що в цій роботі немає
запозичень із праць інших авторів
без відповідних посилань



Студент

(підпис)

Роботу допущено до захисту перед ЕК
рішенням кафедри економічної кібернетики
від «4» травня 2022 р., протокол № 13.

Завідувач кафедри:

доктор економічних наук, професор

Ляшенко Олена Ігорівна

(підпис)

РЕФЕРАТ

Кваліфікаційна робота магістра містить: 43 ст., 12 рис., 2 табл., 31 джерело, 19 фр. коду;

Ключові слова: Forex, Python, LSTM, валютний ринок, часові ряди;

Об'єкт дослідження: валютний ринок Forex;

Мета дослідження: удосконалення теоретичних основ та розробка практичних методів, які можуть використовуватися не тільки у сфері аналізу ринку Forex, але й застосування цих методів в інших сферах аналізу та прогнозування.

Методи дослідження: абстрагування і конкретизація, наукова абстракція, системний підхід, статистичний та графічний.

Наукова новизна, теоретична значимість дослідження: застосування нейронних мереж у прогнозуванні валютних пар.

Практична цінність: використання ефективних алгоритмів для автоматичної торгівлі на Forex.

RESUME

Taras Shevchenko National University of Kyiv,

Faculty of Economics, Department of Economic Cybernetics

Key words: Forex, Python, LSTM, currency market, time series.

The graduation research of student Vladyslav Totskiy deals with Forex research and forecasting.

The work is interesting for market traders, analysts, time series scientists

Pages 43, tables 2, bibliog. 31, code snippets 19.

Розробка інформаційної системи для прогнозування та аналізу ринку Forex

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ СУЧАСНОГО ВАЛЮТНОГО РИНКУ	7
19.1..... Св ітовий валютний ринок FOREX та його учасники	7
19.2..... Фак тори, які впливають на валютні курси	11
РОЗДІЛ 2 МЕТОДИКИ ПРОГНОЗУВАННЯ НА ВАЛЮТНОМУ РИНКУ	14
2.1 Адаптивні методи прогнозування часових рядів.....	14
2.2 FOREX-торгівля за допомогою методу опорних векторів (SVM)	20
РОЗДІЛ 3 ВИКОРИСТАННЯ СУЧАСНИХ ПРАКТИК АВТОМАТИЗОВАНОЇ ТОРГІВЛІ ДЛЯ ПРОГНОЗУВАННЯ ТА АНАЛІЗУ РИНКУ FOREX	26
3.1 Розробка архітектури клієнтської та серверної частини ПЗ.....	26
3.2 API інтеграція взаємодії з клієнтським торговим терміналом та розробка алгоритму аналізу	34
ВИСНОВКИ	41
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	44

ВСТУП

Актуальність. Глобальна пандемія, яка розпочала нову еру віддаленої роботи, продемонструвала, що люди, у зв'язку із втратою основного місця роботи, пошуку додаткового заробітку та додаткового способу інвестування, почали виходити із зони комфортного, традиційного інвестування коштів та використовувати сучасні методи збільшення своїх доходів. Одним із них, є інтернет-трейдинг, спосіб доступу до торгів на валютній, фондовій або товарній біржі з використанням Інтернету як засобу зв'язку. Даний вид трейдингу дозволяє керувати безпосередньо рахунками та торгівлею в режимі онлайн, що дозволяє виділити більше часу на прийняття рішень, відштовхуючись від технічного або (і) фундаментального аналізу ринку.

Завдяки легкодоступності хмарних сервісів та доступів до потужних комп'ютерів, такий вид діяльності як інтернет-трейдинг став більш доступний для багатьох людей сьогодні. Використання технологій TensorFlow [12] і PyTorch [12], які розроблені для дуже легкого використання штучного інтелекту для аналізу даних, дає змогу інтегрувати різноманітні типи аналізу та прогнозування дуже швидко і якісно.

Наявність високорівневих мов програмування, на кшталт Python, дає змогу людям, які не мають досвід у програмуванні, використовувати такі мови для створення базових алгоритмів та вирішення складних розрахункових завдань.

Розробка додатку, який дозволить витратити менше часу на технічні аспекти аналізу, відкриє очі на не менш важливі фактори, які в результаті їх вивчення можуть бути урегульовані та враховані в подальшому прийнятті рішень.

Серед досліджень присвячених аналізу ринку Forex та прогнозування часових рядів існують різні напрями, які безпосередньо стосуються мети наукової роботи, її завдань, предмета та об'єкта. До авторів робіт, які безпосередньо досліджували аналіз ринку Forex та прогнозування часових рядів в різних країнах світу, в тому числі і в Україні можна віднести наступних вчених: Черняк О.І. [2], Ставицький А.В.[2], Ходаківська В. П. [3], Данілов О. Д. [3], А.

Mittal [5], B. Artley [8], J. Chen [13], T. Segal [17], R. Akter [19], R. Sarker [20], S. Elsworht [21], S. Natakarnkitkul [22], Y. Ensafi [28], S. Shojaeipour [30].

Метою роботи є удосконалення теоретичних основ та розробка практичних методів, які можуть використовуватися не тільки у сфері аналізу ринку Forex, але й застосування цих методів в інших сферах аналізу та прогнозування.

Для досягнення мети дипломної роботи були поставлені наступні **завдання**:

- уточнити зміст категорій пов'язаних з Forex та його учасниками, розкрити зміст та теоретичні аспекти діяльності Forex в загальному;
- визначити фактори, які впливають на валютні курси, та які показники в практичному значенні можуть покращити результати аналізу;
- розкрити визначення і сутність адаптивних методів прогнозування часових рядів;
- розкрити теоретичний аспект прогнозування за допомогою методу опорних векторів;
- визначити переваги використання нейронної мережі LSTM для аналізу часових рядів;
- розробити архітектуру інформаційної системи;
- розробити API-інтеграцію з веб та алгоритм для прогнозування цін валютних пар.

Об'єктом дослідження є валютний ринок Forex.

Предметом дослідження є теоретичні, практичні основи та сучасні тенденції методологічного використання технологій для аналізу валютних курсів.

При написанні роботи використовувалися наступні **методи**: абстрагування і конкретизація - при уточненні підходів до визначення валютного ринку Forex та їх учасників; наукова абстракція - при розкритті факторів та їх вплив на валютний курс; системний підхід - при виокремленні переваг адаптивних методів; статистичний та графічний - при визначенні переваг використання LSTM; гнучка

розробка - при розробці основних архітектурних основ додатку; глибоке навчання - при розробці алгоритму LSTM для автоматичної торгівлі.

Наукова та практична новизна роботи. Розроблене програмне забезпечення дає змогу трейдеру управляти торгівлею на валютному ринку в автоматичному режимі, зазначивши декілька гіперпараметрів. Багато подібних алгоритмів у сфері нейронних мереж, які існують на ринку, не мають такої гнучкості порівняно з розробленим програмним забезпеченням у цій роботі. Адаптивність дозволяє вносити інвестору свої корекції в значення параметрів, базуючись на результатах своїх досліджень, наприклад, у фундаментальному аналізі.

Результати впровадження роботи:

1. International conference 16-17 December 2020 Crises and challenges for national and world economies: economic and cultural implications. Vladislav Totkiy. K-means clustering of online delivery service users in a metropolis. URL: http://irn.center/wp-content/uploads/2020/11/Vladislav-Totkiy_Ukraine.pdf.
2. The 1st International scientific and practical conference “Modern research in world science” (April 17-19, 2022) SPC “Sci-conf.com.ua”, Lviv, Ukraine. 2022. 1380 p. Тоцький В. В. Forex-торгівля з використанням методів машинного навчання. URL: <https://sci-conf.com.ua/wp-content/uploads/2022/04/MODERN-RESEARCH-IN-WORLD-SCIENCE-17-19.04.22.pdf>.

Інформаційну базу дослідження становили статистичні матеріали Statistics [6], статистичні та теоретичні матеріали Nasdaq [9, 26], теоретичні матеріали Sciencedirect [10], Investopedia [11, 13, 17] та відповідні наукові статті вітчизняних та зарубіжних вчених.

Дана робота має таку **структуру**: вступ, три розділи, висновки, список використаних джерел.

РОЗДІЛ 1

ТЕОРЕТИЧНІ АСПЕКТИ СУЧАСНОГО ВАЛЮТНОГО РИНКУ

1.1 Світовий валютний ринок FOREX та його учасники

Валютний ринок (Forex) — це глобальний децентралізований ринок для торгівлі валютами. Ринок визначає курс для кожної валюти по відношенню до іноземної валюти та включає такі ключові операції, як продаж та обмін за поточними або встановленими цінами. Forex є найбільшим фінансовим ринком фінансового світу, і, відповідно, найліквідніший серед всіх інших ринків. У світі існує 180 різних видів офіційних валют, однак основні валютні пари, які торгуються на ринку це EUR/USD. Валютні ринки можна розглядати як зв'язок між банками, небанківськими дилерами, форекс-дилерами та брокерами, які пов'язані через мережу комп'ютерних терміналів та автоматизованих дилерських систем.

Ринок Forex складається з трьох основних торгових сегментів: Європи, Північної Америки та Австралазії. Австралазія включає основні торгові центри Бахрейну, Сіднея, Токію, Гонконгу та Сінгапуру. До Європи належать Цюрих, Франкфурт, Париж, Брюссель, Лондон та Амстердам. Регіон Північної Америки включає Нью-Йорк, Монреаль, Торонто, Чикаго, Сан-Франциско та Лос-Анджелес [10].

Для окремих покупців і продавців валюти було б незручно шукати один одного. Отже, валютний ринок розвинувся як посередник. Це найбільший фінансовий ринок у світі, де ціни змінюються, а валюти торгуються десь щогодини кожного робочого дня. Різні функції валютного ринку полягають у наступному:

1. Функція переказу. Основною і найбільш очевидною функцією валютного ринку є переказ коштів або іноземних валют з однієї країни в іншу для здійснення платежів. Ринок в основному конвертує одну валюту в іншу.

2. Кредитна функція. FOREX надає короткострокові кредити імпортерам, щоб полегшити безперебійний потік товарів і послуг з різних країн. Імпортер може використовувати власний кредит для фінансування іноземних закупівель.
3. Функція хеджування. Третя функція валютного ринку полягає в хеджуванні валютних ризиків. Сторони в іноземній валюті часто бояться коливання курсів, тобто ціни однієї валюти в перерахунку на іншу. Це може призвести до виграшу або збитку для зацікавленої сторони [18].

Учасники ринку Forex використовують валюту для хеджування міжнародного валютного та процентного ризику, для спекуляцій на геополітичних подіях та для диверсифікації портфелів. Розуміння поведінки ринку та ефективний аналіз події фундаментальної складової аналізу вимагає розуміння мотивів учасників валютного ринку. Деякі з учасників мають більш доступну інформацію та можуть за рахунок цього суттєво впливати на різні події на ринку, а деякі навпаки – обмежену інформацію, і не мати значного впливу на ринок (табл. 1.1).

Таблиця 1.1

Учасники валютного ринку Forex

Учасник ринку	Визначення
Дилер (Forex Dealer)	це особа або фірма, яка купує іноземну валюту в одного учасника а продає іншому, реалізуючи різницю між купівлею та продажем, який називається спредом [9].
Брокери (Brokers)	як правило фірма, яка допомагає отримати найкращу пропозицію. Наприклад, брокери можуть допомагати клієнтам отримати найнижчу ціну купівлі або найвищу ціну продажу, надаючи доступні пропозиції від кількох дилерів. Також важливою причиною використати послуги брокера є анонімність [11].
Хеджери (Hedgers)	це учасник ринку, який страхує свої валютні активи. Іншими словами, хеджер — це інвестор, тобто учасник ринку, що уникає ризику і готовий платити за це [3].
Спекулянти (Speculators)	це клас трейдерів, які не мають конкретних вимог від валют. Вони здійснюють свою діяльність лише купівлею і продажем валюти у короткостроковий час з надією отримати прибуток від даних операцій [26].
Арбітражники (Arbitrageurs)	це учасники, які отримують прибуток від купівлі валюти на одному ринку за нижчою ціною та її продажу на іншому ринку за вищою ціною. Розбіжність цін є їхнім прибутком [17].

Продовження табл. 1.1

Центробанки	Центральні банки всіх країн первиною мірою беруть участь на ринку валют. Вони можуть брати участь відкрито тобто офіційно, та неофіційно. Це пояснюється тим що банк має свій цільовий діапазон, в межах якого вони б хотіли бачити коливання своєї валюти. Коли ціна випадає з діапазону, банк проводить операції офіційно, щоб повернути курс в діапазон. Також банки беруть участь на ринку для захисту своїх валют від спекулятивних атак [17].
-------------	---

Джерело: складено автором на основі [3, 9, 11, 17, 26].

Крім перерахованих важливих учасників ринку у табл. 1.2, є сектор учасників роздрібних торгів. Наприклад, людина, яка поїхала за кордон і надає попит на валюту країни, куди вона приїхала, тому що їй необхідно купувати товари або послуги саме в обмін на державну валюту. Другий приклад це компанії які займаються зовнішньою торгівлею, теж є не менш важливими учасниками валютного ринку. Дилери мають найбільший вплив, тому що вони володіють максимальною інформацією на ринку, потім за ними йдуть брокери, та інші учасники Forex.

Поширені способи проводити операції з валютами на Forex - це спотові, форвардні операції та опціонні контракти. Останній із перерахованих буде протестований у практичній частині роботи за допомогою торговельного терміналу.

На ринку Forex домінує не єдина біржа, а глобальна мережа комп'ютерів і брокерів з усього світу. Forex-брокери також виступають в ролі маркет-мейкерів і можуть публікувати ставки та ціни на валютну пару, яка відрізняється від найбільш конкурентоспроможної пропозиції на ринку.

Існує три типи валютного ринку:

- Спотовий ринок Forex — це негайний обмін валюти між покупцями та продавцями за поточним обмінним курсом. Спотовий ринок складає більшу частину торгівлі валютою. Основними учасниками спотового ринку є комерційні, інвестиційні та центральні банки, а також дилери, брокери та спекулянти. Великі комерційні та інвестиційні банки становлять основну частину спотових торгів, торгуючи не лише для себе, а й для своїх клієнтів.

- Форвардний ринок Forex. На форвардних ринках дві сторони погоджуються торгувати валютою за встановлену ціну та кількість на певну дату в майбутньому. Двома сторонами можуть бути компанії, окремі фізичні особи, уряди тощо. Форвардні ринки корисні для хеджування. З іншого боку, форвардні ринки не мають централізованої торгівлі і є відносно неліквідними (оскільки є лише дві сторони).
- Ф'ючерсний ринок Forex. Такі ринки подібні до форвардних ринків з точки зору основної функції. Однак велика різниця в тому, що ф'ючерси використовують централізовані біржі. Завдяки централізованим біржам немає ризиків контрагентів для жодної зі сторін. Це допомагає забезпечити високу ліквідність ф'ючерсних ринків, особливо в порівнянні з форвардними ринками [13].

Валютний ринок став прибутковою можливістю для людей із фінансовим досвідом. З низькими вимогами до вступу та ринками, які відкриті 24/7, кожен, хто має ноутбук або смартфон, потенційно може отримати великі прибутки на ринках валют. Однак ці можливості також мають високе кредитне плече і високий ризик та інший ряд недоліків (табл. 1.2). Кожен, хто намагається зробити свою діяльність на ринку прибутковою, потребує суворої дисципліни та навичок, щоб досягти успіху.

Таблиця 1.2

Переваги на недоліки ринку Forex

Переваги	Недоліки
Варіантність торгівлі	Відсутність регулювання
Невеликі трансакційні витрати	Високий ризик, високе кредитне плече
Висока ліквідність за рахунок щоденної торгівлі 24/7	Операційний ризик, який важко контролювати цілодобово
Підходить як для короткострокового, так і довгострокового стилю торгівлі	Комплексний процес визначення ціни за рахунок впливу численних факторів
За розумного розміщення валюти волатильність може забезпечувати високі прибутки	Висока волатильність
Легкість входу та низькі вимоги до капіталу	

Джерело: складено автором на основі [4, 24, 27].

Отже, валютний ринок (також відомий як FX або Forex) є глобальним ринком для обміну національних валют одна на одну. Валютний ринок є найбільшим фінансовим ринком у світі. Банки, комерційні компанії, хедж-фонди, центральні банки та окремі спекулянти беруть участь у ньому та щодня обмінюють валюти як для спекулятивних, так і для хеджувальних цілей.

1.2 Фактори, які впливають на валютні курси

Валютні курси є важливим засобом вимірювання економічного стану країни та чудовим способом оцінити придатність економіки для розширення бізнесу. Обмінний курс визначається якою кількістю можна обміняти валюту на одиницю іншої валюти. Як звичайний ринок, курс може змінюватися в залежності від попиту та пропозиції цих країн на валюту, але є основні категорії факторів на які звертають учасники ринку, аналітики та інвестори.

Темпи інфляції.

Інфляція тісно пов'язана з процентними ставками, які можуть впливати на обмінні курси. Країни намагаються збалансувати процентні ставки та інфляцію, але взаємозв'язок між ними складний і часто важко керований.

Низькі процентні ставки стимулюють споживачів витратити та стимулюють економічне зростання. Якщо споживчі витрати спричиняють підвищення попиту, то виникає інфляція, але це не обов'язково є поганим сценарієм. При цьому вищі процентні ставки, як правило, залучають іноземні інвестиції, що, ймовірно підвищить попит на валюту країни. Кінцевим визначенням вартості та обмінного курсу національної валюти є очікувана бажаність володіння валютою цієї країни. На це сприйняття впливає низка економічних факторів, таких як стабільність уряду та економіка країни. Важливим фактором в аналізі інвестора є безпека утримання готівкових активів у валюті країни інвестування [30].

Процентні ставки.

Зміни процентної ставки впливають на вартість валюти та курс долара до національної валюти. Валютні ставки, процентні ставки та інфляція – все це

корелює між собою. Підвищення процентних ставок спричиняє підвищення курсу валюти країни, оскільки вищі відсоткові ставки забезпечують вищі ставки для кредиторів, тим самим залучаючи більше іноземного капіталу, що спричиняє підвищення обмінних курсів [31].

Поточний рахунок країни (бюджет).

Рахунок поточних операцій платіжного балансу включає сальдо торгівлі товарами та послугами плюс чисті інвестиційні доходи від закордонних активів та чистих трансфертів. Дефіцит зазвичай є результатом збільшення чистого торгового дефіциту, коли вартість імпорту перевищує вартість експорту. В результаті відбудеться чистий відтік грошей з кругового потоку країни. Підприємства платять за імпорт у власній валюті, але вона в кінцевому підсумку конвертується у валюту країни-експортера. Отже, зростання дефіциту рахунку поточних операцій призводить до збільшення пропозиції національної валюти на валютних ринках. Тому на валютному ринку відбудеться зміщення пропозиції назовні. Це – за інших рівних умов – може призвести до падіння зовнішньої вартості валюти.

Державний борг.

Борг, що належить центральному уряду. Така країна має менше шансів на придбання іноземного капіталу, що призведе до підвищення темпу інфляції. Іноземні інвестори продають свої облігації на відкритому ринку, якщо ринок прогнозує державний борг певної країни. Як наслідок, зниження вартості національної валюти. Динаміка обмінного курсу важлива для країн, що розвиваються, які не мають міцних валют і мають значну частину державного боргу, номінованого в доларах США [16].

Спекуляція.

Якщо очікується зростання вартості валюти в країні, інвестори вимагатимуть кількісне збільшення в обороті, в результаті збільшення вартості національної валюти.

Валютне макроекономічне моделювання визначається взаємодією людей як учасників ринку (попит та пропозиція). Валютна модель є різновидом ринкової моделі. Не дивлячись на фактори існує багато моделей які вказують, що рух ціни валюти часто є випадковим. Не варто очікувати, що моделі будуть мати велику прогнозну силу. Тому для моделювання валютних курсів, найчастіше використовуються методи машинного навчання, які показують високі результати прогнозу, за рахунок аналізу історичних курсів. Саме вивчення поведінки курсу в минулому дають змогу визначити адекватні прогнозні значення [19].

Отже, існує дуже багато факторів, які впливають на обмінні курси. Найбільш важливими, що виділені в роботі є: темпи інфляції, процентні ставки, поточний рахунок країни, державний борг та спекуляція. Ці фактори змінюють графік попиту та пропозиції та створюють новий обмінний курс у новому рівноважному стані. Але існують і деякі інші фактори, такі як політична стабільність, умови торгівлі, ринкові рішення тощо, які також відіграють істотну роль у визначенні стану попиту на валюту та визначенні зміни курсу іноземної валюти.

РОЗДІЛ 2

МЕТОДИКИ ПРОГНОЗУВАННЯ НА ВАЛЮТНОМУ РИНКУ

2.1 Адаптивні методи прогнозування часових рядів

Моделі часових рядів належать до типу параметричних моделей. Перш ніж параметричні моделі використовуються для прогнозування, необхідно мати початковий набір даних для оцінки параметрів. Умовно більшість параметричних моделей реалізуються статично. В даному контексті “статичність” означає, що моделі не використовують інформацію з вхідного потоку даних для оновлення параметрів моделі. Крок оцінки забезпечує фіксований набір значень параметрів, які використовуються для процесу прогнозування. У цій структурі параметри оновлюються лише під час періодичної переоцінки. Однак обчислювальне навантаження буде збільшуватись за геометричною прогресією в залежності від параметрів при навчанні, та частоти перерахування. Щоб забезпечити обчислювальну адаптивність існують методи адаптивної фільтрації з метою автоматизації оцінки та оновлення параметрів. Як мінімум два методи потребують детального дослідження: рекурсивні найменші квадрати (RLS) і найменші середні квадрати (LMS). Головні питання полягає в тому, чи можуть такі методи забезпечити в цілому найбільшу продуктивність.

ARMA.

Модель ARMA використовує дані часових рядів і статистичний аналіз для інтерпретації даних і прогнозування. ARMA має на меті пояснити дані, використовуючи дані часових рядів про їх історичні значення, і використовує лінійну регресію для прогнозування.

Двома основними будівельними блоками одновимірної моделі часового ряду ARMA для стаціонарного процесу є компонент авторегресії AR і компонент ковзного середнього MA. У моделі суто AR є функцією його минулих спостережень лише до деякого кінцевого авторегресійного моменту. А процеси MA можна розглядати як рекурсивні фільтри згладжування з експоненціально

спадаючими залежностями від минулих спостережень. Більш загальна модель авторегресійної ковзної середньої (ARMA) містить ці два компоненти. У математичній формі ці моделі виражаються з одним гіперпараметром AR частини — $AR(p)$ (формула 2.1) [7].

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + \epsilon_t \quad (2.1)$$

Модель ковзного середнього параметру q — $MA(q)$ (формула 2.2):

$$x_t = \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q} \quad (2.2)$$

Модель порядку авторегресійної ковзної середньої (p, q) — $ARMA(p, q)$:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q}, \quad (2.3)$$

де x — спостережні дані,

e — помилка,

ϕ, θ — ARMA параметри.

Порядок таких моделей визначається найвищим лагом із запізненням.

Використання оператора зворотного зсуву — B (визначається як $Bx_t = x_{t-1}$, $BSx_t = x_{t-2}$), тоді ці моделі можна переписати таким чином (формула 2.4):

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)x_t = \epsilon_t, \quad (2.4)$$

або:

$$\phi(B)x_t = \epsilon_t, \quad (2.5)$$

$$\text{де } x_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)\epsilon_t, \quad (2.6)$$

або:

$$x_t = \theta(B)\epsilon_t \quad (2.6)$$

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)x_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)\epsilon_t, \quad (2.7)$$

або:

$$\phi(B)x_t = \theta(B)\epsilon_t. \quad (2.8)$$

Модель ARMA можна застосувати лише до слабо стаціонарного часового ряду. Ряд називається слабо стаціонарним, якщо його середнє є постійним, а коваріація залежить лише від затримки між корельованими значеннями ряду, а не від часу (рис. 2.1, рис 2.2) [6].

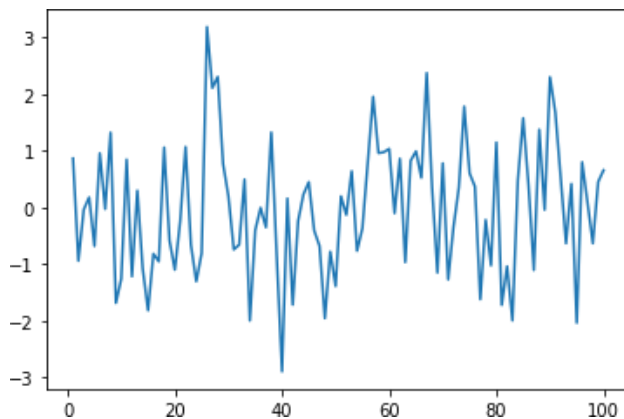


Рис 2.1. Нестационарний процес часового ряду

Джерело: розрахунки автора.

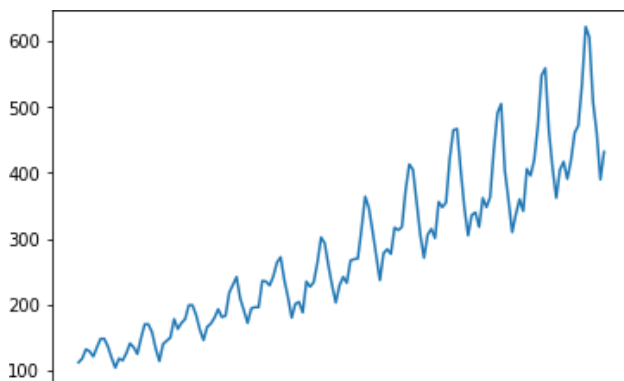


Рис 2.2 Стаціонарний процес часового ряду

Джерело: розрахунки автора.

Найпоширенішим методом для керування стаціонарністю є ARIMA (autoregressive integrated moving average model). Порядок диференційності позначається d . Більшість реальних несезонних рядів можна зробити стаціонарними за допомогою параметру d (формула 2.9).

Модель ARIMA (p, d, q):

$$\phi(B)(1 - B)^d x_t = \theta(B)\epsilon_t \quad (2.9)$$

Іноді часовий ряд включає проміжки, що повторюються через фіксований період. Цю характеристику називають сезонністю. Сезонний часовий ряд можна моделювати за допомогою використання сезонних диференцій і сезонних параметрів AR і MA, якщо це необхідно. Така модель називається сезонною

авторегресійною інтегрованою ковзною середньою (SARIMA) (формула 2.10) [8].

SARIMA (p, d, q) (P, D, Q) S визначається :

$$\phi(B)\Phi(B^S)(1 - B)^d(1 - B^S)^Dx_t = \theta(B)\Theta(B^S)\epsilon_t, \quad (2.10)$$

де $\Phi(B^S)$, $(1 - B^S)^D$ і $\Theta(B^S)$ позначають сезонність авторегресії, та сезонні поліноми ковзного середнього.

Адаптивна сезонна оцінка ARIMA використовує методи нелінійної оптимізації для максимізації функції імовірності (або, точніше, мінімізації від'ємного значення логарифмічної функції правдоподібності). Як згадувалося вище, як тільки параметри були визначені за допомогою набору навчальних даних, вони залишаються статичними, доки оцінка моделі не буде повторена для нових навчальних даних. Основна ідея адаптивних фільтрів полягає в тому, щоб оновлювати параметри з часом, щоб на певних етапах модель удосконалювалась за рахунок нових, що важливо, фактичних даних [2].

Дослідження нових методів адаптивного прогнозування виникло через потреби в наступних перевагах:

- можливість оновлення прогнозної спроможності на основі нових даних;
- точність прогнозів моделей машинного навчання;
- можливість обробити викиди в даних;
- позбутися потреби у ручній оцінці параметрів.

Перед розглядом фільтра RLS потрібно з'ясувати як працює фільтр Калмана. Це алгоритм який забезпечує ефективне рекурсивне рішення задачі найменших квадратів. Цей фільтр зберігає інформацію про середнє та коваріацію вектора стану на основі всіх спостережень до прогнозного моменту t . Після кожного нового спостереження середнє значення та коваріація оновлюється. Ідея Калмана полягає в тому, щоб отримати найкраще наближення до істинної координати вектору даних, шляхом згладжування шумів (формула 2.11) [14].

$$K_t = \frac{P_{tvt-1}Z_t^F}{Z_t P_{tvt-1}Z_t^F + H}, \quad (2.11)$$

де K_t — Коефіцієнт Калмана,

P_t — оцінка коваріації в момент t .

Підвищення коефіцієнт Калмана можна розглядати як вагу, надану останньому спостереженню щодо оновлення середнього і коваріаційного стану. Два рівняння оновлюють вектор і матрицю коваріації після спостереження в момент часу t .

Формула оновлення коваріації:

$$P_t = P_{tvt-1} - K_t Z_t P_{tvt-1} \quad (2.12)$$

Фільтр RLS може бути інтерпретований як окремий випадок більш загального алгоритма Калмана. Різниця між ними полягає в обробці процесу стану. RSL фільтрує стан моделі як детермінований процес, коли Калман розглядає випадкові процеси [22].

Коваріаційний прогноз:

$$P_{tvt-1} = \frac{P_{t-1}}{\lambda} \quad (2.13)$$

Розрахунок поетапного приросту:

$$K_t = \frac{P_{tvt-1}Z_t^F}{Z_t P_{tvt-1}Z_t^F} = \frac{P_{tvt-1}Z_t^F}{Z_t P_{t-1}Z_t^F + \lambda} \quad (2.14)$$

Коефіцієнт λ визначає, наскільки швидко зменшуються ваги. Обернене значення $(1 - \lambda)$ визначає пам'ять фільтра. Простіше кажучи, λ визначає, скільки минулих спостережень потрібно враховувати для оцінки параметра. Для $\lambda = 0,99$ пам'ять дорівнює 100, що означає, що лише останні 100 спостережень відіграють роль у визначенні параметрів. Якщо $\lambda = 1$, то задача зводиться до звичайних найменших квадратів, де пам'ять нескінченна і всім попереднім спостереженням надається однакова вага під час визначення оцінок. Іншими словами, функція вартості, мінімізована фільтром RLS, є експоненціально зваженою сумою квадратів помилок.

Алгоритм RLS можна повністю описати наступними двома моделями:

$$\alpha_t = \alpha_{t-1} + \frac{P_{t-1}Z_t^F}{Z_t^F P_{t-1} Z_t^F + \lambda} \epsilon_t \quad (2.15)$$

$$P_t = \frac{P_{t-1} \frac{P_{t-1} Z_t^F Z_t^F P_{t-1}}{Z_t^F P_{t-1} Z_t^F + \lambda}}{\lambda}, \quad (2.16)$$

де α_t - оцінка стану в момент t , тобто $E(\alpha)$.

Алгоритм LMS належить до сімейства стохастичних градієнтних фільтрів. Алгоритм використовує приблизну оцінку градієнта на кожному кроці часу, щоб наблизитися до оптимального рішення функції середньоквадратичної помилки (MSE) [15].

Квадратична поверхня MSE має оптимальні значення параметрів у самій нижній точці. Ключовою особливістю алгоритму LMS є те, що він використовує грубе наближення до градієнта та уникає обчислення справжнього градієнта (що вимагає значно більше часу для обчислень). Після того, як функція сходиться до свого мінімуму, параметри виконують випадковий рух поблизу оптимальної точки. Оновлення стану за LMS:

$$\alpha_t = \alpha_{t-1} + \mu Z_{t-1} \epsilon_{t-1}. \quad (2.17)$$

Коефіцієнт розміру кроку μ визначає величину зміни, внесеної на кожному кроці до стану. Велике значення μ забезпечує вищу збіжність, але може призвести до нестабільності. З іншого боку, невелике значення забезпечує стабільність, але сповільнює зближення.

Авторегресійні моделі є ефективним інструментом прогнозування, але зазвичай використовуються в класичному варіанті, без адаптивних механізмів. Використання таких інструментів як LMS і RLS дозволяє покращити результати прогнозів, можливості обробити викиди в даних та в деяких випадках позбутися потреби у ручній оцінці параметрів.

2.2 FOREX-торгівля за допомогою методу опорних векторів (SVM)

Прогнозування валютного курсу є одним із складних в застосуванні сучасного прогнозування часових рядів. Реальні темпи за своєю сутністю мають нескінченні викиди, здебільшого нестационарні та детерміновано хаотичні. Історичні данні є основним вхідним матеріалом для процесу прогнозування. Для покращення результатів важливо знайти таку техніку та стратегію щоб задача прогнозування в результаті показувала найкращі показники відповідності фактичних та розрахункових даних.

Протягом більш ніж двох десятиліть для прогнозування часових рядів широко використовується техніка ARIMA (AutoRegressive Integrated Moving Average) Бокса та Дженкінса. За часів вчених для оцінювання коефіцієнтів розроблялись окремі методи кожної моделі, через значних обмежень у обчислювальній техніці. Наразі вони розробили загальний метод максимізації правдоподібності. Проте ARIMA є загальною одновимірною моделлю, і вона розроблена на основі припущення, що часові ряди, які прогнозуються, є лінійними та стаціонарними [20].

В останні роки нейронні мережі знайшли корисне застосування у фінансовому аналізі часових рядів та прогнозуванні. Потік вчених які неодноразово заявляли про це, та на практиці застосовували: Z. Tang, P. Fishwick, Dr. Jhee, U. Lee, M. Kamruzzaman, та багато інших. Вони показали, що штучні нейронні мережі працюють краще ніж ARIMA моделювання, особливо для більш нерегулярних рядів і для прогнозування на декілька періодів вперед. Вони проаналізували здатність методу зворотного розповсюдження помилки (backpropagation) прогнозувати валютних курс. Камруззаман і Саркер дослідили три моделі прогнозування на основі штучних нейронних мереж, використовуючи методу зворотного розповсюдження помилки з байєсовою регуляризацією та метод спряженого градієнта [28].

Останні дослідження були спрямовані на метод опорних векторів (SVM), який став новим потужним інструментом машинного навчання.

Основна перевага методу опорних векторів полягає в тому, що модель описує регресійну задачу в нелінійному просторі тобто квадратичної оптимізації. SVM виконує нелінійне відображення вхідних даних у просторі об'єктів великої розмірності за допомогою функції, а потім виконує лінійну регресію в перетвореному просторі. Весь процес спрощує нормалізація у низьковимірний простір.

Задано набір даних $G = \{(x_i, y_i)\}_{i=1}^N$, реалізуючи деяку функцію $g(x)$, потрібно визначити функцію f , яка апроксимує $g(x)$ базуючись на G (формула 2.18).

$$f(x) = \sum_i \omega_i \varphi_i(x) + b, \quad (2.18)$$

де $\varphi_i(x)$ - матриця залежних факторів, ω_i - ваги (коефіцієнти).

У регресії SVM, Володимир Вапнік запропонував використовувати ε -нечутливу функцію втрат, де помилка не штрафується. Тоді невідомий коефіцієнт оцінюється шляхом мінімізації наступної функції ризику (формула 2.19) [23].

$$R = C \frac{1}{N} \sum_{i=1}^N |y_i - f(x_i)| + \frac{\lambda}{2} \|\omega\|^2 \quad (2.19)$$

Перший доданок описує ε -нечутливу функцію втрат, а другий доданок є мірою плоскості функції. Константа $C > 0$ є регуляризованою константою, що визначає компроміс між похибкою навчання та площинністю моделі. Оптимізаційна задача, що вирішується SVM за відповідних обмежень, залежить від кінцевої кількості параметрів і має такий вигляд (формула 2.20):

$$f(x_i, \alpha_i, \alpha_i^*) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x, x_i) + b, \quad (2.20)$$

де $K(x, x_i)$ - Ядро,

α, α^* - коефіцієнти отримані максимізацією формули (2.21):

$$W(\alpha, \alpha^*) = \sum_{i=1}^N y_i (\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^N (\alpha_i - \alpha_i^*) + \frac{1}{2} \sum_{i,j=1}^N (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) K(x_i, x_j), \quad (2.21)$$

При $0 \leq \alpha_i^*, \alpha_i \leq C, \sum_{i=1}^N (\alpha_i^* - \alpha_i) = 0$.

Функцію ядра слід вибирати на основі поведінки моделі історичних даних курсів окремої валюти. У широкому діапазоні значень для параметра регуляризації опорні вектори залишаються абсолютно однаковими і, отже, не впливають на продуктивність взагалі. Покращення продуктивності може бути

досягнуто шляхом вибору змішаної функції ядра, при вивченні поведінки окремих поведінок процесів часових рядів.

Поєднуючи недоліки та переваги авторегресійних моделей та SVM можна зазначити, що сутність в нелінійності моделювання методу опорних векторів робить SVM більш потужнішим, ніж ARIMA. Але якщо дані апроксимуються до лінійного типу навантаження, то краще використовувати ARIMA моделювання.

2.3 Прогнозування часових рядів за допомогою нейронної мережі LSTM

В останні роки інструменти глибокого навчання, такі як LSTM (Long Short Term Memory), стали популярними і були визнані ефективними для багатьох проблем прогнозування часових рядів. Загалом, такі проблеми зосереджені на визначенні майбутніх значень даних часових рядів з високою точністю. Однак у задачах прогнозування напрямку точність не можна визначити як просто різницю між фактичними та прогнозованими значеннями. LSTM це різновид архітектури рекурентних нейронних мереж (RNN). RNN може витягувати інформацію часових рядів, забезпечувати збереження інформації та використовувати попередні знання для висновку про наступні моделі. Розглянемо послідовність $x = (x^{(1)}, x^{(2)}, \dots, x^{(l)})$ де кожен елемент $x^{(i)} \in R^d$ – вектор d розмірності. Навчаючи традиційну нейронну мережу на цих даних, подається вся інформація за один раз в мережу (рис. 2.3). Цей підхід ігнорує будь-які тимчасові змінні залежності, присутні в послідовності X . Крім того, кількість ваг у мережі буде зростати лінійно з довжиною послідовності [29].

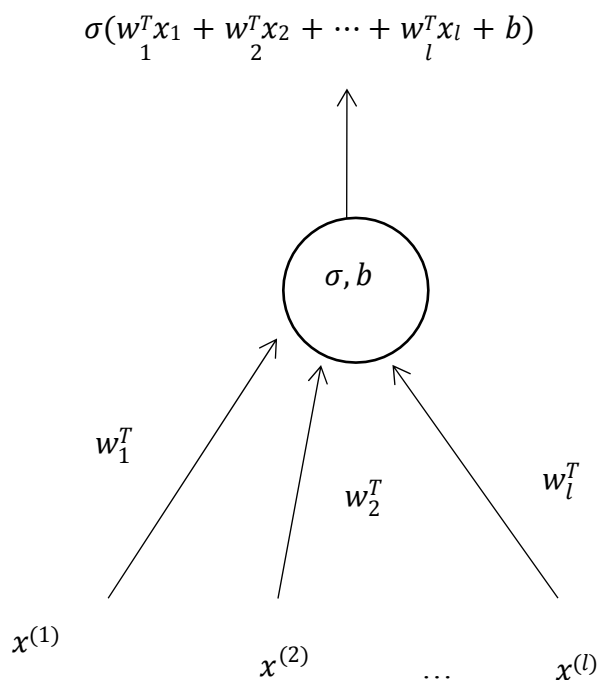


Рис. 2.3. Звичайна нерекурентна мережа

Джерело: [21].

Рекурентні нейронні мережі розроблені для більш ефективної обробки послідовних даних, беручи до уваги послідовний характер даних. Стандартний рекурентний нейронний блок може обробляти об'єкти послідовності по черзі, починаючи з першого елемента послідовності перед подачею другого. На кожній точці часу мережа приймає вхідні дані: об'єкт послідовності $x_i \in R^d$ та попередній об'єкт $h_{i-1} \in R$, це дозволяє корегувати поточні ваги моделі мережі [21]. Графічне зображення повторюваної нейронної одиниці показано на рис 2.4.

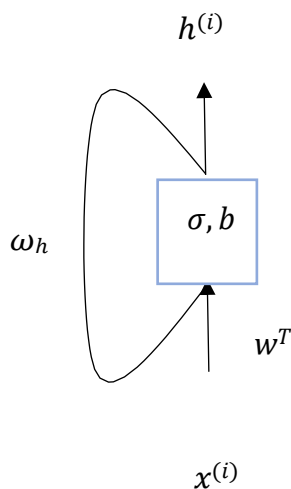


Рис. 2.4. Принцип роботи рекурентного нейрону

Джерело: [21].

LSTM-мережа здатна вчитися на базисі довгострокових залежностей. LSTM представлена в роботі Гохрайтера і Шмідхубера, 1997, а потім оптимізована та популяризована у багатьох подальших роботах. Такі мережі чудово справляються з вирішенням багатьох завдань і знаходять широке застосування зараз. Вона встановлює тривалу затримку між входом, зворотним зв'язком і запобіганням градієнтного вибуху, а також успішно усуває недоліки звичайної циклічної нейронної мережі. На додаток до циклу RNN, мережа циклів LSTM ще має внутрішній цикл «блоки LSTM» (самоцикл). Найважливішим її компонентом є показник стану. Пов'язані з ним константи часу керуються вентилям забуття (час T і блок i). Сигмоїдна складова встановлює вагу від 0 до 1 (формула 2.22).

$$f_i^{(t)} = \sigma(b_i^f + \sum_j U_{ij}^f x_j^{(t)} + \sum_j W_{ij}^f h_j^{(t-1)}) \quad (2.22)$$

Поточний вхідний вектор, який є поточним вектором прихованого шару. b -зміщення, вхідна вага та вага циклу [25]. Таким чином, внутрішній стан блоків LSTM оновлюється таким чином:

$$S_i^{(t)} = f_i^{(t)} S_i^{(t-1)} + \sigma(b_i + \sum_j U_{ij} x_j^{(t)} + \sum_j W_{ij} h_j^{(t-1)}) \quad (2.23)$$

Розраховуються зміщення, вхідна вага та вага циклу вентиля забування в блоках LSTM. Зовнішній блок вхідного вентиля оновлюється подібним чином до сигмоїдної (який має значення від 0 до 1), але зі своїми власними параметрами:

$$g_i^{(t)} = \sigma(b_i^g + \sum_j U_{ij}^g x_j^{(t)} + \sum_j W_{ij}^g h_j^{(t-1)}) \quad (2.24)$$

З наведеного вище процесу обробки інформації LSTM ми можемо чітко побачити, що він має хорошу пам'ять попередньої інформації і може ефективно вирішити проблему традиційного градієнта RNN. Оскільки новий стан блоку можна розглядати як деяку форму накопичення попереднього стану блоку. Похідною від нового стану блоку є також режим накопичення, що вирішує проблему традиційного градієнта RNN. Оскільки кінцевий результат цього є числовим значенням, після прихованого шару LSTM додається шар повного з'єднання, а потім останній вихідний шар. Функція активації цього шару визначається як лінійна функція [5].

Отже перевага LSTM полягає в тому, що модель може виводити прогнози з кількома часовими кроками. Але через концепцію ринкової ефективності модель може бути менш ефективною ніж звичайні авторегресійні аналоги. Також LSTM може показати погані результати використовуючи принцип маленької вибірки до якої застосовують просту модель. Поточна структура даної методики моделювання нейронних мереж може не вирішувати багатьох завдань регресії.

РОЗДІЛ 3

ВИКОРИСТАННЯ СУЧАСНИХ ПРАКТИК АВТОМАТИЗОВАНОЇ ТОРГІВЛІ ДЛЯ ПРОГНОЗУВАННЯ ТА АНАЛІЗУ РИНКУ FOREX

3.1 Розробка архітектури клієнтської та серверної частини ПЗ

Для розробки інформаційної системи, особливо в епоху web, потрібно виокремити два розділи розробки програмного забезпечення: бекенд та фронтенд (клієнтська та серверна частина). У комп'ютерному світі «бекенд» відноситься до будь-якої частини веб-сайту або програмного забезпечення, яку користувачі не бачать. Він використовує інтерфейс, який відноситься до користувальницької частини або сторінки веб-сайту. У термінології програмування бекенд — це «рівень доступу до даних», а інтерфейс — це «рівень презентації». Більшість сучасних веб-сайтів є динамічними, тобто вміст веб-сторінки створюється на льоту. Динамічна сторінка містить певні команди, які запускаються на веб-сервері при кожному зверненні до сторінки. Ці сценарії генерують вміст сторінки, який надсилається у веб-браузер користувача. Усе, що відбувається до того, як сторінка буде відображена у веб-браузері, є частиною бекенда (табл. 3.1).

Таблиця 3.1

Приклади серверної та клієнтської частини програмного забезпечення

Фронтенд	Бекенд
<ul style="list-style-type: none"> • Мови дизайну та розмітки, такі як HTML, CSS та JavaScript, • Пошукова оптимізація (SEO), • Інструменти графічного дизайну та редагування зображень, • Веб-продуктивність і сумісність з браузером. 	<ul style="list-style-type: none"> • Мови програмування такі як PHP, Python та C#, • Автоматизовані фреймворки, • Масштабованість і доступність мережі, • Управління базами даних і перетворення даних, • Кібербезпека та методи резервного копіювання даних.

Джерело: розроблено автором.

Інформаційна система яка представлена в роботі розроблена у форматі десктоп додатку яка взаємодіє з сервером (бекенд) через https запити. Потреба у

серверній розробці виникла з ціллю безперервної та стабільної роботи алгоритмів моделей.

Для реалізації серверної частини ПЗ був використаний Python фреймворк Flask [12]. Це фреймворк веб-додатків, написаний на Python. Його розробив Армін Роначер, який очолював команду міжнародних ентузіастів Python під назвою Pocco. Flask [12] базується на наборі інструментів Werkzeug WSGI [12] та механізмі шаблонів Jinja2. Обидва проекти Pocco. WSGI - інтерфейс шлюзу веб-сервера (Web Server Gateway Interface) використовувався як стандарт для розробки веб-додатків Python. WSGI — це специфікація спільного інтерфейсу між веб-серверами та веб-додатками. Werkzeug — це набір інструментів WSGI, який реалізує запити, об'єкти відповідей та допоміжні функції. Це дозволяє створити на ньому веб-фрейм. Фреймворк Flask [12] використовує Werkzeug як одну зі своїх баз. jinja2 — популярний механізм шаблонів для Python. Система веб-шаблонів поєднує шаблон із певним джерелом даних для відтворення динамічної веб-сторінки [12].

Це дозволяє передавати змінні Python у шаблони HTML, наприклад:

```
<html>
  <head>
    <title>{{ title }}</title>
  </head>
  <body>
    <h1>Hello {{ username }}</h1>
  </body>
</html>
```

де значення тега в {{ value }} змінна у пам'яті Python.

Flask часто називають мікрофреймворком тому, що ядро програми було простим. Замість рівня абстракції для підтримки бази даних, Flask підтримує вільне розширення у додаванні нових можливостей.

У створеному віртуальному середовищі був створений файл такого змісту:

блок імпорту:

```
from flask import Flask, request, send_file, render_template
import logging
from sqlalchemy import SQLite
```

Створення об'єкта Flask додатка:

```
app = Flask(__name__)
```

Створення маршрутизації програми це зіставлення URL-адреси з функціями які обраблюють логіку цієї адреси. Наприклад виконується запит в будь-якому браузері наприклад www.google.com, після цього йде сигнал за адресою до сервера по відповідному протоколу http/https, де виконується код який, наприклад, забирає дані вашого браузера про регіон, і передається до функції на сервері. Ці данні впливають на те яку інформацію отримає користувач в результаті локально на комп'ютер користувача. Була створено перші дві функції.

1. При запиті тільки домена адреси:

```
@app.route('/')def index():
    return "<h1>My Web-application</h1>"
```

- 2.

Функція для запиту бази даних з сервера:

```
@app.route('/get_db_file')
def get_file():
    file = 'pass.db'
    down_file = send_file(file, as_attachment=True,)
    return down_file
```

Перша функція відправляє текст розмітки html браузеру на машину користувача, і відображає цю сторінку. Друга функція означає коли користувач відправить запит у вигляді “домен адреси” + /get_db_file, він отримає завантаження файлу бази даних pass.db.

В управлінні базою даних не використовувалась наявна ORM програма, а був створений об'єкт Sqlite, де були реалізовані функції управління базою даних в sqliter.py файл.

```

import sqlite3
import secrets

class SQLiter:
    def __init__(self, db_name, table):
        self.db_name = db_name
        self.table = table
        self.conn = sqlite3.connect(self.db_name, check_same_thread=True)
        self.cursor = self.conn.cursor()

    def is_exist_user(self, login):
        result = self.cursor.execute( f"SELECT * FROM '{self.table}' WHERE login =
'{login}'" ).fetchall()
        if result:
            return True
        return False

    def add_user(self, login, password, email):
        if self.is_exist_user(login):
            return "Already exist!"
        else:
            api = secrets.token_hex(16)
            self.cursor.execute( f"INSERT INTO {self.table} (login, password, email, apikey)"
f" VALUES('{login}', '{password}', '{email}', '{api}');" )
            return "Successfully added user!"

    def enter(self, login, password):
        result = self.cursor.execute( f"SELECT * FROM '{self.table}' WHERE login = '{login}'
AND" f" password = '{password}'" ).fetchall()
        if not result:
            return False
        return result

```

Клас `Sqliter` ініціалізує аргументи які формуються при створенні екземпляра, а саме назва бази даних і назва таблиці в цій базі даних. В цьому об'єкті для початку була реалізована авторизація користувача та реєстрація в системі з погляду на безпеку, так як система буде відрито комунікувати з торговим терміналом та виконувати автоматично всі угоди. Функція “`is_exist_user`” приймає в якості аргументів логін і перевіряє чи існує користувач в базі даних. Функція “`add_user`” реєструє користувача в базі і “`enter`” виконує перевірку логіну та паролю в таблиці.

Для розробки інтерфейсу для користувачів була використана бібліотека `PyQt5`. Це набір прив'язок Python для фреймворку додатків Qt5 від Digia. Бібліотека Qt є однією з найпотужніших бібліотек GUI. `PyQt5` розроблено Riverbank Computing. `PyQt5` реалізовано як набір модулів Python. Він має понад 620 класів і 6000 функцій і методів. Це багатоплатформний набір інструментів,

який працює на всіх основних операційних системах, включаючи Unix, Windows і Mac OS.

Для даної системи було розроблено 2 головних вікна для користувача. Вікно авторизації з можливістю реєстрації та збереження поточних даних при вході в платформу, а також відновлення паролю (рис. 3.1), і код об'єкта класа.

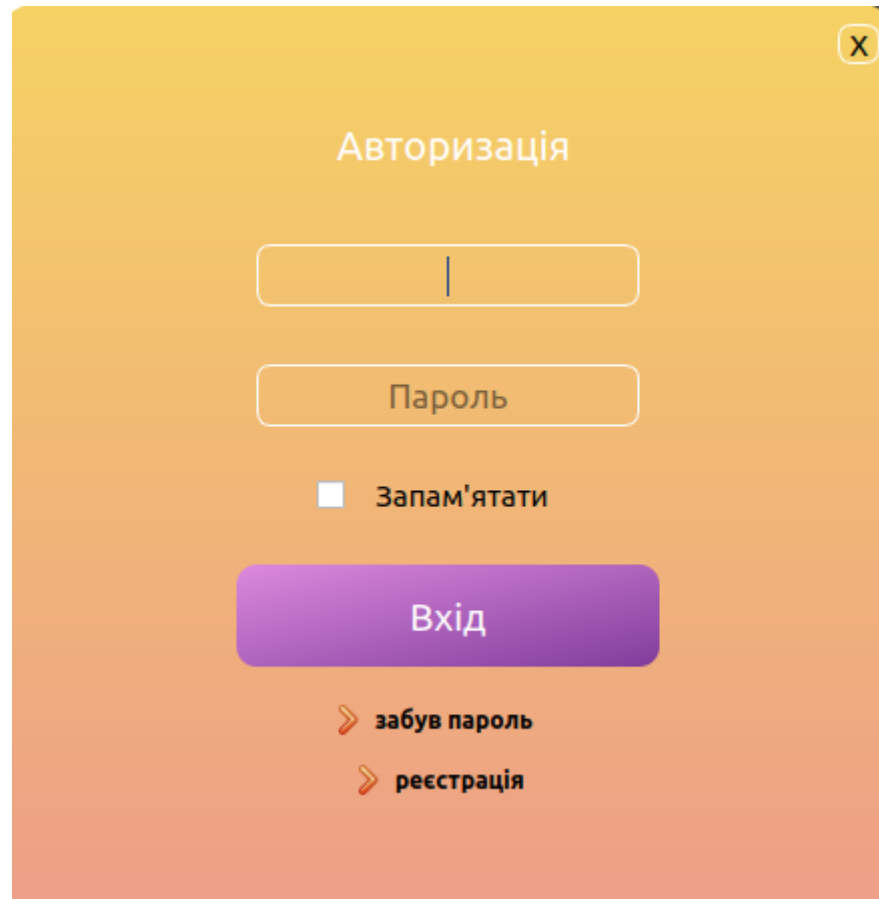


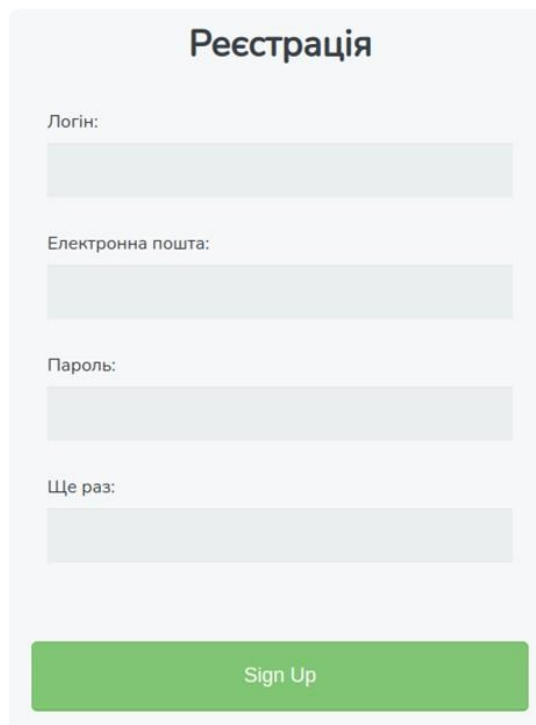
Рис. 3.1. Вікно авторизації

Джерело: розроблено автором.

Інтерфейс створений на платформі QtDesigner та конвертований в Python код. Для того щоб допрацювати та реалізувати функціонал було імпортовано та наслідувано клас даного вікна:

```
class AuthWindow(QtWidgets.QDialog):
    def __init__(self):
        super(AuthWindow, self).__init__()
        self.ui = Ui_Dialog()
        self.ui.setupUi(self)
        self.init_ui()
        self.init_widgets()
    def init_ui(self):
        self.setWindowFlag(Qt.FramelessWindowHint)
        self.setAttribute(Qt.WA_TranslucentBackground)
        self.ui.lineEdit_2.setEchoMode(QtWidgets.QLineEdit.Password)
    def init_widgets(self):
        self.ui.pushButton_2.clicked.connect(lambda: self.close())
        self.ui.pushButton.clicked.connect(lambda: self.login())
        self.ui.commandLinkButton_2.clicked.connect(lambda: self.open_registration())
```

Функція `super()` виконує все що наслідує поточний об'єкт класу `UI_Dialog()` який ініціалізує інтефейс та відображає вікно авторизації. Після чого сама модифікація виконується в дочірньому об'єкті. Кнопка реєстрації на рис. 3.1 відкриває сторінку в браузері для реєстрації користувача на рис. 3.2.



The image shows a registration form with the following elements:

- Title: **Реєстрація**
- Field 1: **Логін:** (Login) with a text input field.
- Field 2: **Електронна пошта:** (Email) with a text input field.
- Field 3: **Пароль:** (Password) with a password input field.
- Field 4: **Ще раз:** (Repeat) with a text input field.
- Button: A green button labeled **Sign Up** at the bottom.

Рис. 3.2. Вікно реєстрації користувача

Джерело: розроблено автором.

Головне вікно складається з панелі меню, головного фрейма та інформації про аккаунт з даних торгового терміналу (рис. 3.3).

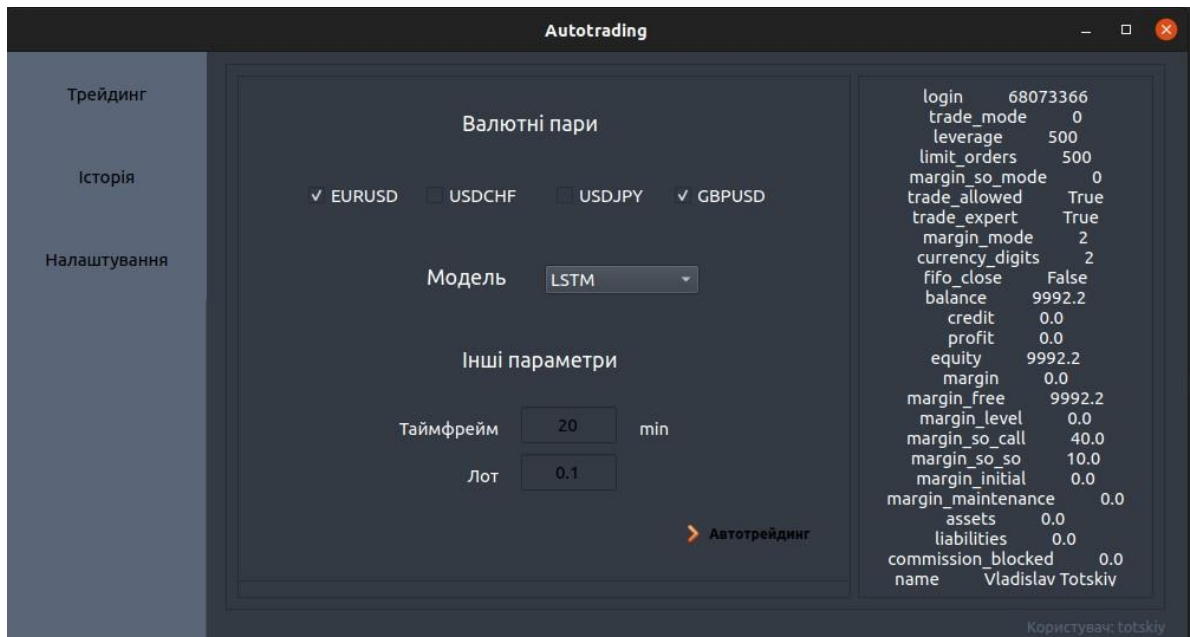


Рис. 3.3. Головне вікно

Джерело: розроблено автором.

Для інтеграції з торговим терміналом існує бібліотека `mt5`. Цей пакет дозволяє управляти торговими операціями за допомогою методів об'єкту класа `MetaTrader5`. Для реалізації алгоритму автоматичної торгівлі розроблений об'єкт з заданими константами для комфортного подальшого використання у алгоритмі:

```
class MetaTrader:
    def __init__(self, localhost, port):
        print("Running...")
        self.host = localhost
        self.port = port
        self.mt5 = MetaTrader5(host='localhost', port=self.port)
        print(f"Status is {self.mt5.initialize()}")
    def get_obj(self):
        return self.mt5
```

Де створений новий клас який ініціалізує такі змінні як хост підключення, порт і об'єкт mt5. Метод `get_obj()` повертає об'єкт, який в свою чергу надає доступ до основних функцій торгівлі. В рамках даного проекту використовується функція `order_send()` для відкриття угод.

Всередині поточного класу адаптована функція `up_order()`, у зв'язку з деякою кількістю констант:

```
def up_order(self, symbol, lot, type, deviation, comment='comment'):
    m_trader = self.get_obj()
    types = { 'buy': m_trader.ORDER_TYPE_BUY, 'sell': m_trader.ORDER_TYPE_SELL, }
    point = m_trader.symbol_info(symbol).point
    price = m_trader.symbol_info_tick(symbol).ask
    stop_rate = 1300
    sl = price - stop_rate * point
    tp = price + stop_rate * point
    request = {
        "action": m_trader.TRADE_ACTION_DEAL
        , "symbol": symbol,
        "volume": lot,
        "type": types[type],
        "price": price,
        "sl": sl,
        "tp": tp,
        "deviation": deviation,
        "magic": 200000,
        "comment": comment,
        "type_filling": m_trader.ORDER_FILLING_FOK,
        "type_time": m_trader.ORDER_TIME_GTC,
    }
    result = m_trader.order_send(request)
```

Функція приймає в якості аргументів:

`symbol` — (string) валютна пара,

`lot` — (float) величина угоди,

`deviation` — (float) максимально прийнятне відхилення від запитуваної ціни,

`comment` — (string) опис угоди.

Основа структури функціонування інформаційної системи створена на двох додатках: локальний десктоп та скрипти які виконуються на сервері віддалено за допомогою веб-запитів. Таким чином в другій частині даного розділу для реалізації розрахунків буде використаний метод який базується на запитах в свою чергу виконувати всі базові розрахунки на сервері повертаючи json-відповідь. JSON (JavaScript Object Notation) [1] — стандартний текстовий формат для

представлення структурованих даних на основі синтаксису JavaScript. Він зазвичай використовується для передачі даних у веб-додатках (наприклад, відправлення деяких даних із сервера клієнту, таким чином, щоб це могло відобразитися на веб-сторінці або навпаки) [1].

Отже було створено базовий функціонал клієнтської та серверної частини на популярних інструментах Flask [12] і PyQt5 відповідно. Було також створено об'єкт mt5 для управління metatrader клієнтом для реалізації автотрейдингом наряду з нашої інформаційної системи. Така структура програмного забезпечення дозволяє легко орієнтуватися в репозиторіях та просто додавати додатковий функціонал для розширення можливостей системи.

3.2 API інтеграція взаємодії з клієнтським торговим терміналом та розробка алгоритму аналізу

Long Short Term Memory Networks (LSTM) здатні вловлювати закономірності даних тимчасових рядів і тому можуть бути використані для прогнозування майбутнього тренду даних. Для реалізації алгоритму в роботі була використана мова програмування Python, так як вона дає відкритий доступ до різних методів та моделей машинного навчання та штучного інтелекту. Такі розробки як numpy, pandas, scikit-learn, theano, tensorflow, keras та багато інших які часто використовуються спеціалістами для вирішення конкретних завдань.

Перед реалізацією алгоритму для торгівлі, потрібно розглянути які саме інструменти для цих завдань треба розглянути. Також ці інструменти використаємо на простому прикладі.

В першу чергу були імпортовані пакети:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from keras.layers
import LSTM,
Dense
```

- Numpy – (лінійна алгебра) для роботи з векторами та матрицями,

- Pandas — робота з датасетами та датафреймами,
- Matplotlib — побудова графіків,
- keras — це надбудова Tensorflow, яка дозволяє взаємодіяти з штучними нейронними мережами.

Для створення тестового набору даних, написана функція яка задає випадкові коливання в діапазоні тренду:

```
DATE_START = '2021-01-01'  
DATE_END = '2021-12-31'  
df_dates = pd.date_range(DATE_START, DATE_END)  
df = pd.DataFrame({'date': df_dates, 'value': np.random.normal(0, 1, df_dates.size)})  
df.set_index('date', inplace=True)
```

Змінна `df` — об'єкт даних часового ряду зі явним випадковим стаціонарним процесом, нормального розподілу (рис. 3.4).

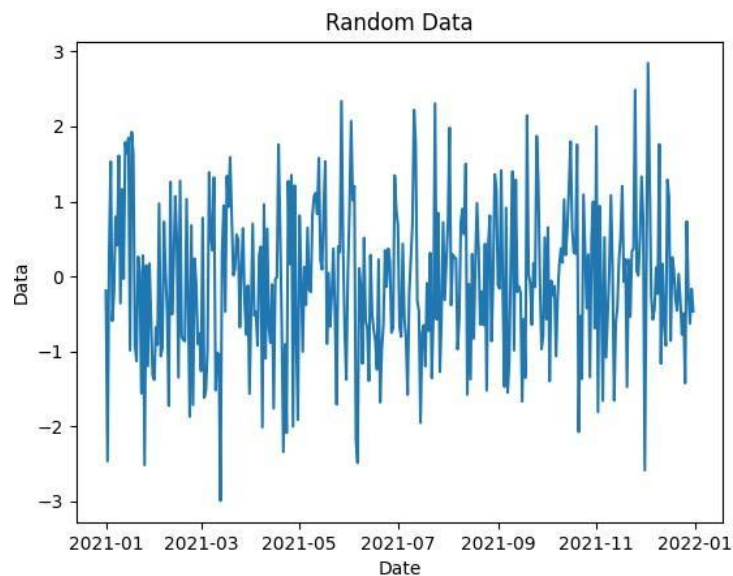


Рис. 3.4. Випадкові дані

Джерело: розроблено автором.

Щоб зробити дані випадково нестационарні прописана функція такого виду:

```

def random_walk( df, start_value=0, threshold=0.5, step_size=1, min_value=-np.inf, max_value=np.inf):
    previous_value = start_value
    for index, row in df.iterrows():
        if previous_value < min_value:
            previous_value = min_value
        if previous_value > max_value:
            previous_value = max_value
        probability = random.random()
        if probability >= threshold:
            df.loc[index, 'value'] = previous_value + step_size
        else:
            df.loc[index, 'value'] = previous_value - step_size
        previous_value = df.loc[index, 'value']
    return df

```

Функція `random_walk()` повертає об'єкт перетвореного датафрейму (рис. 3.5).

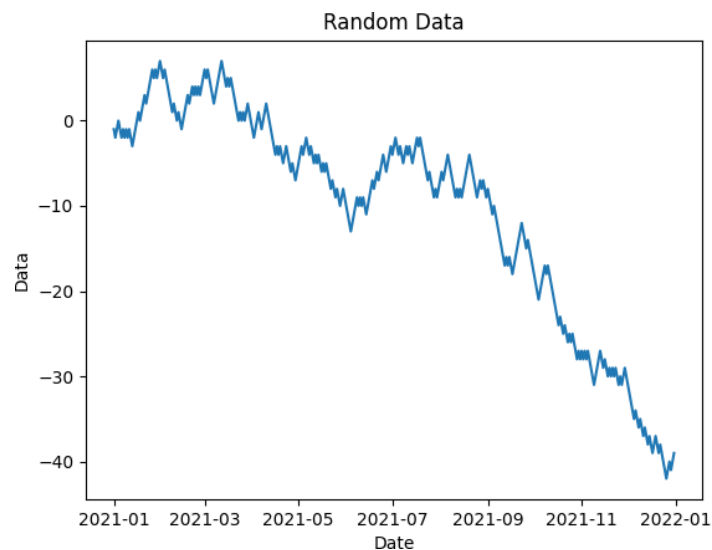


Рис. 3.5. Нестационарний тестовий часовий ряд

Джерело: розроблено автором.

На цьому етапі проведено нормалізацію даних. Нормалізація — це перетворення даних до деяких безрозмірних одиниць. Іноді - в рамках заданого діапазону.

```

from sklearn.preprocessing
import MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
df_training_scaled = scaler.fit_transform(df)

```

де `feature_range` — заданий діапазон,
`df_training_scaled` — трансформовані дані.

У завданнях часового ряду ми повинні передбачити значення в момент часу t , ґрунтуючись на даних із днів $T-N$, де N може бути будь-яким числом кроків. Виконано перетворення на скорочення T на N кроків:

```
features_set = []
labels = []
for i in range(5, 365):
    features_set.append(df_training_scaled[i-5:i, 0])
    labels.append(df_training_scaled[i, 0])
```

У наведеному вище скрипті ми створюємо два списки `feature_set` і `labels`. У даних навчання міститься 365 записів. Ми виконуємо цикл, який починається з 6-го запису та зберігає всі попередні 5 записів до списку `feature_set`. 6 запис зберігається в списку `labels`. Потрібно перетворити `feature_set` і `labels` в масив `numpy`, перш ніж ми зможемо використовувати його для тестування. Наступний скрипт:

```
features_set = np.array(features_set)
labels = np.array(labels)

features_set = np.reshape(features_set, (features_set.shape[0], features_set.shape[1], 1))
```

Першим кроком потрібно створити екземпляр класу `Sequential`. Це буде клас моделей, в цей контейнер додається модель LSTM, Dropout та щільні шари. Як вказано у скрипті:

```
model.add(LSTM(units=40, return_sequences=True, input_shape=(features_set.shape[1], 1)))
model.add(Dropout(0.25))
```

де `DropOut` — це шар відсіву для того щоб не було оверфитингу.

`Units` — кількість нейронів

`return_sequences` — зазвичай приймає значення `True` коли в модель приймає додатковий шар.

Подальшим кроком додається ще декілька моделей LSTM тобто збільшення шаровість нейромережі:.

```
model.add(LSTM(units=40, return_sequences=True))
model.add(Dropout(0.25))
model.add(LSTM(units=40))
model.add(Dropout(0.25))
```

Створення щільного шару зробить модель більш надійною, таким чином встановимо 1 нейрон:

```
model.add(Dense(units=1))
```

Компіляція та навчання моделі:

```
model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(features_set, labels, epochs=100, batch_size=32)
```

Оптимізація Адама — це метод стохастичного градієнтного спуску, який базується на адаптивній оцінці моментів першого та другого порядку (`optimizer='adam'`).

Для використання алгоритму на практиці оцінено на реальних даних валютної пари EUR/USD (370 торгових днів від 1.05.2020) (рис. 3.6):

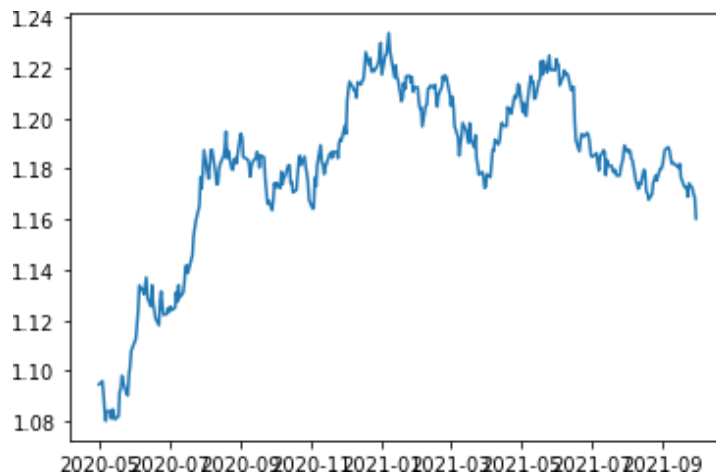


Рис. 3.6. Котирування EUR/USD

Джерело: розроблено автором.

Можна побачити з графіка, тренд дуже нелінійний, важко передбачити напрямком. За рахунок того, що LSTM запам'ятовує попередню інформацію про ряд, вона показує непогані результати на рис. 3.7.

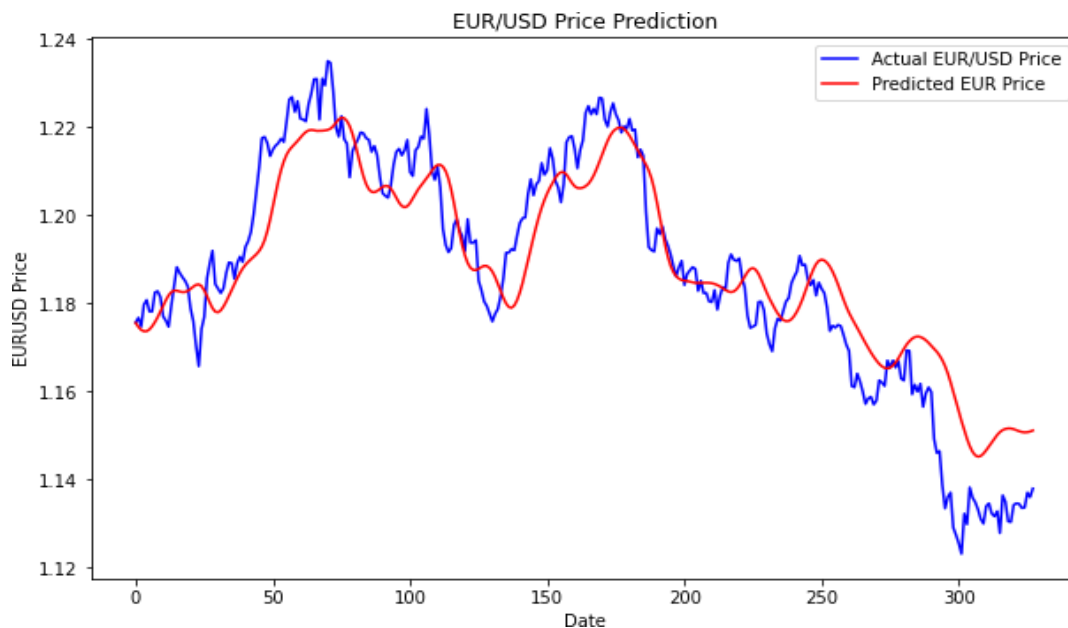


Рис. 3.7. Прогнозування валюної пари EUR/USD

Джерело: розроблено автором.

Для тестування було взято наступний рік для порівняння фактичних значень котирувань. Як сигнал для здійснення угоди може слугувати перетин значень фактичних та прогнозованих даних.

Показники моделі:

MSE — 1.08162,

MAE — 0.81809.

Layer (type)	Output Shape	Param #
lstm_4 (LSTM)	(None, 60, 50)	10400
lstm_5 (LSTM)	(None, 60, 50)	20200
dropout_3 (Dropout)	(None, 60, 50)	0
lstm_6 (LSTM)	(None, 60, 50)	20200
dropout_4 (Dropout)	(None, 60, 50)	0
lstm_7 (LSTM)	(None, 50)	20200
dropout_5 (Dropout)	(None, 50)	0
dense_1 (Dense)	(None, 1)	51

=====
 Total params: 71,051
 Trainable params: 71,051
 Non-trainable params: 0

Рис. 3.8. Результати багатошарової нейронної мережі

Джерело: розроблено автором.

В цьому розділі виконаний покроковий аналіз алгоритму навчання нейронної мережі на базі LSTM. Навідміну від інших моделей машинного навчання такі як SARIMA, Long Short Term Memory навчається на довгострокових залежностях та оновлює інформацію коли це потрібно. Навіть родинний RNN відстає на крок від сучасного LSTM. В роботі розроблений алгоритм торгівлі який протягом усього циклу аналізу атоматично приймає рішення щодо реєстрування угоди на Forex. Завдяки механізму “stop loss / take profit” угода атоматично закривається на певних межах що встановлені користувачем. Ці межі повинні залежати від величини валотильності котирувань, найчастіше це залежить від торгових сесій. Наприклад валютна пара EUR/USD має валотильність набагато вищу під час торгів у європейській сесії ніж у азійській. Продемонстрований графік на Рис.3.7 показує, що нейронна мережа яка використовувалась для моделювання часового ряду показує цікаві та корисні результати, покладаючись на які алгоритм в змозі чітко виокремити тренди та атоматично відкривати та закривати угоди отримуючи прибуток.

ВИСНОВКИ

Завдяки легкодоступності хмарних сервісів та доступів до потужних комп'ютерів, такий вид діяльності як інтернет-трейдинг став більш доступний для багатьох людей сьогодні. Окрім розвитку хмарних середовищ, за останні 10 років розвиток у сфері Data Science досяг виробничих масштабів. Тепер у повсякденному житті ми не звертаємо увагу на те, що практично 80% всіх веб-сервісів використовують дану технологію. Високий рівень знань в сфері аналізу даних дав змогу використовувати програмування та алгоритмізацію не тільки в цілях розробки програмних продуктів, але й вирішення складних розрахункових завдань, результати яких можуть слугувати відправною точкою у прийнятті рішень в сфері інвестицій.

Розглянувши підходи до визначення ринку Forex та його учасників, було визначено, що валютний ринок (також відомий як FX) є глобальним ринком для обміну національних валют одна на одну. Валютний ринок є найбільшим фінансовим ринком у світі. Банки, комерційні компанії, хедж-фонди, центральні банки та окремі спекулянти беруть участь у ньому та щодня обмінюють валюти як для спекулятивних, так і для хеджувальних цілей.

Встановлено фактори впливу на валютний курс. Найбільш важливими, що виділені в роботі є: темпи інфляції, процентні ставки, поточний рахунок країни, державний борг та спекуляція. Ці фактори змінюють графік попиту та пропозиції та створюють новий обмінний курс у новому рівноважному стані. Але існують і деякі інші фактори, такі як політична стабільність, умови торгівлі, ринкові рішення тощо, які також відіграють істотну роль у визначенні стану попиту на валюту та визначенні зміни курсу іноземної валюти.

Досліджено, що авторегресійні моделі є ефективним інструментом прогнозування, але зазвичай використовуються в класичному варіанті, без адаптивних механізмів. Використання таких інструментів як LMS і RLS дозволяє

покращити результати прогнозів, можливості обробити викиди в даних та в деяких випадках позбутися потреби у ручній оцінці параметрів.

Порівняно недоліки та переваги авторегресійних моделей та SVM, в результаті чого можна зазначити, що сутність в нелінійності моделювання методу опорних векторів робить SVM більш потужнішим, ніж ARIMA. Але якщо дані апроксимуються до лінійного типу навантаження, то краще використовувати ARIMA моделювання відповідних процесів.

Встановлено перевагу LSTM. Вона полягає в тому, що модель може виводити прогнози з кількома часовими кроками. Але через концепцію ринкової ефективності модель може бути менш ефективною ніж звичайні авторегресійні аналоги. Також LSTM може показати погані результати використовуючи принцип маленької вибірки до якої застосовують просту модель. Поточна структура даної методики моделювання нейронних мереж може не вирішувати багатьох завдань регресії.

Створено базовий функціонал клієнтської та серверної частини ПЗ на популярних інструментах Flask і PyQt5. Було також створено клас mt5 з управління metatrader клієнтом для реалізації автоматичного здійснення угод в інформаційній системі. Така структура програмного забезпечення дозволяє легко орієнтуватися в репозиторіях та просто додавати додатковий функціонал для розширення можливостей системи.

Досліджено, що практичні аспекти використання LSTM, в створенні багатошарової моделі показує ефективні результати. На відміну від інших моделей машинного навчання, такі як SARIMA, Long Short Term Memory навчається на довгострокових залежностях та оновлює інформацію коли це потрібно. Навіть родинний RNN відстає на крок від сучасного LSTM.

В роботі розроблений алгоритм торгівлі який протягом усього циклу аналізу автоматично приймає рішення щодо реєстрування угоди на Forex. Завдяки механізму “stop loss / take profit” угода автоматично закривається на певних

межах що встановлені користувачем. Ці межі повинні залежати від величини волатильності котирувань, яка в свою чергу залежить від торгових сесій.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Работа с JSON. URL: <https://developer.mozilla.org/ru/docs/Learn/JavaScript/Objects/JSON>.
2. Черняк О.І., Ставицький А.В. Навчально-методичний комплекс з курсу „Часові ряди” для студентів спеціальностей „Економічна кібернетика” та „Прикладна економіка”. – К. : Видавничо-поліграфічний центр „Київський університет”, 2004. – 26 с. URL: http://andriystav.cc.ua/Downloads/TS/TS_method.pdf.
3. Ходаківська В. П., Данілов О. Д. Ринок фінансових послуг: Навчальний посібник. – Ірпінь: Академія ДПС України, 2001. – 501 с. URL: <https://readbookz.net/books/129.html>.
4. Advantages and Disadvantages of Forex Market. Management Study Guide. URL: <https://www.managementstudyguide.com/importance-of-forex-education.htm>.
5. A. Mittal. Understanding RNN and LSTM. URL: <https://aditi-mittal.medium.com/understanding-rnn-and-lstm-f7cdf6dfc14e>.
6. Autoregression and Moving Average (ARMA) Models. URL: <https://www.statistics.com/glossary/autoregression-and-moving-average-arma-models/>.
7. Autoregressive Moving Average Model. Essentials of Time Series for Financial Applications. 2018. URL: <https://www.sciencedirect.com/topics/economics-econometrics-and-finance/autoregressive-moving-average-model>.
8. B. Artley. Time Series Forecasting with ARIMA , SARIMA and SARIMAX. Towards Data Science. Apr 26, 2022. URL: <https://towardsdatascience.com/time-series-forecasting-with-arima-sarima-and-sarimax-ee61099e78f6>.
9. Foreign exchange dealer. Nasdaq. URL: <https://www.nasdaq.com/glossary/f/foreign-exchange-dealer>.

10. Foreign Exchange Market. Handbook of Safeguarding Global Financial Stability. 2013. URL: <https://www.sciencedirect.com/topics/economics-econometrics-and-finance/foreign-exchange-market>.
11. Forex Broker. Investopedia. URL: <https://www.investopedia.com/terms/forex/c/currency-trading-forex-brokers.asp#:~:text=A%20forex%20broker%20is%20a,pair%20of%20two%20different%20currencies>.
12. Introduction to Flask. URL: <https://pymbook.readthedocs.io/en/latest/flask.html>.
13. J. Chen. Forex Market. Investopedia. January 31, 2022. URL: <https://www.investopedia.com/terms/forex/f/forex-market.asp>.
14. Kalman filter. URL: <http://pzs.dstu.dp.ua/DataMining/kalman/index.html>.
15. Least Mean Square (LMS) Filters. URL: keil.com/pack/doc/CMSIS/DSP/html/group__LMS.html.
16. T. Pettinger. Factors which influence the exchange rate. Economics Help. 28 June 2019. URL: <https://www.economicshelp.org/macroeconomics/exchangerate/factors-influencing/>.
17. T. Segal. Forex Market: Who Trades Currencies and Why. Investopedia. August 18, 2021. URL: <https://www.investopedia.com/articles/forex/11/who-trades-forex-and-why.asp#:~:text=Central%20banks%2C%20which%20represent%20their,its%20native%20currency%20on%20forex>.
18. The Foreign Exchange Market. Vedantu. URL: <https://www.vedantu.com/commerce/the-foreign-exchange-market>.
19. R. Akter. Factors that Determine and Influences Foreign Exchange Rates. Preprints 2021, 2021060240. DOI: 10.20944/preprints202106.0240.v1.
20. R. Sarker. Comparing ANN based models with ARIMA for prediction of forex rates. Asor Bulletin. 2003. URL:

- https://www.academia.edu/12697067/Comparing_ANN_based_models_with_ARIMA_for_prediction_of_forex_rates.
- 21.S. Elsworht, S. Guttel. Time Series Forecasting Using LSTM Networks: A Symbolic Approach. arXiv:2003.05672v1 [cs.LG] 12 Mar 2020. URL: <https://arxiv.org/pdf/2003.05672.pdf>.
 - 22.S. Natakarnkitkul. Time Series prediction using Adaptive filtering. Machine Learning, Towards AI. July 23, 2020. URL: <https://towardsai.net/p/machine-learning/time-series-prediction-using-adaptive-filtering>.
 - 23.S. Ruping. SVM Kernels for Time Series Analysis. URL: <https://www-ai.cs.tu-dortmund.de/EVENTS/FGML2001/FGML2001-Paper-Rueping.pdf>.
 - 24.S. Seth. Forex Trading Career: Pros and Cons. Investopedia. January 17, 2022. URL: <https://www.investopedia.com/articles/forex/050115/pros-cons-forex-trading-career.asp>.
 - 25.SIMA SIAMI NAMIN, AKBAR SIAMI NAMIN. FORECASTING ECONOMIC AND FINANCIAL TIME SERIES: ARIMA VS. LSTM. URL: <https://arxiv.org/ftp/arxiv/papers/1803/1803.06386.pdf>.
 - 26.Speculator. Nasdaq. URL: <https://www.nasdaq.com/glossary/s/speculator>.
 - 27.What Is Forex Trading? (With Advantages and Disadvantages). Indeed. September 30, 2021. URL: <https://www.indeed.com/career-advice/career-development/what-is-forex-trading>.
 - 28.Y. Ensafi, S. Hassanzadeh. Time-series forecasting of seasonal items sales using machine learning – A comparative analysis. International Journal of Information Management Data Insights. Volume 2. Issue 1. April 2022. DOI: <https://doi.org/10.1016/j.jjime.2022.100058>.
 - 29.Y. Qu, X. Zhao. Application of LSTM Neural Network in Forecasting Foreign Exchange Price. Journal of Physics Conference Series 1237(4):042036. June 2019. DOI:10.1088/1742-6596/1237/4/042036.

30.S. Shojaeipour. The Relationship Between Exchange Rates and Inflation: The Case of Iran. *European Journal of Sustainable Development* (2017). 6, 4, 329-340 ISSN: 2239-5938 DOI: 10.14207/ejsd.2017.v6n4p329.

31.6 factors influencing exchange rates and what you can do about it. URL: <https://www.ofx.com/en-au/blog/exchange-rate-volatility-causes/>.

Київський національний університет імені Тараса Шевченка

Економічний факультет

Кафедра економічної кібернетики

Розробка інформаційної системи для прогнозування та аналізу ринку Forex

ЗАВДАННЯ

на кваліфікаційну роботу магістра

студента 2 курсу магістратури спеціальності 051 «Економіка», ОНП «Економічна кібернетика»

Тоцького Владислава Віталійовича

1. Тема роботи: «Розробка інформаційної системи для прогнозування та аналізу ринку Forex».
2. Термін завершення роботи: 18.05.2022.
3. Об'єкт дослідження: валютний ринок Форекс.
4. Предмет дослідження: теоретичні, практичні основи та сучасні тенденції методологічного використання технологій для аналізу валютних курсів.
5. Мета дослідження: удосконалення теоретичних основ та розробка практичних методів, які можуть використовуватися не тільки у сфері аналізу ринку Forex, але й застосування цих методів в інших сферах аналізу та прогнозування.
6. Завдання дослідження:
 - 6.1. Уточнити зміст категорій пов'язаних з Forex та його учасниками, розкрити зміст та теоретичні аспекти діяльності Forex в загальному.
 - 6.2. Визначити фактори, які впливають на валютні курси, та які показники в практичному значенні можуть покращити результати аналізу.
 - 6.3. Розкрити визначення і сутність адаптивних методів прогнозування часових рядів.
 - 6.4. Розкрити теоретичний аспект прогнозування за допомогою методу опорних векторів.

- 6.5. Визначити переваги використання нейронної мережі LSTM для аналізу часових рядів.
- 6.6. Розробити архітектуру інформаційної системи.
- 6.7. Розробити API-інтеграцію з веб та алгоритм для прогнозування цін валютних пар.

Науковий керівник: кандидат економічних наук, доцент Шпирко Віктор Васильович.

Студент: Тоцький Владислав Віталійович.

Затверджено на засіданні кафедри економічної кібернетики протокол №3
від 12 жовтня 2021 р.

Календарний план виконання кваліфікаційної роботи магістра

№	Етапи роботи	Терміни виконання	Відмітка керівника про виконання
1	Вибір теми магістерської роботи	01.09.2021- 30.09.2021	
2	Розробка та затвердження завдання кваліфікаційної роботи магістра	21.02.2022- 27.02.2022	
3	Пошук і дослідження літератури по темі магістерської роботи	01.03.2022- 05.03.2022	
4	Складання плану роботи	06.03.2022- 07.03.2022	
5	Написання вступу і першого розділу магістерської роботи	09.03.2022- 14.03.2022	
6	Написання другого розділу магістерської роботи	15.03.2022- 30.03.2022	
7	Написання третього розділу магістерської роботи	02.04.2022- 29.04.2022	
8	Узгодження з керівником висновків і рекомендацій по магістерській роботі	02.05.2022- 03.05.2022	
9	Оформлення магістерської роботи та підготовка графічних матеріалів	03.05.2022- 04.05.2022	
10	Подання роботи до перевірки на плагіат	16.05.2022	
11	Подання роботи до захисту	19.05.2022	

Науковий керівникк.е.н., доц. Шпирко В. В.

Студент Тоцький А. О.