

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-54-11>

УДК 004.853, 004.942

Єськіна Анна Андріївна, магістр

<https://orcid.org/0009-0008-4992-4286>

Кулаковська Інесса Василівна, к.ф.-м.н., доцент

<https://orcid.org/0000-0002-8432-1850>

Чорноморський національний університет імені Петра Могили, м Миколаїв, Україна

ДОСЛІДЖЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ КЛАСИФІКАЦІЇ ВОЛАТИЛЬНОСТІ ВАЛЮТ ЗА РЕЗУЛЬТАТАМИ МАШИННОГО НАВЧАННЯ

Єськіна А.А., Кулаковська І.В. Дослідження інтелектуальної системи класифікації волатильності валют за результатами машинного навчання. Розглядається процес класифікації та аналізу даних щодо волатильності валют використовуючи методи машинного навчання. Досліджуються моделі та методи машинного навчання, а саме наївний баєсівський класифікатор та метод опорних векторів для вирішення задач класифікації на основі сформованого набору даних. Метою дослідження є розробка та дослідження інтелектуальної системи, яка здатна ефективно класифікувати рівень волатильності на валютних ринках. Робота спрямована на вивчення різних методів машинного навчання та класифікаторів на основі різних мір подібності, з метою визначення оптимального підходу для класифікації волатильності.

Ключові слова: інтелектуальна система, методи машинного навчання, класифікація, метрики оцінки якості класифікації, волатильність.

Yeskina A., Kulakovska I. Study of the intelligent system of classification of currency volatility according to the results of machine learning. The process of classification and analysis of currency volatility data using machine learning methods is considered. Machine learning models and methods are studied, namely the naive Bayesian classifier and the support vector method for solving classification problems based on the generated data set. The purpose of the study is to develop and research an intelligent system that is able to effectively classify the level of volatility in currency markets. The work is aimed at studying different machine learning methods and classifiers based on different measures of similarity, with the aim of determining the optimal approach for volatility classification.

Keywords: intelligent system, machine learning methods, classification, classification quality assessment metrics, volatility.

Постановка наукової проблеми. Фінанси відіграють важливу роль у різноманітних сферах нашого життя, від особистих фінансів кожного з нас до глобальних економічних процесів. Ця сфера вимагає постійного аналізу, прогнозування та прийняття рішень в реальному часі. Водночас, розвиток сучасних технологій, таких як штучний інтелект і машинне навчання, відкриває нові можливості для поліпшення аналізу та прийняття рішень у фінансовій галузі. Крім цього, ураховуючи зростання ролі соціальних мереж у сучасному інформаційному суспільстві, актуальним є вивчення можливостей та обмежень використання цих платформ для класифікації ринкових тенденцій. Визначенно для надійності обраної платформи – Телеграм.

Актуальність дослідження обумовлена важливістю вивчення та аналізу динаміки валютних ринків. З урахуванням нестабільності сучасного економічного середовища та впливу різноманітних факторів, таких як економічні новини, глобальні кризи, та природні катастрофи, розробка інтелектуальної системи для класифікації волатильності валют стає критично важливою.

Для наглядного прикладу проведення аналізу та попередньої обробки був використаний ключовий набір даних USD/UAH, отриманий за допомогою Yahoo Finance API. На основі цього набору даних ми провели докладний аналіз, спрямований на визначення важливих параметрів та особливостей. Головною задачею наших подальших моделей є оцінка рівня волатильності валютної пари USD/UAH та її класифікація на низький чи високий рівень ризику. Обраний датасет відображає динаміку валютного курсу долара США до української гривні протягом останніх п'яти років, з 4 лютого 2019 року по 3 лютого 2024 року [1].

Попередня обробка даних – це процес підготовки та очищення даних перед їх використанням у моделях машинного навчання. Як правило, при обробці ми маємо справу з великими обсягами необроблених вихідних даних. Алгоритми машинного навчання розраховані на те, що, перш ніж вони зможуть розпочати процес тренування, отримані дані будуть відформатовані певним чином. Щоб привести дані до форми, що прийнятна для алгоритмів машинного навчання, ми повинні попередньо підготувати їх і перетворити на потрібний формат [2].

Для класифікації волатильності валют можна обрати різні ознаки, які допоможуть системі визначати та присвоювати рівні волатильності. Проте декілька виділених мною ознак, таких як

процентне змінення та стандартне відхилення, можуть бути особливо значущими. Давайте розглянемо їх більш детально.

Процентне змінення (Percent Change). Ця ознака визначає, наскільки змінилася ціна валюти від відкриття до закриття. Вона розраховується як відсоткова різниця між ціною закриття та ціною відкриття. Високе значення цієї ознаки може свідчити про значний рух цін протягом періоду, що може бути індикатором волатильності.

Стандартне відхилення (Std Deviation). Ця ознака вимірює міру розкиду процентних змін між ціною відкриття та ціною закриття. Вона дозволяє визначити, наскільки великі відхилення від середнього можна спостерігати в рухах цін. Більше значення стандартного відхилення може вказувати на вищий рівень невизначеності та непередбачуваності в ціновому русі.

Для наглядності щодо розподілу кількості класів волатильності був сформований та представлений точковий графік (рис. 1). Ця візуалізація допомагає легше усвідомити співвідношення між високою та низькою волатильністю, надаючи графічну інформацію про це.

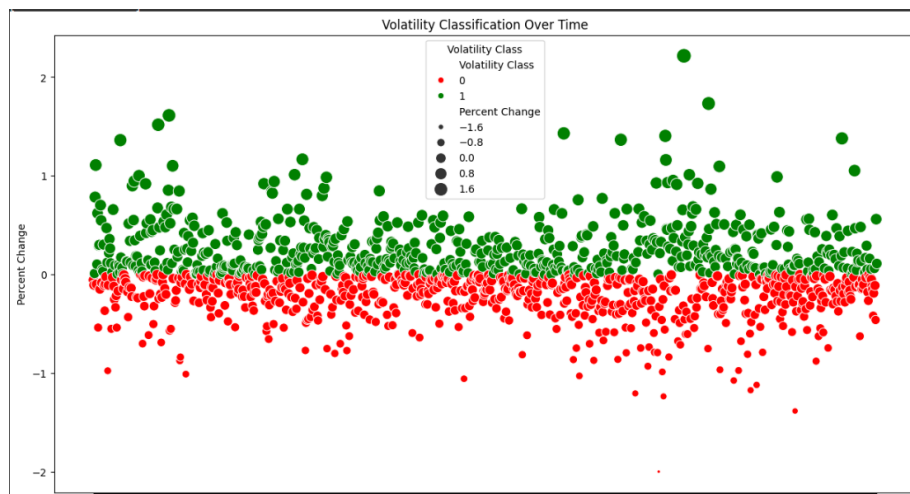


Рис. 1 – Точковий графік співвідношення між класами волатильності

Другий етап підготовки до моделювання включає в себе визначення цільової змінної та визначення ознак, які будуть використовуватися для класифікації [3,4]. На цьому етапі важливо розділити дані на навчальний та тестовий набори для ефективної оцінки моделі. Дотримання балансу між класами (якщо є класифікаційна задача) та врахування стратифікації допомагають уникнути перекосу результатів.

Додатково, проводиться аналіз розподілу класів та визначення метрик якості [3,4], які будуть використовуватися для оцінки ефективності моделі. Наприкінці цього етапу дані готові до подальшого застосування алгоритмів машинного навчання та навчання моделей на навчальному наборі для подальшого тестування та оцінки їхньої точності та ефективності.

Метод відкладених даних (англ. Holdout Method) – це стратегія поділу набору даних на дві неперетинаючі частини: тренувальний (навчальний) та тестовий. Тренувальний набір використовується для навчання моделі, в той час як тестовий набір використовується для оцінки ефективності моделі та перевірки її здатності до узагальнення на нових даних.

Метод відкладених даних дозволяє оцінити роботу моделі на реальних даних, які вона раніше не бачила, та визначити її точність та ефективність перед застосуванням до реальних сценаріїв чи вирішення завдань. У нашому випадку, 80% даних використовується для тренування моделі, а 20% залишається для тестування. Ця процедура забезпечує наявність 1305 записів для моделювання нових прогнозів.

Модель буде навчатися на випадковій вибірці даних за допомогою функції `train_test_split`, де випадковим чином вибираються підмножини даних для тренування та тестування. Випадкова вибірка – це вибір певної кількості об'єктів чи прикладів з набору даних таким чином, що кожен об'єкт має однакову ймовірність бути обраним.

У контексті машинного навчання випадкова вибірка допомагає забезпечити репрезентативність даних для тренування та тестування моделі. Вона дозволяє уникнути впливу порядку даних та забезпечити, що модель отримує різноманітні приклади для навчання, що сприяє

її загальній ефективності та здатності узагальнення на нові дані. Якщо рандомізація виконана правильно, то розподіл класів волатильності в тренувальній та тестовій вибірках є репрезентативним та відображає вихідну кількість об'єктів кожного класу. У нашому випадку/

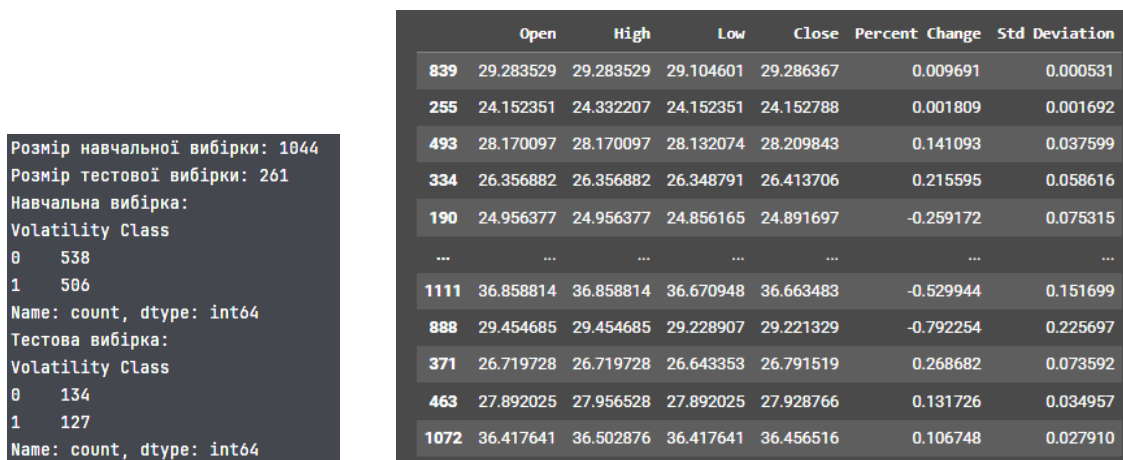


Рис. 2 – Розподілення на тренувальний та тестовий набори

Тепер, коли дані підготовлені та розподілені на тренувальний та тестовий набори (рис. 2), можна приступати до побудови та навчання моделей наївного баєсівського класифікатора і моделі машинного навчання на основі методу опорних векторів (SVM).

Аналіз досліджень. Yahoo Finance API пропонує широкий набір можливостей і функцій, включаючи історичні котирування, новини фондового ринку та вичерпні фінансових даних. Ці дані можна використовувати для створення власних фінансових додатків та аналітичних інструментів, таких як аналіз фондового портфеля, сповіщення про ціни на акції та інші сервіси, що базуються на даних.

Даний портал дозволяє розробникам отримувати доступ до фінансових даних з різних ринків і цінних паперів, включаючи акції, взаємні фонди, опціони та ф'ючерси. Маючи доступ до ринкових даних у режимі реального часу, розробники можуть створювати інтерактивні графіки та торгові системи, здійснювати моніторинг портфельів та аналіз фондового ринку [1].

Методи класифікації машинного навчання – це алгоритми та підходи, які дозволяють автоматично розподіляти об'єкти чи дані у визначені категорії або класи на основі їхніх характеристик чи ознак. Ці методи грають ключову роль у вирішенні задач класифікації та прогнозуванні в багатьох областях, включаючи медицину, фінанси, маркетинг, техніку та інші. Вибір методу класифікації залежить від конкретних цілей і даних, які використовуються.

Деякі з найпоширеніших методів класифікації машинного навчання включають [5, 6]: наївний баєсівський класифікатор (Naive Bayes). Використовує теорему Баєса та припущення про незалежність ознак для класифікації об'єктів; метод опорних векторів (SVM – Support Vector Machines). Шукає оптимальний гіперплощину для розділення об'єктів різних класів у високорозмірному просторі; дерева рішень (Decision Trees). Використовують структуру дерева для послідовного прийняття рішень та класифікації; k-найближчих сусідів (K-Nearest Neighbors). Класифікує об'єкти на основі класів їхніх найближчих сусідів у вхідних даних; логістична регресія (Logistic Regression). Використовує логістичну функцію для моделювання ймовірностей класів; нейронні мережі (Neural Networks). Моделі, які імітують структуру та функції людського мозку для класифікації; градієнтний бустінг (Gradient Boosting). Комбіную декілька слабких моделей для покращення точності. Це лише кілька прикладів, і вибір методу залежить від конкретного завдання, властивостей даних та вимог проекту.

В даному розділі ми розглядатимемо два популярних методи класифікації, а саме: наївний баєсівський класифікатор (NB) та метод опорних векторів (SVM). Ці методи виявляються ефективними у реалізації інтелектуальних систем. Розглянемо за допомогою таблиці ключові аспекти, що роблять ці методи важливими та ефективними.

Таблиця 1 – Порівняльна характеристика наївного баєсівського класифікатора (NB) та методу опорних векторів (SVM)

Аспект	Наївний баєсівський класифікатор (NB)	Метод опорних векторів (SVM)
Простота та швидкість навчання	Висока простота та швидкість, особливо при великих обсягах даних.	Ефективний, але трошки більше обчислювальної витрати.
Робота з великим обсягом даних	Легко масштабується для великих обсягів даних.	Добре працює з об'єктами в великорозмірному просторі ознак.
Ефективність в текстовому аналізі	Застосовується з успіхом у задачах аналізу тексту.	Може бути використаний в аналізі тексту, але вимагає налаштувань.
Розділення класів в неоднорідних даних	Менш ефективний в умовах неоднорідних даних.	Добре працює з неоднорідними та складними даними.
Обмежений обсяг даних	Ефективний навіть при обмеженому обсязі даних.	Надає гарні результати в умовах обмежених ресурсів.

Таблиця 1 відображає порівняльний аналіз наївного баєсівського класифікатора та методу опорних векторів за різними аспектами. Обидва методи мають свої сильні сторони та використовуються в різноманітних сценаріях залежно від вимог та особливостей задачі, тож розглянемо кожен з них більш детально.

Для класифікації волатильності валют за допомогою наївного баєсівського класифікатора та методу опорних векторів (SVM), можна використовувати різні ознаки, такі як процентний скачок, стандартне відхилення і інші метрики, які вказують на ступінь коливань цін.

Мета роботи. Метою дослідження є розробка та реалізація інтелектуальної системи, яка здатна ефективно класифікувати рівень волатильності на валютних ринках. Робота спрямована на вивчення різних методів машинного навчання та класифікаторів на основі різних мір подібності, з метою визначення оптимального підходу для класифікації волатильності та взаємодії з користувачами через Telegram-бот на надання оперативного доступу до порівняння результатів різних методів. Очікується, що розроблена інтелектуальна система стане інструментом для управління ризиками та прийняття рішень на валютних ринках, допомагаючи трейдерам та інвесторам ефективно адаптуватися до змін у рівнях волатильності.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. У якості базових моделей обрано моделі: наївний баєсівський класифікатор (NB) та метод опорних векторів (SVM). Два підходи, зокрема наївний баєсівський класифікатор та SVM, використовуються для класифікації волатильності на низьку та високу. Вони навчаються на тренувальному наборі (1044 записи), який містить дані з відомими класами, і потім випробовуються на тестовому наборі (261 запис) для оцінки ефективності та здатності узагальнення на нових даних.

На завершальному етапі проведемо оцінку ефективності обох моделей за допомогою різних метрик, таких як коефіцієнт успішності, точність, повнота, та інші. Це допоможе визначити, яка модель краще справляється з класифікацією волатильності на тестовому наборі даних.

Наївний баєсівський класифікатор (NB). Ініціалізація та навчання моделі наївного баєсівського класифікатора є важливим етапом у процесі машинного навчання. Використовуючи бібліотеку `scikit-learn` в Python, цей процес може бути легко виконаний за кілька кроків. Спочатку імпортуємо необхідну бібліотеку для наївного баєсівського класифікатора: `from sklearn.naive_bayes import GaussianNB`. Створюємо екземпляр класу `GaussianNB()`, який представляє наївний баєсівський класифікатор з гаусовим розподілом: `nb_classifier = GaussianNB()`. Навчаємо модель на тренувальних даних за допомогою методу `fit`, передаючи тренувальні ознаки та відповідні цільові значення: `nb_classifier.fit(X_train, y_train)`, де `X_train` – це тренувальні ознаки, а `y_train` – це відповідні цільові значення.

Щоб з'ясувати, чи добре навчена модель, необхідно звірити її з даними навчання. Результат отримуємо як таблиці. Результат показує, що помилка (або рівень помилкової класифікації)

становить 65 із 1044, або 6%. Таким чином, точність класифікації на тренувальних даних – 94%. Для відображення точності класифікації тестового набору ми сформуємо і виведемо матрицю невідповідностей (confusion matrix). Це дасть нам детальну інформацію про те, як модель класифікує дані і які помилки вона може допускати. Результати матриці невідповідностей надають можливість обчислити такі метрики, як коефіцієнт успішності (accuracy), повноту (recall), специфічність (specificity), та інші, що допомагає отримати повну картину ефективності класифікатора.

Таблиця 2 – Порівняльна характеристика звітів з класифікації

Звіт з класифікації для навчального набору					Звіт з класифікації для тестового набору				
Звіт з класифікації для навчального набору даних:					Звіт з класифікації для тестового набору даних:				
	precision	recall	f1-score	support		precision	recall	f1-score	support
0	0.90	0.99	0.94	538	0	0.93	0.99	0.96	134
1	0.99	0.88	0.93	506	1	0.99	0.92	0.96	127
accuracy			0.94	1044	accuracy			0.96	261
macro avg	0.94	0.94	0.94	1044	macro avg	0.96	0.96	0.96	261
weighted avg	0.94	0.94	0.94	1044	weighted avg	0.96	0.96	0.96	261
<p>Звіт з класифікації для навчального набору даних відображає високий рівень коефіцієнту успішності моделі, який складає 94%.</p> <p>Для класу 0 (низька волатильність) відзначається висока точність на рівні 90%, що свідчить про ефективність класифікації валютних пар з низькою волатильністю. Для класу 1 (висока волатильність) досягнута вражаюча точність на рівні 99%, що є дуже високим показником. Проте показник повноти (recall) для класу 1 складає лише 88%, що може вказувати на те, що деякі випадки високої волатильності можуть бути пропущені моделлю.</p> <p>Середній гармонічний показник F1-score для обох класів також є високими, становлячи 0,94 для класу 0 і 0,93 для класу 1, що підтверджує ефективність моделі в обох випадках.</p>					<p>Звіт з класифікації для тестового набору даних показує, що коефіцієнт успішності складає 96%.</p> <p>Для класу 0 (низька волатильність) точність становить 93%, а для класу 1 (висока волатильність) – 99%. Показник повноти (recall) для класу 0 складає 99%, а для класу 1 – 92%.</p> <p>Середній гармонічний показник F1-score для класу 0 дорівнює 0,96, а для класу 1 – 0,96. Об'єднана макро-середня точність, повнота і F1-score становить 96%.</p>				

Матриця невідповідностей використовується для визначення точності та частоти помилок. Результат показує, що помилка становить 11 з 261, або 4%. Отже, точність класифікації на тестових даних становить приблизно 96%.

Виведення звіту з класифікації для навчального та тестового набору даних спрямовані на оцінку продуктивності моделі класифікації та надає комплексну інформацію про ефективність моделі. Основні метрики включають precision, recall та f1-score для кожного класу (0 і 1), а також коефіцієнт успішності (accuracy) та середні значення цих метрик.

На основі наведених звітів з класифікації для навчального та тестового наборів даних можна зробити висновок, що модель демонструє досить високу ефективність у класифікації волатильності.

Коефіцієнт успішності (accuracy) для навчального набору даних складає 94%, що означає, що модель правильно класифікує близько 94% випадків. Для класу 0 точність становить 90%, а для класу 1 – 99%, що свідчить про те, що модель досить добре впоралася з обома класами, хоча точність для класу 1 є незначно вищою. Показники повноти для обох класів також високі (99% для класу 0

та 88% для класу 1), а значення середнього гармонічного показника F1-score підтверджують добру збалансованість моделі.

Для тестового набору даних коефіцієнт успішності становить 96%, що також є досить високим показником. Точність для обох класів є високою, а значення повноти та F1-score також підтверджують добру працездатність моделі на тестових даних.

Отже, можна зробити висновок, що модель є ефективною та добре узгодженою для класифікації волатильності на основі навчального та тестового наборів даних.

Метод опорних векторів. Для навчання моделі методом опорних векторів (SVM) використовується клас SVC (Support Vector Classifier). Цей клас представляє собою реалізацію методу опорних векторів у бібліотеці scikit-learn в Python.

Метод опорних векторів (SVM) - це потужний і ефективний алгоритм машинного навчання, який широко використовується для класифікації та регресії.

Важливою складовою SVM є вибір ядра, яке визначає функцію відстані між точками в просторі ознак. Це важливий параметр, оскільки він визначає, як модель розділить класи.

У бібліотеці scikit-learn можна вибрати різні типи ядер, такі як лінійне, поліноміальне, радіальне базисне функції (RBF) тощо. Параметр ядра можна встановити за допомогою аргумента kernel при ініціалізації об'єкта SVC. Проте у бібліотеці scikit-learn за замовчуванням ядро для методу опорних векторів (SVM) - це радіальне базисне ядро (RBF), також відоме як гаусове ядро. Тому, якщо ви не вказали ядро явно, SVM буде використовувати RBF-ядро для класифікації даних.

Дії для навчання та використання моделі методу опорних векторів (SVM) подібні до дій для наївного баєсівського класифікатора.

Перед тим, як робити прогнози, модель SVM повинна бути навчена на навчальному наборі даних, щоб вона могла зрозуміти взаємозв'язки між ознаками та їх класифікаційні мітки. Цей процес включає в себе побудову оптимальної границі розділення між класами.

Щоб додати порівняння результатів класифікації з даними навчання, використовуємо той самий звіт про класифікацію для навчального набору даних.

Для того щоб з'ясувати, чи добре навчена модель, необхідно звірити її з даними навчання. Результат отримуємо як таблиці. Результат показує, що помилка (або рівень помилкової класифікації) становить 470 із 1044, або 45%. Таким чином, точність класифікації на тренувальних даних – 55%. Для відображення точності класифікації тестового набору ми сформуємо і виведемо матрицю невідповідностей (confusion matrix). Це дасть нам детальну інформацію про те, як модель класифікує дані і які помилки вона може допускати.

Матриця невідповідностей представлена у наступному вигляді. Розглянемо кожне значення більш детально: 120 (Actual Class 0, Predicted Class 0). Це кількість випадків, коли модель правильно визначила, що елементи належать до класу 0 (низька волатильність); 14 (Actual Class 0, Predicted Class 1). Це кількість випадків, коли модель помилилася, визначивши, що елементи належать до класу 1 (висока волатильність), тоді як насправді вони належать до класу 0; 96 (Actual Class 1, Predicted Class 0). Це кількість випадків, коли модель помилилася, визначивши, що елементи належать до класу 0, тоді як насправді вони належать до класу 1; 31 (Actual Class 1, Predicted Class 1). Це кількість випадків, коли модель правильно визначила, що елементи належать до класу 1.

Ці значення допомагають нам зрозуміти, як модель класифікує дані та де вона може робити помилки. Наприклад, можемо побачити, що модель схильна до помилок у визначенні класу 0 (низька волатильність), плутаючи його з класом 1 (висока волатильність), але досить точно визначає клас 1. Результат показує, що помилка становить 110 з 261, або 42%. Отже, точність класифікації на тестових даних становить приблизно 58%.

Виведення звіту з класифікації для навчального та тестового набору даних спрямовані на оцінку продуктивності моделі класифікації та надає комплексну інформацію про ефективність моделі. Основні метрики включають precision, recall та f1-score для кожного класу (0 і 1), а також коефіцієнт успішності (accuracy) та середні значення цих метрик.

Accuracy (точність) – це метрика, що оцінює загальну ефективність моделі без розгляду на розподіл класів. Вона визначає відсоток правильно класифікованих екземплярів серед усіх екземплярів у тестовому наборі даних.

Чим вищі значення цих метрик, тим краще модель класифікує валютні пари. Нижче наведена порівняльна таблиця з результатами моделювання і звіти з класифікації.

Таблиця 3 – Порівняльна характеристика звітів з класифікації

Звіт з класифікації для навчального набору даних показує, що коефіцієнт точності моделі складає 55%, що вказує на помірний рівень правильних прогнозів.					Звіт з класифікації для тестового набору даних показує, що коефіцієнт точності моделі складає 58%, що вказує на помірний рівень правильних прогнозів.				
Звіт з класифікації для навчального набору даних:					Звіт з класифікації для тестового набору даних:				
	precision	recall	f1-score	support		precision	recall	f1-score	support
0	0.54	0.91	0.68	538	0	0.56	0.90	0.69	134
1	0.64	0.16	0.26	506	1	0.69	0.24	0.36	127
accuracy			0.55	1044	accuracy			0.58	261
macro avg	0.59	0.54	0.47	1044	macro avg	0.62	0.57	0.52	261
weighted avg	0.59	0.55	0.48	1044	weighted avg	0.62	0.58	0.53	261
<p>Для класу 0 (низька волатильність) відзначається нижча точність на рівні 54%, що свідчить про те, що модель має тенденцію помилитися у визначенні валютних пар з низькою волатильністю.</p> <p>Для класу 1 (висока волатильність) точність становить 64%, що є трохи вище, але цей клас має низький показник повноти (recall) на рівні 16%, що може вказувати на те, що багато випадків високої волатильності помічено як низька.</p> <p>Середній гармонічний показник F1-score для обох класів також є низькими, становлячи 0,68 для класу 0 і 0,26 для класу 1, що підкреслює помірну ефективність моделі.</p>					<p>Для класу 0 (низька волатильність) відзначається помірна точність на рівні 56%, що свідчить про ефективність класифікації валютних пар з низькою волатильністю.</p> <p>Для класу 1 (висока волатильність) точність становить 69%, що трохи вище, але цей клас має низький показник повноти (recall) на рівні 24%, що може вказувати на те, що багато випадків високої волатильності помічено як низька.</p> <p>Середній гармонічний показник F1-score для обох класів також є низькими, становлячи 0,69 для класу 0 і 0,36 для класу 1, що підкреслює помірну ефективність моделі.</p>				

Обидва звіти з класифікації показують, що модель має проблеми з коректною класифікацією високо-волатильних валютних пар. Хоча точність класифікації для низько-волатильних валютних пар є вищою, для високо-волатильних валютних пар вона значно нижча. Модель схильна до помилкового класифікування високо-волатильних валютних пар як низько-волатильних, що підтверджується низьким показником recall для класу 1. Загальна ефективність моделі оцінюється за допомогою середнього гармонічного показника F1-score, який для високо-волатильних валютних пар є низьким у обох випадках. Ці результати вказують на те, що модель потребує подальшої оптимізації або використання інших методів для покращення її здатності до класифікації високо-волатильних валютних пар.

Порівняльний аналіз роботи моделей класифікаторів

Для прийняття рішення про кращу модель для розв'язання задачі класифікації волатильності валют порівнюються результати моделювання, зібрані у таблицю 4. Для оцінки ефективності кожної моделі порівнюються метрики precision, recall, F1-score для обох класів волатильності (низька та висока) та accuracy.

Таблиця 4 – Порівняльна таблиця методів класифікації

Тип моделей	Коефіцієнт успішності (accuracy)	Точність (precision) клас 0	Точність (precision) клас 1	Повнота (recall) клас 0	Повнота (recall) клас 1	F-міра (F-measure) клас 0	F-міра (F-measure) клас 1
Наївний Баєсів	0.96	0.93	0.99	0.99	0.92	0.96	0.96
Метод опорних векторів	0.58	0.56	0.69	0.90	0.24	0.69	0.36

За результатами оцінки ефективності моделей для задачі класифікації волатильності валют, найкращою моделлю виявився наївний баєсівський класифікатор. Це відображено вищеуказаними метриками ефективності. Наївний Баєсів класифікатор демонструє кращі значення коефіцієнта успішності (accuracy), точності (precision), повноти (recall), а також F-міри (F-measure) порівняно з методом опорних векторів. Таким чином, на основі сукупності цих показників обрано наївний баєсівський класифікатор як кращу модель для розв'язання задачі класифікації волатильності валют.

Реалізація обгортки інтелектуальної системи – Телеграм-боту

Інтелектуальна система класифікації волатильності валют, реалізована у вигляді телеграм-бота, представляє собою комплексний інструмент для аналізу та класифікації волатильності на ринку валют. Цей бот був розроблений використовуючи набір методів та технологій, які описані раніше у даній статті.

Нарешті, після побудови та валідації моделі, інтелектуальна система була успішно впроваджена в середовище телеграм-боту з метою надання користувачам зручного та ефективного інструменту для відстеження та аналізу валютних ринків. Використання телеграм-боту дозволяє забезпечити простий та доступний інтерфейс для користувачів, які можуть отримувати актуальну інформацію та аналіз валютних курсів у режимі реального часу.

Основні переваги впровадження інтелектуальної системи в телеграм-бот включають:[7]

1) зручний доступ. Користувачі можуть отримувати інформацію про волатильність валют та їх класифікацію просто взаємодіючи з телеграм-ботом. Це робить процес отримання аналітичних даних більш зручним та ефективним;

2) регулярні оновлення. Система автоматично оновлює та надсилає інформацію про волатильність валютних пар, що дозволяє користувачам залишатися в курсі останніх подій на ринку;

3) інтерактивні функції. Телеграм-бот може взаємодіяти з користувачами, надаючи їм можливість отримати додаткову інформацію та взаємодіяти з функціоналом системи;

4) персоналізований досвід. Користувачі можуть налаштовувати отримання інформації відповідно до своїх індивідуальних потреб та вибирати конкретні валютні пари для відстеження.

Це робить високотехнологічну систему класифікації волатильності валют більш доступною та корисною для широкого кола користувачів через інтерфейс телеграм-боту.

Реєстрація телеграм-бота є основою у процесі розробки обгортки для інтелектуальної системи класифікації волатильності валют. Цей процес полягає у створенні та налаштуванні бота за допомогою спеціальних засобів, наданих платформою Telegram [7].

Перший крок - це створення нового бота в Telegram через спеціального бота-конструктора @BotFather. BotFather - це офіційний бот Telegram для створення та налаштування інших ботів. Користувач відправляє команду /newbot та слідує інструкціям для створення нового бота, вибору його імені та отримання токена доступу [7].

Команда "/mybots" використовується для перегляду списку ботів, які ви створили або адмініструєте в Telegram. Ця команда допомагає користувачам швидко отримати доступ до управління своїми ботами та їх налаштуваннями. На рис. 3. відображено процес завантаження фото профілю для бота.

Структура проекту телеграм-боту – це організація файлів та коду, що визначає, як різні частини функціональності боту організовані та взаємодіють між собою. Нижче наведено загальну структуру проекту (рис. 3).

Структура проекту телеграм-боту (рис. 3) може включати в себе різні компоненти, які спільно забезпечують його функціональність та ефективність. Для початку роботи з чат-ботом необхідно надіслати команду "start". Ця команда ініціює взаємодію з ботом і дозволяє користувачеві отримати доступ до основного функціоналу.

Основний функціонал телеграм-бота включає наступні запити, які формують основу нашої системи.

- 1) Поняття волатильності. Цей запит дозволяє користувачам отримати роз'яснення щодо поняття волатильності на ринку валют. Це може включати пояснення, що таке волатильність, як вона вимірюється, і чому вона важлива для трейдерів та інвесторів;
- 2) Короткий FAQ по користуванню системою. Цей запит надає короткий перелік питань і відповідей щодо використання системи;

- 3) Класифікація волатильності валют. Цей запит дозволяє користувачам отримати класифікацію рівнів волатильності різних валютних пар (заздалегідь вибравши необхідні параметри роботи системи). Включає інформацію про стабільні та волатильні валютні пари, їх характеристики та фактори, що впливають на волатильність;
- 4) Відстеження графіків валютних пар в реальному часі. Цей запит дозволяє користувачам переглядати графіки валютних пар в реальному часі. Реалізація проводилась через інтеграцію з фінансовими API для отримання актуальних даних про ціни валют та їх зміни у часі.

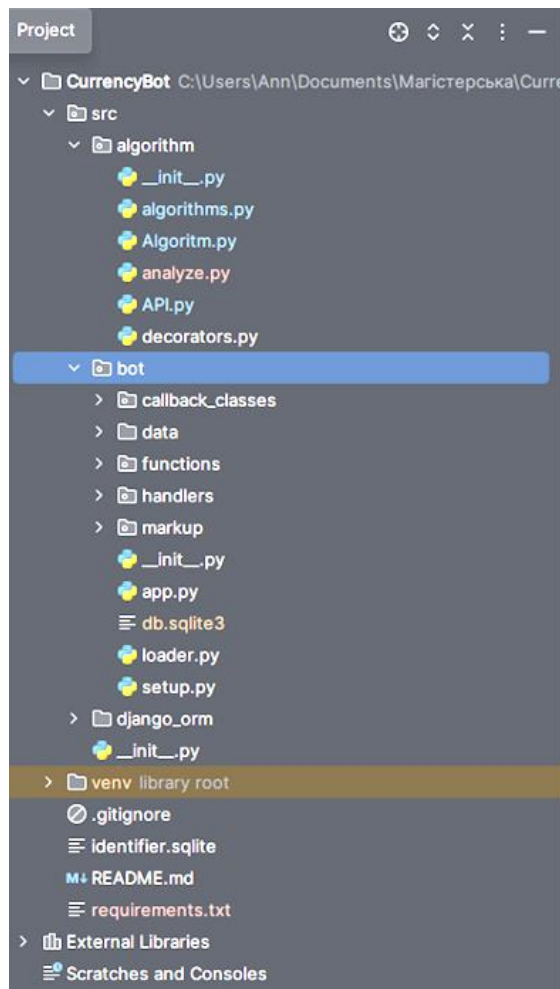
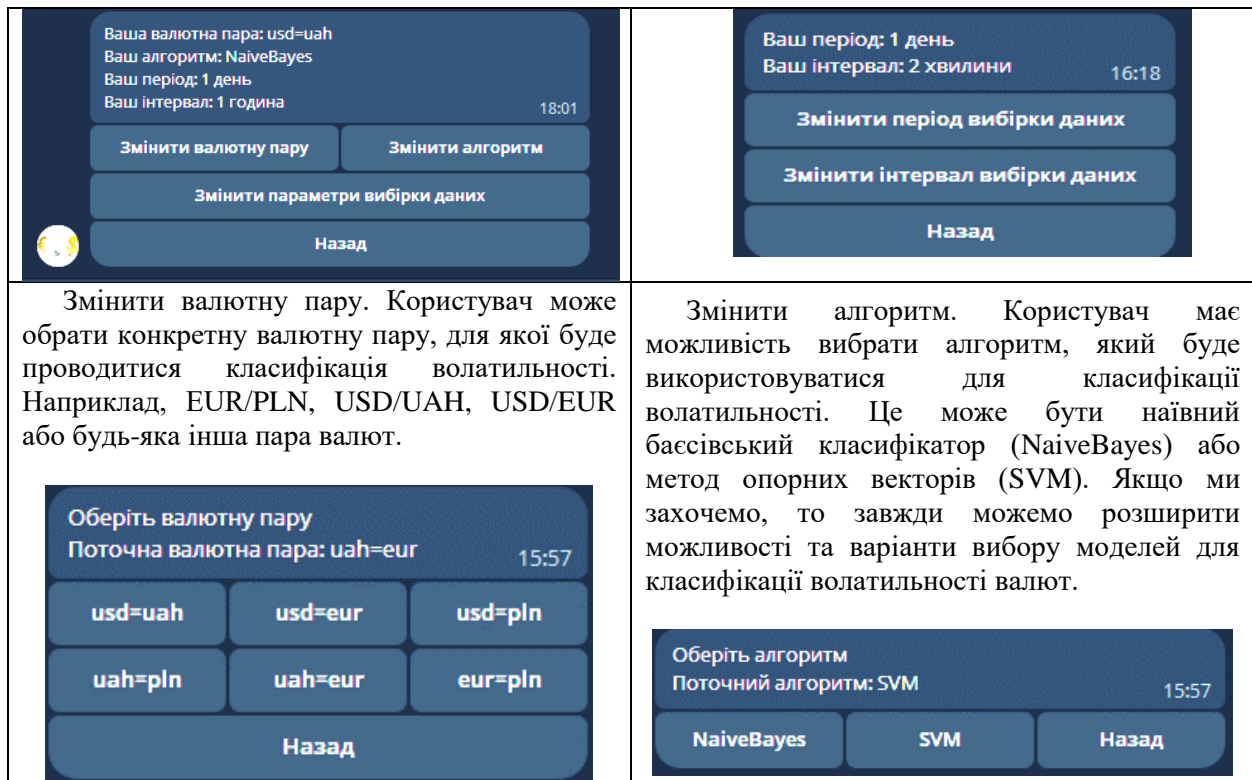


Рис 3. – Структура проекту телеграм-бота

Ці запити становлять основу функціоналу чат-бота і дозволяють користувачам отримувати широкий спектр інформації щодо волатильності валют та відстежувати її динаміку у реальному часі. Тож розглянемо їх більш детально.

Таблиця 5 – Функціонал чат-бота

<p>При виборі запиту "Класифікація волатильності валют" користувач отримує доступ до меню з наступними опціями:</p>	<p>Змінити параметри вибірки даних. Перед нами відкривається меню, де користувач може переглянути обрані параметри та змінити період або інтервал вибірки даних. Це меню надає користувачеві можливість налаштувати параметри класифікації волатильності валют відповідно до їхніх потреб та вимог.</p>
---	---



Відтепер користувач може зручно відстежувати динаміку волатильності валютних пар та отримувати об'єктивну оцінку ефективності використовуваної моделі класифікації. Цей інструментарій дозволяє здійснювати оперативний аналіз ринкових умов та приймати обгрунтовані рішення з максимальною ефективністю.

На рисунку 4 представлена таблиця з даними за період 24 січня з інтервалом оновлення кожні 2 хвилини, яку використовує наївний баєсовий класифікатор для прогнозування волатильності валютних пар. Таблиця містить різні стовпці, які дозволяють відслідковувати зміни волатильності на ринку. Кожен рядок таблиці представляє собою окремий часовий відрізок, під час якого здійснювалися виміри волатильності для певних валютних пар.

Ця таблиця дозволяє нам аналізувати та візуалізувати динаміку волатильності на ринку, а також оцінювати ефективність прогнозування, здійсненого наївним баєсовим класифікатором. Шляхом вивчення цих даних можна зробити висновки щодо того, наскільки ефективно модель прогнозує волатильність та якість її прогнозів.

The screenshot shows a table with columns: Datetime, Open, Close, High, Low, Percent Change, and Volatility Class. Below the table are performance metrics for a GaussianNB model.

Datetime	Open	Close	High	Low	Percent Change	Volatility Class
2024-01-24 15:12:00	37.877	37.857	37.877	37.877	-0.051	0
2024-01-24 15:14:00	37.930	37.877	37.930	37.810	-0.140	0
2024-01-24 15:16:00	37.849	37.810	37.849	37.757	-0.102	0
2024-01-24 15:18:00	37.749	37.757	37.811	37.749	0.019	1
2024-01-24 15:20:00	37.700	37.708	37.956	37.700	0.020	1
2024-01-24 15:22:00	37.892	37.956	37.892	37.867	0.170	1
2024-01-24 15:24:00	37.879	37.867	37.923	37.879	-0.030	0
2024-01-24 15:26:00	37.812	37.923	37.812	37.617	0.292	1
2024-01-24 15:28:00	37.597	37.617	37.597	37.597	0.054	1
2024-01-24 15:30:00	37.590	37.524	37.632	37.590	-0.177	0

Accuracy	Precision	Recall	Specificity	F1 Score	Roc Auc Score
0.887446	0.857143	0.688525	0.958824	0.763636	0.823674

Рис. 4 – Дані за період 24 січня з інтервалом оновлення 2 хвилини

Важливо відзначити, що графіки будуть відображати динаміку вартості саме тих валютних пар, які ви вибрали для класифікації. Це дозволяє вам не лише відстежувати зміни на ринку у

реальному часі, але й порівнювати їх з результатами класифікації, щоб з'ясувати, наскільки ефективно ваша модель прогнозує рухи цих валютних пар. Такий підхід надає вам цінну інформацію для прийняття рішень на основі аналізу ринку та результатів прогнозування.

За допомогою цієї системи користувачі можуть ефективно аналізувати волатильність валютних пар, відстежувати їхню динаміку та здійснювати точні прогнози майбутніх рухів цін. Крім того, система дозволяє користувачам швидко та зручно змінювати параметри аналізу, вибирати різні алгоритми класифікації та переглядати реальний часовий графік зміни курсу валют.

Висновки. Результатом дослідження є розробка та впровадження інтелектуальної системи, яка здатна ефективно класифікувати рівень волатильності на валютних ринках. Ключовим етапом в розробці системи було попереднє оброблення даних, яке включало в себе очищення, нормалізацію та інші методи підготовки даних для моделювання. Далі проводився аналіз та порівняльне дослідження різних моделей класифікаторів для вибору найефективнішої. Також було проаналізовано ефективність реалізованих методів класифікації з використанням метрик; реалізовано обгортку інтелектуальної системи у вигляді телеграм-боту для взаємодії з користувачами та обробки їхніх запитів; інтегровано розроблені моделі в телеграм-бот для автоматичного аналізу запитів користувачів та надання прогнозів; проведено тестування розробленого телеграм-боту для перевірки його працездатності та надійності.

Таким чином, представлена інтелектуальна система класифікації волатильності валют має потенціал стати невід'ємною частиною аналітики фінансових ринків, допомагаючи їй користувачам отримувати важливу інформацію та робити обдумані рішення в умовах постійної динаміки на ринках.

Список бібліографічного опису

1. Yahoo. Yahoo Finance API [Електронний ресурс] / Yahoo – Режим доступу до ресурсу: <https://finance.yahoo.com/quote/UAN%3DX/history>. (дата звернення: 16.12.2023).
2. Preparing Your Dataset for Machine Learning: 10 Basic Techniques That Make Your Data Better. [Електронний ресурс] / altexsoft – Режим доступу до ресурсу: <https://www.altexsoft.com/blog/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/>. (дата звернення: 20.12.2022).
3. Метрики в задачах машинного навчання. [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/company/ods/blog/328372>. (дата звернення: 20.12.2022).
4. Ставицький А.В. Класифікаційні метрики. [Електронний ресурс] / Ставицький А.В. – Режим доступу до ресурсу: http://www.andriystav.cc.ua/Downloads/MITER/Lecture_04.pdf. (дата звернення: 20.12.2022).
5. Classification in Machine Learning: What it is & Classification Models. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.simplilearn.com/tutorials/machine-learning-tutorial/classification-in-machine-learning>. (дата звернення: 30.11.2023).
6. Types of Classification in Mschine Learning. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.knowledgehut.com/blog/data-science/types-of-classification-in-ml>. (дата звернення: 30.11.2023).
7. Telegram FAQ: веб-сайт. URL: <https://telegram.org/faq> (дата звернення: 05.12.2023).

References

1. Yahoo. Yahoo Finance API [Electronic resource] / Yahoo - Resource access mode: <https://finance.yahoo.com/quote/UAN%3DX/history>. (date of application: 16.12.2023).
2. Preparing Your Dataset for Machine Learning: 10 Basic Techniques That Make Your Data Better. [Electronic resource] / altexsoft - Resource access mode: <https://www.altexsoft.com/blog/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/>. (application date: 12/20/2022).
3. Metrics in machine learning tasks. [Electronic resource] – Mode of access to the resource: <https://habr.com/ru/company/ods/blog/328372>. (application date: 12/20/2022).
4. Stavitskyi A.V. Classification metrics. [Electronic resource] / Stavitskyi A.V. – Mode of access to the resource: http://www.andriystav.cc.ua/Downloads/MITER/Lecture_04.pdf. (application date: 12/20/2022).
5. Classification in Machine Learning: What it is & Classification Models. [Electronic resource] - Resource access mode: <https://www.simplilearn.com/tutorials/machine-learning-tutorial/classification-in-machine-learning>. (application date: 11/30/2023).
6. Types of Classification in Machine Learning. [Electronic resource] - Resource access mode: <https://www.knowledgehut.com/blog/data-science/types-of-classification-in-ml>. (application date: 11/30/2023).
7. Telegram FAQ: Website. URL: <https://telegram.org/faq> (access date: 12/05/2023).