

ІНТЕГРАЦІЯ META TRADER ТА PYTHON ДЛЯ ОБРОБКИ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ ТА ВИКОНАННЯ ОПЕРАЦІЙ ОНЛАЙН

Вінницький національний технічний університет

Анотація

У статті наведено рішення щодо спільної роботи в онлайн торговій системі Meta Trader, через яку виконуються торгові операції з фінансовими інструментами, і Python, на бік якої покладається інтелектуальна обробка часових фінансових рядів.

Ключові слова: часові ряди, Meta Trader, трейдинг, прогнозування, нейронні мережі, Python.

Abstract

The article presents solutions for collaboration in the online trading system Meta Trader, through which financial instruments are traded, and Python, which is responsible for the intelligent processing of financial time series.

Keywords: axis series, Meta trader, trading, forecasting, neural networks, Python.

Вступ

Сфера інформаційних технологій стрімко та неспинно розвивається, даючи поштовх усім іншим аспектам як нашого сьогодення, так і іншим галузям сучасного життя. Вплив на фінансову сферу є надзвичайно важливим, адже нові технології дають можливість брати участь у фінансових операціях широкому колу людей. Вже зараз такі платформи як MetaTrader дозволяють здійснювати операції онлайн з фінансовими інструментами маючи акаунт в спеціалізованій фінансовій компанії (брокера) та невеличкий капітал, який є у розпорядженні. Але для вдалих операцій необхідно дещо більше. Аналіз даних та виявлення залежностей є основою успіху, саме тому прогнозування являє собою важливу тему для досліджень.

З наведеного у [1] доцільно звернути увагу на мову програмування Python з ціллю інтеграції існуючих методів аналізу часових рядів, які вже реалізовані в експертних системах типу MetaTrader та за допомогою широких можливостей мови, перш за все інтелектуальних технологій [2-4], покращити результати прогнозування фінансових часових рядів.

Реалізація

Для реалізації сумісної роботи терміналу MetaTrader та модуля Python необхідно знайти універсальний спосіб передачі даних, який би підтримувався з обох сторін та дозволяв двом різним мовам програмування спілкуватися між собою. Хоча існують й інші способи уніфікованого обміну даних (XML, JSON тощо), сокетна реалізація є найбільш прийнятною за умови обміну даними між двома процесами на одному сервері. Для зручного оперування сокетами на стороні Python доцільно створити клас та наповнити його усіма необхідними інструментами.

Сокет – назва програмного інтерфейсу для забезпечення обміну даними між процесами. При такому обміні процеси можуть виконуватись як на одній ЕОМ, так і на різних ЕОМ за умови спільної мережі. Сам сокет являється абстрактним об'єктом, який представляє собою кінцеву точку зв'язку. Сокет визначається адресною парою – номером порту та адресом серверу стека TCP/IP. Кожен процес може створити «слухаючий» сокет та прив'язати його до певного порту сервера. Такий процес знаходиться у стані очікування. Він здатен перевіряти статус підключення, встановити тайм-аут тощо. Зазвичай клієнт напряму «зв'язується» до слухача, після чого будь-яке читування або запис будуть передавати дані між ним та сервером. Програмну реалізацію сокетів підтримують майже усі сучасні мови програмування. Майже усі реалізації слідує одному шаблону набору методів та функцій.

Реалізована модель взаємодії визначає два процеси: процес роботи торгового експерта та процес роботи програми на Python. Схема взаємодії зображена на рисунку 1.

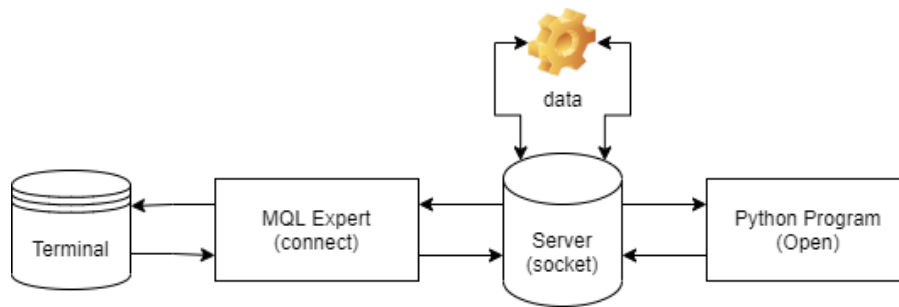


Рисунок 1 – Модель взаємодії MQL та Python.

Процес обміну даними розбитий на 6 етапів. Даний механізм утворює єдину систему для передачі, обробки, зворотної передачі та прийняття рішення.

1. Розгортання сокет-серверу з боку програми Python.
2. Приєднання до сокета з боку експерта. Майже ідентична реалізація у скрипті терміналу.
3. Збір даних та відправка за допомогою програмних засобів MQL. Дані відправляються у формі масиву символів.
4. Прийом даних та обробка інформації на стороні Python.
5. Відправка оброблених даних назад на сокет.
6. Прийом оброблених даних на стороні терміналу та прийняття рішення.

Застосована реалізація слідує основним засадам використання сокетів, але з деякими модифікаціями. Так як сигнал потрібно відправляти та приймати один за одним в одному інтервалі на протязі невизначеного проміжку часу, обидві сторони є слухачами. Процес терміналу MT перший надсилає сигнал на вже відкритий сокет, після чого приймає відповідь зі сторони Python. Така послідовність дій повторюється до визначеної користувачем зупинки. Таким чином за допомогою безперервного прослуховування затримка сигналу мінімізується. При застосуванні необхідної бізнес-логіки користувач має можливість автоматизувати будь-які обрахунки як на стороні MQL, так і на стороні Python, та об'єднати їх у єдину систему.

Висновок

Використовуючи запропоноване рішення по інтеграції Python та MetaTrader надалі можливо використовувати всю потужність інтелектуальних технологій аналізу, бібліотек першого для обробки масивів даних часових фінансових рядів та проведення онлайн операцій на основі отриманих рішень останньою.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Глебов І. В., Кабачій В. В. Прогнозування часових рядів та інтегрування торгових сигналів за допомогою Python Ефективні // Матеріали конференції «L Науково-технічна конференція підрозділів Вінницького національного технічного університету (2021)», Вінниця, 2021. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/allvntu/index/pages/view/zbirn2021>
2. Кветний Р. Н. Імовірнісні нейронні мережі в задачах ідентифікації часових рядів [Електронний ресурс] / Р. Н. Кветний, В. В. Кабачій, О. О. Чумаченко // Наукові праці Вінницького національного технічного університету. – 2010. – № 3. – 6 с. – Режим доступу до журн. : <http://www.nbu.gov.ua/e-journals/VNTU/2010-3/2010-3.html> .
3. Raschka S. Python Machine Learning: Machine Learning and Deep. Learning with Python, scikit-learn, and TensorFlow 3rd Edition / S. Raschka, V. Mirjalili., 2019. – 770 с.
4. Нейронні мережі як інструмент прогнозування [Електронний ресурс]. – Режим доступу: <https://012.ucoz.ru/publ/1-1-0-21>

Кабачій Владислав Володимирович – кандидат технічних наук, доцент кафедри АІТ, Вінницький національний технічний університет, м. Вінниця, e-mail: vkabachiy@gmail.com

Глебов Іван Юрійович – програміст, м. Вінниця, e-mail: morpice0009@gmail.com

Kabachii Vladyslav Volodmyrovych - Candidate of Technical Sciences, Associate Professor of АІТ, Vinnytsia National Technical University, Vinnitsia, e-mail: vkabachiy@gmail.com

Глебов Іван Юрійович – programmer, Vinnytsia, e-mail: morpice0009@gmail.com