Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий(магістерський)

Метод прогнозування курсу валют
за допомогою нейронної мережі

(тема)

Виконав:

студент II курсу, групи СПм-20-1

Корягіна П.О
(прізвище, ініціали)

Спеціальність

123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма

Системне програмування
(повна назва освітньої програми)

Керівник: проф. Кучук Н.Г.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Корягіній Поліні Олександрівні
(прізвище, ім’я, по батькові)

1. Тема роботи Метод прогнозування курсу валют за допомогою нейронної мережі

затверджена наказом по університету від “ 5 ” листопада 2021 р. № 1657 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 13 грудня 2021 р.

3. Вхідні дані до роботи Значення курсів долара США та євро за НБУ

4. Перелік питань, що потрібно опрацювати у роботі

Аналіз предметної області

Опис базових понять, інструментів створення нейронних мереж

Аналіз попередніх досліджень

Розробка нейронних мереж

Аналіз результатів прогнозування на прикладі курсів долару США та євро за НБУ

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Слайди презентації

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної області	09.11.2021-15.11.2021	
2	Створення нейронних мереж	16.11.2021-20.11.2021	
3	Створення ансамблів	21.11.2021 – 25.11.2021	
4	Аналіз отриманих результатів	26.11.2021-30.11.2021	
5	Оформлення матеріалів атестаційної роботи	06.12.2021 – 09.12.2021	
6	Подання атестаційної роботи керівникові та	10.12.2021.-10.11.2021	
	її попередній захист		
7	Подання роботи на рецензування	12.12.2021 – 13.12.2021	

Дата видачі завдання 08 листопада 2021 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Кучук Н.Г.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 67 с., 12 рис., 2 табл., 2 дод., 22 джерел.

ШТУЧНА НЕЙРОННА МЕРЕЖА, ШНМ, LSPM, GRU, RNN, ОПТИМІЗАТОР, КУРС ВАЛЮТ, ОБМІННИЙ КУРС, АНАЛІЗ, ТОЧНІСТЬ, АНСАМБЛЬ, РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ

Метою кваліфікаційної роботи с розробка методики прогнозування курсів валют за допомогою нейронних мереж.

У ході виконання кваліфікаційної роботи було розглянуто основні поняття про валютний курс, основні поняття про нейронні мережі, проаналізовано попередні дослідження на сходні теми. Було розглянуто доступні архітектури нейронних мереж, їх переваги і недоліки. Представлено алгоритм створення і оцінки точності нейронних мереж та їх ансамблів. Проаналізовано результати дослідження

ABSTRACT

Master’s thesis: 67 pages, 12 figures, 2 tables, 2 appendices, 22 sources.

ARTIFICIAL NEURAL NETWORK, SNM, LSPM, GRU, RNN, OPTIMIZER, EXCHANGE RATE, EXCHANGE RATE, ANALYSIS, ACCURACY, ACCURACY, ENSEMBLE, RECURRENCE NURSERY

The purpose of the qualification work is to develop a methodology for forecasting exchange rates using neural networks.

In the course of the qualification work the basic concepts of the exchange rate, the basic concepts of neural networks were considered, previous studies on similar topics were analyzed. Available neural network architectures, their advantages and disadvantages were considered. An algorithm for creating and estimating the accuracy of neural networks and their ensembles is presented. The results of the research are analyzed

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ШНМ – штучна нейронна мережа

MSE – середньоквадратична похибка

ARIMA – авторегресійна інтегрована ковзна середня

RNN – рекурентна нейронна мережа

LSTM – модель довгої короткострокова пам’яті

GRU – мережа закритого повторного блоку

Adagrad – адаптивний градієнтний спуск

RMSprop – середньоквадратичное поширення

Adam – оцінка адаптивного моменту

Nadam – оцінка адаптивного моменту з імпульсом Нестерова

ReLU – випрямлений лінійний блок

LeakyReLU – випрямлений лінійний блок з витоком

ВСТУП

Наразі, неможливо уявити функціонування суспільства без грошей. Завдяки їм ми оцінюємо речі, послуги і, навіть, час та людські життя. І оскільки, гроші не є універсальними, валютний курс допомагає надати ціну і їм, виразивши її у валюті іншої країни. Валютний курс є інструментом міжнародного ринку і економіки, тож його прогнозування є питанням не тільки математичним , а й економічним та соціальним.

Валютний курс виконує низку важливих функцій – за допомогою курсу долається економічне обмеження національної валюти, з його допомогою виконуються порівняння і нормалізація цін, виконується порівняння вартісних показників країн, та виводиться на світовий ринок національний продукт. Отже, валютний курс виконує чималу роль у міжнародних відносинах і на міжнародному ринку, тож його прогнозування цікавить багатьох людей – від економістів, для того щоб прогнозувати поведінку ринку та можливі ризики, до звичайного обивателя, що зацікавлений у продажі і покупці валюти.

До того ж, методи прогнозування валютного курсу і їх удосконалення можуть бути поштовхом для детальніших взаємодій економіки та комп’ютерної інженерії. Виявлення деяких основних закономірностей і удосконалення методології зможуть, у певний час, створити достатньо чіткий механізм прогнозування валютного курсу. А також, оскільки валютний курс є багато зіставною складною величиною методи його прогнозування можуть бути використані і у інших сферах, у яких необхідно прогнозувати складні та багатофакторні явища

Наразі, висуваються достатньо велика кількість теорій про прогнозування курсу валют, але жодна з них не є досконалою. Багато з них створенні для прогнозування курсу однієї валюти, і оптимізовані лише для її прогнозів.

Основною метою даної роботи є розробка такого підходу, що можна буде використати з достатньою стабільністю більш ніж для однієї валюти. Підходу, що передбачає можливість масштабування, додаткової оптимізації, перенесення на інші платформи і набори даних.

Для досягнення цієї мети перш за все необхідно виявити основні особливості валютних курсів, фактори формування обмінних курсів та економічні методи їх прогнозування, ознайомитись з основними поняттями про нейронні мережі, та з вже існуючими методами прогнозування валютного курсу за допомогою нейронних мереж. Необхідно виявити особливості доступних даних, і спираючись на це знання проаналізувати вже існуючі рішення, та сформувати оптимальний підхід для їх прогнозування. Перевірити підхід на практиці.

1 ПОСТАНОВКА ЗАДАЧІ ТА АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Актуальність прогнозування валютного курсу

Дана робота присвячення методології прогнозування валютного курсу за допомогою нейронних мереж, тож з першу доцільно визначити основні поняття про валютний курс у розрізі економіки і соціального життя. За визначенням, валютний курс є вираженням ціни грошових одиниць однієї країни в грошових одиницях іншої[2]. І з цього визначення можемо припустити, що валютний курс є складним механізмом регулювання міжнародних відносин.

Валютний курс виконує низку важливих функцій – за допомогою курсу долається економічне обмеження національної валюти, з його допомогою виконуються порівняння і нормалізація цін, виконується порівняння вартісних показників країн, та виводиться на світовий ринок національний продукт. Отже, валютний курс виконує чималу роль у міжнародних відносинах і на міжнародному ринку, тож його прогнозування цікавить багатьох людей – від економістів, для того щоб прогнозувати поведінку ринку та можливі ризики, до звичайного обивателя, що зацікавлений у продажі і покупці валюти. І може бути використано в низці областей людського життя тісно пов’язаних з валютними операціями. Прогнозуючи курс можливо прогнозувати прибуток від довгострокові та короткострокові вклади, зменшити втрати на валютних операціях. У більш широкій перспективі, прогнозування курсу дає можливість передбачати та корегувати деякі процеси на міжнародному ринку.

Отже, дуже багато людей, не залежно від сфери діяльності зацікавленні у прогнозуванні обмінних курсів. Тож, розглянемо, для подальшого дослідження результати попередніх робіт та базові поняття про валютний курс.

1.2 Аналіз попередніх досліджень

Перш ніж починати формування і опис методології прогнозування валютного курсу за допомогою нейронних мереж, необхідно ознайомитись з попередніми дослідженнями пов’язаними з прогнозуванням обмінного курсу.

Також, зазначається[15], що коливання курсу, все ж має чіткі відстежуванні залежності. У статті Ппаселлі (2011) було підтверджено, що ціноутворення на фінансовому ринку не є цілком випадковим.

Згідно з результатами огляду попередніх досліджень прогнозування валютного курсу з використанням багатофакторних методів точніш, ніж при використанні одномірного ряду. Так за результатами роботи «Нейронні мережі для прогнозування курсу долара з використанням астрологічних циклічних індексів Гюшон і Ганю»[9] прогнозування курсу валют на місяць вперед має похибку у 1,34402 % проценти. У тій самій роботі зазначається, що зі збільшенням періоду, на який необхідно прогнозувати, точність падає до показників випадкового вгадування – 50% та нижче. Тоді як у роботі «Прогнозування обмінних курсів за допомогою штучних нейронних мереж»[6], у якій для навчання нейронних мереж було використано одномірний часовий ряд точність прогнозування складає від 62% до 76%.

В у той самий час, к показано[13][14] не лінійні ШНМ демонструють вищу ефективність у прогнозуванні курсу валют, порівняно з лінійними моделями, а саме з моделлю Garch. Також, як можна побачити[3][13] повно зв’язні нейронні мережі повного сполучення, або FNN, показують задовільні результати у процесі прогнозування валютного курсу.

У роботі Хансен (1992) показав, що прості ансамблі можуть виконувати кращі узагальнення ніж, одна ШНМ, а підмножина усіх нейронних мереж здібна досягти кращих результатів ніж одна мережа. Крім того, Адхікарі (2013) показав у своїй роботі, що у розрізи прогнозування ШНМ значно кращі і вдосконалюється за допомогою ансамблевого методу.

У роботі «Ансамбль тимчасових загорткових і нейронні мережі

довготривалої пам’яті застосовувати для прогнозування обмінного курсу USDCOP» показано ефективність ансамблю мереж довгострокової короткої пам’яті для прогнозування валютного курсу[15].

Отже, усе вище сказане надає нам підстави вважати, шо у подальшому, доцільно буде використовувати для прогнозування саме ансамбль нейронних мереж, а також допускає використання простих ансамблів як способу підвищення точності.

1.3 Поняття про валютний курс

Як вже було зазначено вище, валютний курс є інструментом регулювання фінансових відносин між країнами, і у країні. Розрізнять реальний і номінальний курси, різниця їх у тому, що реальний ефективний курс є розрахунком конкурентоспроможності вітчизняних товарів стосовно товарів країн-партнерів.

Зазвичай у країнах існує одночасно декілька режимів валютного курсу. Для того, щоб уникнути непорозумінь у подальшому, для прогнозування валютного курсу, як вхідні данні було обрано данні за Національним банком України – такі данні є офіційними, не включають в себе комісій банку та доданої вартості, не залежать від розміру установи і конкуруючих установ, як то буває для обмінників.

Валютний курс є багатофакторним явищем, тісно пов’язаним з іншими економічними категоріями. Основними факторами формування валютного курсу є: темп інфляції, стан платіжного балансу країни, різниця процентних ставок у різних країнах, діяльність фінансових ринків та спекулятивні операції, ступінь використання певної валюти на світовому фінансовому ринку, прискорення або затримка міжнародних платежів, ступінь довіри до валюти на національному та світовому ринках, ринкове та державне регулювання валютного курсу.[2] Важливо зазначити, що без глибокого економічного дослідження неможливо визначити, який з факторів найбільше

впливає на ту чи іншу валюту, тож спиратися на них у даній роботі не доцільно.

Основними методами прогнозування валютного курсу є модель паритету купівельної спроможності, модель відносної економічної сили, економетричні моделі, моделі часового ряду.[2] Лише моделі часового ряду є повністю технічним підходом у прогнозування валют. Тож, з усього вище зазначеного, випливає, що прогнозування курсу на далі буде розглянуто як прогнозування часового ряду.

1.4 Базові поняття про часовий ряд

Часовий ряд – послідовність досліджуваної ознаки, впорядкована у хронологічному порядку. Окремі спостереження часового ряду – його рівні або елементи, кожен з них відповідає певному моменту часу. Якщо кожному моменту часу відповідає лише один показник – ряд є одновимірним, інакше – багатовимірним.

Важливо пояснити, що є основними компонентами часового ряду. Довгострокова тенденція — це загальний напрямок даних, отриманих без урахування будь-яких короткострокових ефектів, таких як сезонні коливання або шум.

Сезонність відноситься до періодичних коливань, які повторюються протягом усього періоду часового ряду.

Стаціонарність є важливою характеристикою часових рядів. Часовий ряд називається стаціонарним, якщо його середнє значення, дисперсія та ко варіація не зазнають значних змін з часом. Існує багато перетворень, які можуть виділити стаціонарну частину нестаціонарного процесу.

Кожен набір даних має шум, який відноситься до випадкових коливань або варіацій через неконтрольовані фактори.

Автокореляція – це кореляція між часовим рядом і його версією з затримкою, і використовується для визначення сезонності та тенденції в

даних часових рядів.

Отже, для подальшого аналізу ми маємо одновимірний ряд, що має тенденцію до зростання, йому притаманні деякі сезонні коливання – щороку, ми можемо відстежити зниження курсу закордонних валют в літку і їх зростання взимку, він не є стаціонарним, є достатньо зашумленним.

Тож, ми маємо достатньо складний набір даних, і до того ж, достатньо обмежений. У подальшому, однією з основних задач буде віднайти такий метод прогнозування часових рядів, який здатен достатньо ефективно оброблювати такі данні, можливі довгострокові відношення у цих даних.

1.5 Основні припущення та вимоги до методу

Першим і основним припущенням стосовно методології прогнозування валютного курсу є точність прогнозування. Жоден з методів не може гарантувати точність до однієї копійки. Тож, засновуючись на тому, наскільки варіативним є обмінний курс, і на те, що будь які валюті операції включають у себе додаткову вартість, додану банком, будемо вважати відхилення у 10 копійок в меншу або в більшу сторону допустимим у прогнозуванні валютного курсу.

Другим припущенням є демонстрація методики на короткостроковому прогнозуванні. Спираючись на особливості даних і їх зміни, доцільно спершу дослідити можливість прогнозувати обмінний курс на наступну дату.

Маючи припущення, сформуймо основні вимоги до підходу. Основними вимогами до підходу є:

- можливість методу знаходити складні залежності у даних та зберігати їх;
- можливість знаходити сезонні кореляції;
- можливість використовувати один підхід до прогнозування для різних даних;
- можливість у подальшому збільшувати діапазон прогнозування;

- не чутливість до пропущених даних;
- можливість масштабування;
- можливість додаткової оптимізації;
- точність не нижча за 60% при використанні даних з великою кількістю коливань;
- точність не нижча 80% на даних з середньою кількістю коливань.

Тож, спираючись на данні вимоги до методології сформуємо далі основне бачення моделі, алгоритми роботи, підготуємо набори даних.

1.6 Основні гіпотези стосовно методу

Наданим вимогам, за винятком точності, що повинна бути досліджена емпірично, відповідають рекурентні нейронні мережі, а саме їх варіації LSTM та GRU.

Згідно з дослідженнями[15],ансамблі нейронних мереж показують вищу точність у порівнянні з одиничними нейронними мережами. Але, для того, щоб сформувати успішний ансамбль з однакових за структурою нейронних мереж, необхідно навчити їх на різних наборах даних. Як можна побачити з аналізу доступних даних – навчати ансамблі на різних наборах не вийде – кількість записів про валютний курс чітко обмежена. Наприклад, розмір вибірки даних про курс євро ледь перевищує шість тисяч одиниць, що е достатньо малою вибіркою.

Першим припущенням стосовно цієї методології буде припущення про те, що для підвищення точності доцільно використовувати ансамблі різних за структурою нейронних мереж.

Вище було описано декілька методів створення ансамблів, а саме ансамбль конкатенації, усереднений ансамбль і середньозважений ансамбль. Доцільно буде розглянути на практиці кожен з наведених видів ансамблів, для визначення оптимального рішення.

Отже, для даного дослідження необхідно навчити декілька нейронних

мереж кожного типу, оцінити їх не залежно одна від одної, і тільки після цього, сформувати та оцінити ансамблі нейронних мереж.

Очікується що даний алгоритм зменшить похибки окремих нейронних мереж, і вищу точність ніж окремі нейронні мережі.

1.7 Нейронні мережі

Штучні нейронні мережі, або ШНМ – це математична модель, а також її програмне або апаратне втілення, побудоване за принципом організації та функціонування біологічних нейронних мереж – мереж нервових клітин живого організму[7].

Нейронні мережі складаються з вузлів що місять вхідний, прихований та вихідний шари. Кожен нейрон за певною схемою з’єднується з іншими. Нейрон складеться з входів, суматора, функції активації та виходу. Результати навчання, та ваги, що нейрон передають надалі залежать від функції активації.

Нейронні мережі покладаються на навчальні дані, щоб з часом вивчати та покращувати свою точність. Однак, як тільки ці алгоритми навчання будуть точно налаштовані на точність, вони стануть потужними інструментами в інформатиці та штучному інтелекті, що дозволить класифікувати та перетворити на кластери дані з високою швидкістю.

Кожен окремий вузол можна вважати моделлю лінійної регресії, що складається з вхідних даних, вагових коефіцієнтів, зміщення, або порогу, та вихідних даних.

Після визначення вхідного шару призначаються ваги. Ці коефіцієнти допомагають визначити важливість будь-якої даної змінної, причому більші з них роблять більш значний внесок у результат порівняно з іншими вхідними параметрами. Потім усі вхідні дані помножуються на відповідні ваги, а потім підсумовуються. Після цього вихідні дані передаються через функцію активації, яка визначає вихід. Якщо цей вихід перевищує заданий поріг, він

активує вузол, передаючи дані наступному шару мережі. Це призводить до того, що вихід одного вузла стає на вхід наступного вузла. Цей процес передачі даних від одного шару до наступного визначає цю нейронну мережу як мережу прямого зв’язку. Точність навчання моделі можна оцінити за за допомогою функції витрат. Це також зазвичай називають середньоквадратичною помилкою або MSE. У рівнянні нижче:

$$MSE = \frac{1}{2m}\sum_{i=1}^{m}(\hat{y} - y)^2$$

i представляє індекс вибірки, $\hat{y}$ – це прогнозований результат, y – фактичне значення, і m – кількість зразків.

Зрештою, мета полягає в тому, щоб мінімізувати нашу функцію витрат, щоб забезпечити правильність відповідності для будь-якого даного спостереження. Коли модель коригує свої ваги та зміщення, вона використовує функцію вартості та навчання з підкріпленням, щоб досягти точки зближення або локального мінімуму. Процес, у якому алгоритм коригує свої ваги, здійснюється шляхом градієнтного спуску, що дозволяє моделі визначати напрямок, який слід рухатися, щоб зменшити помилки (або мінімізувати функцію вартості). З кожним навчальним прикладом параметри моделі коригуються, щоб поступово зближатися до мінімуму.

1.8 Технічні засоби реалізації

Python — це мова програмування високого рівня, яка має багато різних застосувань, включаючи науку про дані та внутрішню веб-розробку. Заснована Python Foundation на початку 1990-х років, це потужний інструмент аналізу даних, який широко використовується в технологіях великих даних. Йому надається особливий статус завдяки великій спільноті розробників машинного навчання, які зосереджені насамперед на швидкозростаючій сфері штучного інтелекту. Завдяки активній спільноті для

Python з’явилося багато готових бібліотек машинного навчання. Ця мова не залежить від платформи, тому її можна адаптувати практично до будь-якої операційної системи. Ще однією перевагою Python є його відкритість – він заснований на технології з відкритим кодом, тому розробники можуть отримати доступ до будь-якої мови. Що стосується недоліків Python, оскільки це мова з динамічним типом, робота з нею може викликати проблеми з машинним навчанням. Однією з них є складність відстеження помилок коду через зростаючу кодову базу програми і, відповідно, її складність. У деяких випадках аудит коду може бути дорогим, трудомістким і вплинути на продуктивність проекту.

Основною, використаною для розробки нашої методології прогнозування валютного курсу бібліотекою є Keras – відкрита бібліотека для створення нейронних мереж. Вона була створена для пришвидшення та спрощення експериментів з мережами глибинного навчання. Це дає нам змогу у даній роботі зосередитись на розробці методології, а не на дублюванні вже існуючих технологій.

1.9 Постановка задачі

Отже, об’єктом цього дослідження, як вже було визначено, є прогнозування валютного курсу. А саме, прогнозування валютного курсу за допомогою нейронних мереж. Як вже було зазначено, дана тема широко досліджувалася і раніше, і у даній роботі було прийнято рішення сконцентрувати увагу на комбінуванні різних мереж одного виду у ансамблі з метою підвищення точності короткострокового прогнозування обмінних курсів. Метою роботи є розробка оптимального методу прогнозування валютного курсую за допомогою нейронних мереж та підвищення точності прогнозування за допомогою об’єднання нейронних мереж у ансамблі нейронних мереж.

Згідно з аналізом досліджених і зазначеними вище джерелами, такий

підхід у перспективі здатен підвищити точність прогнозування. І спираючись на опрацьовані дослідження, нелінійні нейронні мережі мають вищу точність прогнозування, у подібних випадках, хоча їх комбінацій не було знайдено у поданих роботах.

Тож, для досягнення поставленої мети необхідно виконати низку завдань, а саме:

- формувати та обрати такі набори даних, які б показували основні тенденції валютного ринку і його характерні особливості;
- створити моделі штучних нейронних мереж на основі LSTM та GRU;
- навчити моделі на наявних наборах даних;
- створити на їх основі ансамблі нейронних мереж;
- порівняти та проаналізувати отримані результати.

2 МЕТОД ПРОГНОЗУВАННЯ ТА ПІДГОТОВКА ДАННИХ

2.1 Етапи розробки нейронної мережі

Як і для будь який проект, нейронні мережі, незалежно від структури мають свій життєвий цикл. Життєвий цикл звичайного проекту складеться з п’яти фаз: планування, проектування, розробка, експлуатація та підтримка. Ці п’ять фаз залежно від методології роботи з проектом можуть бути застосовані по різному. Життєвий цикл нейронних мереж дещо довший. Він складається з одинадцяти етапів:

- постановка задачі та вибір архітектури нейронної мережі ;
- визначення кількісного і якісного складу входів та виходів;
- формування початкового набору даних;
- попередньої обробки та нормалізації вхідної вибірки;
- поділ вхідної вибірки на навчальну та тестові складові ;
- визначення структури нейронної мережі;
- налаштування параметрів нейронної мережі та алгоритму її навчання;
- навчання нейронної мережі;
- тестування нейронної мережі;
- практичне використання;
- додаткове навчання нейронної мережі.

У даній роботі, крок за кроком буде пройдено описані вище етапи, для того щоб створити і перевірити на практиці метод прогнозування валютного курсу за допомогою нейронних мереж.

Виходячи з цього, увесь цикл починається з аналізу предметної області для якої створено мережу. У нашому випадку це обмінний курс. Надалі ми розглянемо основні особливості предметної області, та визначмо вже існуючі методи прогнозування та шляхи їх удосконалення, розробимо алгоритм

створення і оптимізації мережі, а також розробимо і продемонструймо на практиці отримані результати. Отже, надалі буде розглянуто предметну область, доступні методи прогнозування та технологічні засоби.

2.2 Оцінка часового ряду

Перш ніж складати оптимальний підхід до прогнозування валютного курсу, необхідно визначити і зазначити основні деталі про данні з якими надалі необхідно буде працювати.

Як вже було зазначено, валютний курс є складним багатофакторним явищем. Це призводить до появи складних кореляцій у часовому ряді.

Також, багатофакторність впливає на кількість шумів у даних. Данні залежать від сезону та, факторів які достатньо складно передбачити.

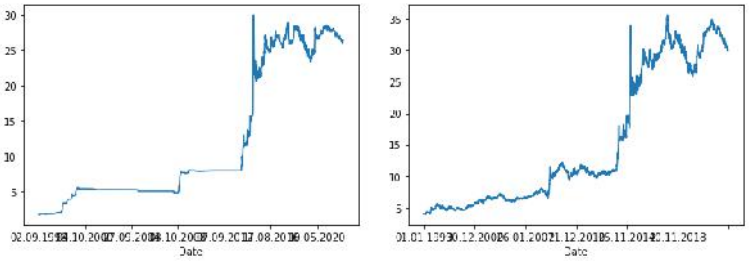

Рисунок 2.1 – Обмінні курси долара США та Євро з 2 вересня 1996 року по 20 листопада 2021 року

Як можна побачити на рисунку 2.1, у один і той самий період у курсах сталося достатньо багато різних коливань.

Зазначимо, що результати прогнозування для більш стабільної вибірки даних будуть стабільніші не залежно від використаної архітектури чи архітектур нейронних мереж. Тож, необхідно сформувати такий підхід, який буде достатньо стабільним для різних наборів даних, з різним рівнем шуму.

Отже, знаючи основні особливості вхідного набору, сформуймо вимоги до методу.

2.3 Архітектури нейронних мереж

Спираючись на попередні дослідження, сформовані вимоги та інформацію що ми маємо про доступні валютні курси виберемо і опишемо базову архітектуру нейронних мереж, що будуть використані

2.3.1 Прогнозування часових рядів за допомогою нейронних мереж

На разі існують декілька методів прогнозування часових рядів. Для подальшого формування методології необхідно розглянути вже існуючі методи.

Класичним методом прогнозування часових рядів є модель ARIMA. Ця модель є комбінацію підходів авто регресії та ковзаючого середнього. Це достатньо проста модель, що здібна враховувати параметри сезонності, довгострокові тенденції, авто регресії та ковзного середнього, щоб обробляти автокореляцію, що включена у дані.

У методі експоненціального згладжування прогнози робляться на основі середньозважених величин, як і у моделі ARIMA, але у цьому випадку кожному спостереженню призначаються різні ваги, що зменшуються і спостереженням, що далі від сьогодення надається менше значення. [9]

Основними недоліками та обмеженнями таких традиційних моделей є зменшення точності, у випадку, якщо деякі данні втрачені, неможливість відстежити складні закономірності в даних та відсутність можливості здійснити достатньо точне довгострокове прогнозування. [9]

Уникнути даних недоліків здібні методи глибокого навчання, а саме повторювані нейронні мережі та їх варіації.

Рекурентні нейронні мережі, так само к і звичайні ШНМ, складаються з шарів що поділяються на вхідний, прихований та вихідний шари. Так сам,, як і у звичайних нейронних мережах кожне з’єднання має вагу яку необхідно тренувати.

Різниця у тому яким чином повторювана нейронна мережа обробляє данні. Основною її особливістю є те, що вона, як показано на рисунку 1.1 повторює один і той самий крок для кожного елементу вхідного набору, але поведінка кожного разу залежить від попередніх наданих даних. Тож, вихід прихованого шару одного часового кроку є входом наступного шару.

Основні переваги рекурентних нейронних мереж складаються у тому, що відсутні значення істотно не впливають на продуктивність нейронної мережі, вони можуть знаходити складні закономірності у вхідних часових рядах, вони дають прийнятні результати за декілька кроків, і вони здатні моделювати послідовність даних, зв’язуючи кожну вибірку з попередньою. [9]

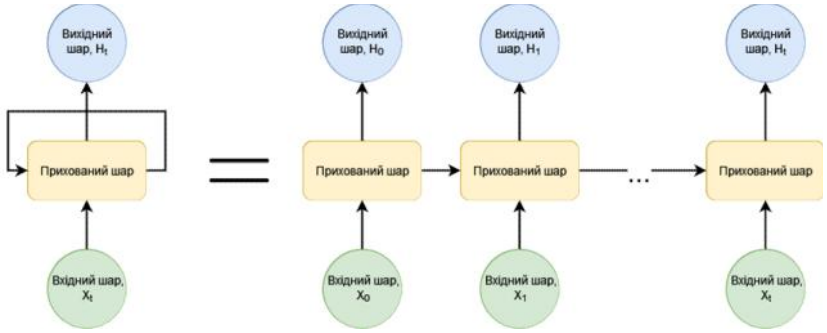

Рисунок 1.1 – Структура рекурентної нейронної мережі

І хоча, рекурентні нейронні мережі виправляють більшість недоліків традиційних нейронних мереж у розрізі прогнозування часових рядів, вони все ж мають свої недоліки. Підчас навчання на великих вибірках даних. RNN зазвичай страждають від проблеми зникаючого градієнта або градієнта, що вибухає, що означає, що параметри в прихованих шарах або не змінюються так сильно, або вони призводять до числової нестабільності та хаотичної поведінки. Внутрішні рекурентні мережі, описані вище, страждають від слабкої пам’яті, не здатної врахувати декілька елементів минулого при передбаченні майбутнього. Також, навчання рекурентної нейронної мережі важко паралелізувати, а обчислення займають велику кількість часу.[9]

Для виправлення цих недоліків з’явилися мережі довгої

короткострокової пам’яті та мережі закритого повторного блоку.

У мережах довгої короткострокової пам’яті або LSTM, для вирішення проблеми зникаючого градієнта прихований шар було замінено блоком LSTM, структура якого наведена на рисунку 1.2. LSTM складається з стану комірки, що переносить інформацію по всій послідовності і представляє собою пам’ять мережі, ворота забуття, що вирішують, що важливо утримати від попередніх кроків часу, вхідний шлюз, який вирішує, яку інформацію необхідно додати з поточного кроку часу та вихідний вентиль, який визначає значення виходу на поточному кроці часу. [9]

Стан клітинки передає відповідну інформацію під час навчання, так що інформація з попередніх кроків часу надходить на кожному часовому етапі, зменшуючи вплив короткочасної пам’яті. Під час навчання на всіх кроках часу, ворота дізнаються, яку інформацію важливо зберегти або забути, і додають її до стану клітинки або видаляють з неї.

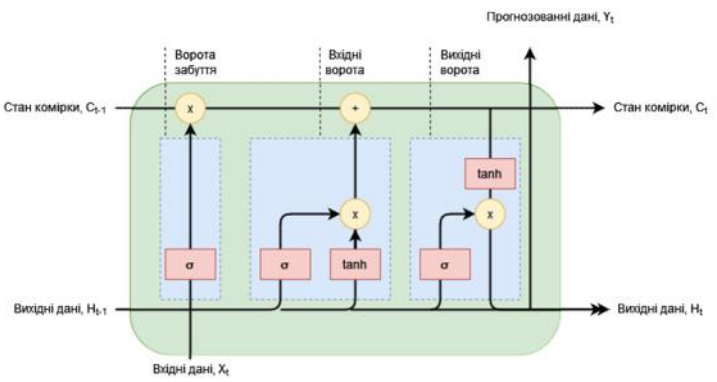

Рисунок 1.2 – Структура мережі довгої короткострокової пам’яті

Таким чином LSTM дозволяє відновити дані, передані в пам’яті, вирішуючи проблему зникаючого градієнта. LSTM корисні для класифікації, обробки та прогнозування часових рядів з часовими лагами невідомої тривалості.

У мережах закритого повторного блоку або GRU для вирішення проблеми зникаючого градієнту прихований шар було замінено блоком GRU, структура якого наведена на рисунку 1.3.

У цій моделі, на відміну від LSTM усього два шлюзи – шлюз забуття та шлюз оновлення. Шлюз скидання вирішує скільки інформації з попередніх кроків часу можна забути. Шлюзу оновлення, вирішує, скільки інформації з попередніх кроків часу необхідно зберегти, пам’яті, яка переносить інформацію по всій послідовності і представляє пам’ять мережі. Це дозволяє не втрачати данні та вирішує проблему плаваючого градієнта

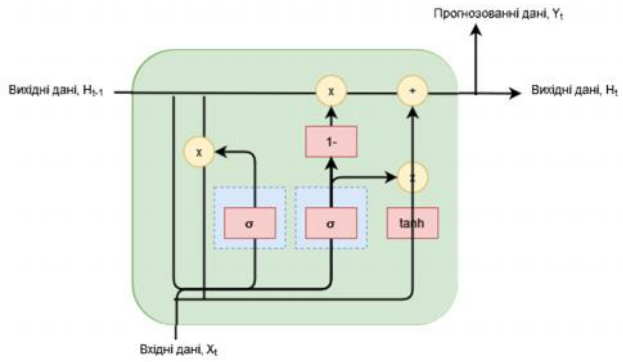

Рисунок 1.3 – Структура мережі закритого повторного блоку

2.3.1 Ансамблі нейронних мереж

У загальних рисах, ансамбль можна вважати наразі поширеною технікою навчання, де дека кількість моделей об’єднуються для вирішення проблеми. Це робиться тому, що ансамбль, як правило показує кращу продуктивність ніж одиночні нейронні мережі, через те, що покращує здатність до узагальнення.[15]

Ансамблі складаються з декількох, попередньо навчених, нейронних мереж.

Надалі, розглянемо найпоширеніші техніки формування ансамблів нейронних мереж. Можна зазначити, що таких технік на практиці три: конкатенація, усереднений ансамбль та зважений ансамбль.

Ансамбль конкатенації отримує різні вхідні дані, незалежно від їх розмірів, і об’єднує їх на заданій осі. Недоліком приєднання інформації поруч може бути вибух розмірності. Більше мереж в ансамблі або найбільше

— це щільні розмірності, і більший вихід конкатенації. Ця операція може бути дисперсійною, не дозволяючи кінцевій частині мережі дізнатися важливу інформацію або призводити до переобладнання.

Ансамбль усереднення можна розглядати як протилежність операції конкатенації. Об’єднані виходи наших мереж пропускаються через щільні шари з фіксованою кількістю нейронів, щоб вирівняти їх. Таким чином ми можемо обчислити середнє значення. Результатом є вихід, який має однакову розмірність кожного входу, але це середнє з них. Як інтуїтивно зрозуміло, недоліком такого підходу може бути втрата інформації, викликана характером середньої операції.

Зважений ансамбль — це особлива форма середньої операції, де вихідні дані тензора множаться на вагу, а потім лінійно об’єднуються. Кожному входу, що надходить від наших моделей, призначається вага. Ці ваги визначають внесок кожної окремої моделі в кінцевий результат. Особливість полягає в тому, що ці ваги не є фіксованими та попередньо встановленими, але їх значення можна оптимізувати під час навчального процесу. Єдиним обмеженням є те, що суми ваги повинні бути до 1, але цього легко досягти, застосувавши до них softmax. За допомогою цього типу ансамблю ми також можемо отримати внесок кожної моделі, просто витягуючи ваги ансамблю.

У даній роботі будуть сформовані і оцінені усі з наведених видів ансамблів.

2.4 Алгоритми роботи

Першим кроком після аналізу даних та вибору архітектури є формування алгоритму навчання, оцінки, аналізу та роботи фінального механізму прогнозування.

На рисунку 2.5 представлений алгоритм навчання нейронної мережі. Деякі з блоків, такі як попередня обробка даних та оцінка точності моделі

приховують у собі додаткові алгоритми. Їх буде розглянуто пізніше. Важливо також зазначити, що єдиним блоком, що потребує втручання є блок оптимізації моделі. Як би прикро не було, але модель сама себе написати та змінити не зможе. Цей блок необхідний для того, щоб підібрати оптимальні змінні для навчання – від кількості епох, до функцій оптимізації та втрат.

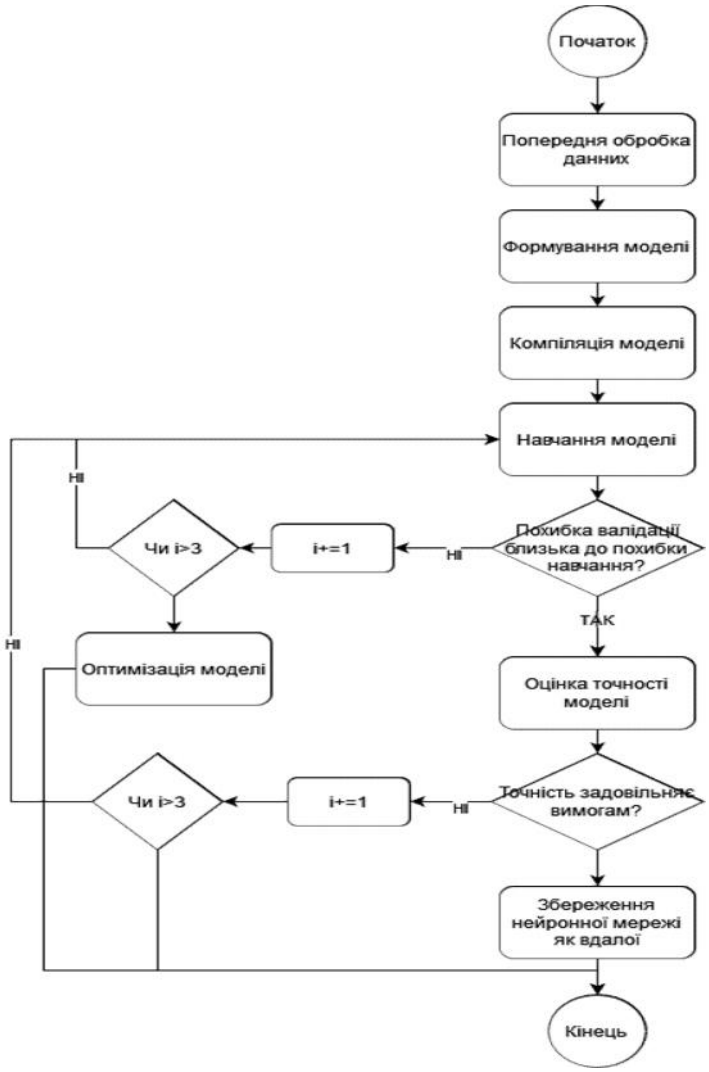

Рисунок 2.5 – Алгоритм навчання нейронної мережі

Як вже було зазначенно у данному алгоритмі є блоки, що представляють собою додаткові алгоритми. Для попереднього підготування данних використовуеться алгоритм, який отримуе данні з вже існуючого файлу, якщо файл відсутній, алгоритм отримує данні з офіційного сайт Національного банку України. Потім данні шкалюються, розподіляються на

три наоби – навчальний, валідаційний та тестовий, потім наобори кешуються, змішіються та пакетуються. Отсання маніпуляція з данними робится для уникнення перенавчання нейронної мережі, але навчання кожнохї нової нейроннлї мережі буде відрізнятися від попередньої. Тож, успішні варіанти тотрібно зберігати одразу після навчання. З цієї причини, мережі надається шість стпрб для показання оптимальних результатів. Увесь цей процесс є процессом підготовки данних для навчання нейронно мережі.

Алгоритм оцінки точності представляє собою два цикли. Перший цикл генерує необхідну кількість досліджень точності, другий – проводить дослідження. У процесі дослідження точності, кожен вхідний елемент набору данних віддається нейронній мережу для прогнозування, потім, прогнозованне значення проівнюється зі значенням фактичним, ураховуючи усі зроблені припущення і робиться запис про вдали чи не вдале прогнозування. Після завершення циклу, оцінюється кількість вдалих прогнозів. Після завершення першого циклу, на підставі усих отриманних данних про точність, вираховується середня точність моделі.

Алгоритм створення ансамблю достатньо простий, але вадлливо, щоб на попередньому кроці, кроці навчання усі необхідні нейронні мережі мали достатню точніть. Також, для створення ансамблю необхідно точно знати кількість нейронних мереж, це необхідно для того, щоб коректно завантажити попередньо збереженні данні. Формування ансамблю також завершується оцінкою його точності.

Механізм прогнозування представляе собою компіляцію цих алгортмів. Після оцінки продуктивності та відбору нейронних мереж для ансамблю нейронних мреже, та після оцінки точності ансамблів, модна говорити про узагальнюючий результат роботи – механізм прогнозування.

Для прогнозування цьому механізму необхідно два вхідних парметри – валюта для прогнозування, та бажана дата. Зауважимо, що данна программа націлена на прогнозування валютного курсу на наступний день, тож очікується, що дата буде не пізніше ніж завтра. Програма послідовно

отримае, шкалює, і доцільно зауважити що для шкалювання будуть використані ті самі данні що й для навчального та тестового наборів, тож, прогрмама передбачае деяку роботу з початковими даннми, завантажить ансамбль нейронних мереж, відповідний заданійвалюті та спрогнозує значення на задану дату. Вивід програми – числове значення валютного курсу у гривні, на наступний день.

2.4.1 Недоліки і переваги наданного алгоритму

Як і будьякий метод, створенний до цього, даний має свох певін недоліки і переваги. Розглянемо їх. Перш за все, основною вадою цього метода є необхідність зовнішнього втручання. Данний метод передбачає аналіз продуктивності та зіставлення ансамблю нейронної мережі мануальними методами. Тож, для кожної нової валюти необхідно буде пройти один і той самий шлях від навчання до створення ансамблю.

До переваг створенного методу відноситься адаптивність – не зважаючи на те, що для кожної валюти потрібно пройти достатньо довгий шлях, ми можемо бути впевнені в тому, що данна метода спрацює навть для валюти якій притаманні великі коливання, що можна буде побачити у резільтаті емпіричного дослідження. Також, важливою перевагою цього методу є можливість маштабування – у перспективі, створенні мережі можуть бути розширені для довгостроковго прогнозування. Ще однією перевагою цього підходу є можливість додавання багатофакторного прогнозування.

3 ПРОГРМАНА РЕАЛІЗАЦІЯ ОКРЕМИХ НЕРОЙННИХ МЕРЖ ТА ЇХ АНСАМБЛІВ

3.1 Особливості підготування даних для навчання нейронної мережі

Спершу, поглянемо на початковий на набір даних. На рисунку 3.1 наведено фрагмент навчальної вибірки у тому вигляді, у якому Національний банк України зберігає данні про валютний курс. Важливо зазначити, що данні до 2020 року зберігались не за одну одиницю валюти, а за сто одиниць, тож, при створенні вибірки, усі данні було приведено до ціни за одну одиницю.

Національний банк України
встановлює такі офіційні курси гривні до іноземної валюти - **Долар США**
Період з 02.11.2018 по 01.01.2020

Дата	Час	Кількість одиниць	Офіційний курс
02.11.2018	00.00	100	2815.7863
17.12.2019	00.00	100	2349.0376
18.12.2019	00.00	100	2346.9132
19.12.2019	00.00	100	2341.3086
20.12.2019	00.00	100	2337.4052
21.12.2019	00.00	100	2332.5284
22.12.2019	00.00	100	2332.5284
23.12.2019	00.00	100	2329.1225
24.12.2019	00.00	100	2327.5766
25.12.2019	00.00	100	2327.5766
26.12.2019	00.00	100	2325.5155
27.12.2019	00.00	100	2329.2885
28.12.2019	00.00	1	23.6862
29.12.2019	00.00	1	23.6862
30.12.2019	00.00	1	23.6862
31.12.2019	00.00	1	23.6862
01.01.2020	00.00	1	23.6862

Рисунок 3.1 – Данні про валютний курс за Національним банком України

Надалі, усі дані потрібно шкалювати – або нормалізувати, або стандартизувати. Нормалізація передбачає зміну діапазонів даних без зміни форми розподілу, а стандартизація змінює форму розподілу даних, данні приводяться до нормального розподілу. Це потрібно для підвищення точності навчання. Звісно, можна навчити нейронну мережу на необроблених

даних, і є вірогідність, що вона буде мати точність вищу за випадкове відгадування, що дорівнює 50% вірних відповідей, але, як видно на рисунках 3.2.б та 3.2.в попередня обробка даних зменшує похибку, і як результат, нейронна мережа має вищу точність після навчання.

Лістинг 3.1 – Стандартизація даних

```
trainMean = data[:trainSplit].mean()
trainStd = data[:trainSplit].std()
data = (data-trainMean)/trainStd
```

Лістинг 3.2 – Нормалізація даних

```
trainMin = data[:trainSplit].min()
trainMax = data[:trainSplit].max()
data = (data-trainMin)/(trainMax - trainMin)
```

Важливо зазначити, що при нормалізації даних похибка при навчанні менша, ніж при стандартизації даних, і обидві вони менші за похибку при навчанні на не оброблених даних. І хоча, як можна побачити на рисунку 3.2.а похибки навчання та валідації все ж наближуються одна до одної, не рекомендується використовувати не обробленні данні. Отже, для навчання нейронних мереж було використано нормалізацію данні. Надалі буде оцінено точність нейронних мереж, що були навчені на по різному підготовлених наборах даних.

Зауважимо, що і для нормалізації і для стандартизації необхідно використовувати значення отримані з навчальної вибірки.

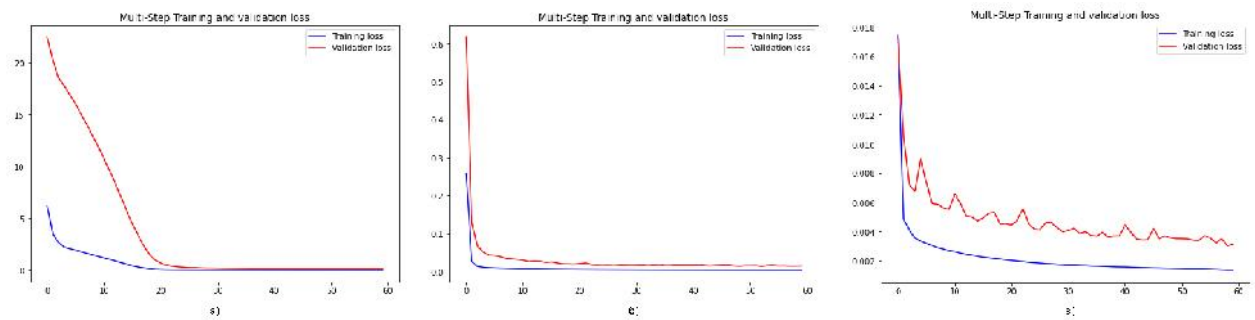

Рисунок 3.2 – Графіки втрат на навчанні та валідації з: а) без попередньої обробки данних; б)з стандартизаціею данних; в) з нормалізаціею данних.

Після проведення шалювання необхідно сформувати вхідні та очікувані вектори. Зазначимо також, що для тестування нейронних мереж доцільно використовувати тестову вибірку, і очевидно, до цієї вибірки повинні входити данні з якими мережа не знайома. На рисунку 3.3 можна побачити принцип формування цих векторів. Для їх формування ми визначаємо розмір вхідного вектору, та розмір очікуваного. За навчальну множину ми беремо 80% усіх даних, 20% порівну ділимо на валідаційну та тестову множини.

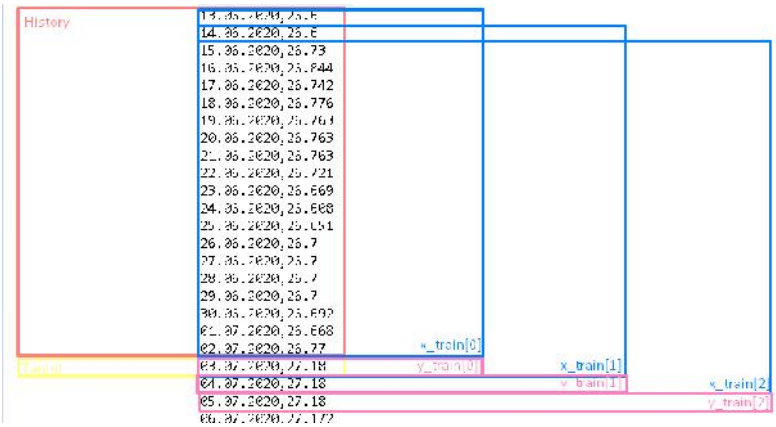

Рисунок 3.3 – Принцип формування вхідного та цільового векторів

У прикладі 3.3 можна побачити функцію за допомогою якої формуються цільові та вхідні вектори. Навчальні та тестові множини мають однаковий формат, тож ми можемо використовувати дану функцію для формування усіх необхідних множин даних.

Лістинг 3.3 – Функція формування вхідних векторів та цільового вектору

```
def univariateData(data, startIndex, endIndex, historySize,
targetSize, number):
  xSet = []
  ySet = []
  startIndex = startIndex + historySize +number
  if endIndex is None:
    endIndex = len(dataset) - targetSize
  for i in range(startIndex, endIndex):
    indices = range(i-historySize, i)
    xSet.append(np.reshape(data[indices], (historySize, 1)))
    ySet.append(data[i+targetSize])
  return np.array(xSet), np.array(ySet)
```

У прикладі 3.3 наведено функцію, у прикладі 3.4 – її використання. Функції передається вже нормалізована множина, початковий, та кінцевий індекси, цільовий розмір вхідного та вихідного векторів, і якщо потрібно, зміщення. Формування тестової вибірки проводитися так само як і начальної та валідаційної. Вона не буде приймати участі у навчанні мережі, але буде використана для визначення точності прогнозування.

Лістинг 3.4 – Фомування вхідних та цільових векторів

```
xTrain, yTrain = univariateData(data, 0, trainSplit,
                                pastHistory,
                                futureTarget,
                                number)
```

На рисунку 3.3 наведено приклад вхідного та цільового вектору. Кожен з цих векторів це NumPy масив. Зазвичай, такого формату даних достатньо для навчання нейронної мережі, адже на далі над ними буде проведено ще кілька дій. Для того, щоб уникнути перенавчання нейронної мережі – тобто ситуації у якій вона просто запам’ятає дані, необхідно змішати, пакетувати та кешувати данні.

```
Вхідний вектор
[[0.85557522][0.8560708][0.85550442][0.85415929][0.85178761][0.85253097][0.8539823][0.85451327][0.85507965]
[0.85292035][0.85299115][0.85465487][0.8540177][0.85249558][0.85044248][0.84860177][0.84831858][0.84715044]
[0.84729204][0.84499115]]
Цільовий вектор
0.8413823008819557
```

Рисунок 3.3 – Приклад вхідного та цільового векторів

Увесь процес обробки наборів даних приведений у прикладі 3.4. Кешування даних використовуються для того, щоб під час першого повторення набору даних його елементи будуть кешуватися або у вказаному файлі, або в пам’яті. Подальші ітерації використовуватимуть кешовані дані. Перемішування набору даних використовується для того, щоб використовувати випадкові послідовності, що дозволить уникнути перенавчання. Функція пакетування перетворює вхідний та цільовий вектор

на пакет.

Лістинг 3.5 – Змішування, пакетування та кєшування наборів

```
  trainData= tf.data.Dataset.from_tensor_slices((xTrain,
yTrain))
  trainData=trainData.cache()
               .shuffle(BUFFER_SIZE)
               .batch(BATCH_SIZE).repeat()
```

3.2 Реалізація моделі нейронної мережі довгострокової короткої пам’яті

Маючи підготовані набори даних, можна починати створення моделі нейронної мережі . Бібліотека Keras надає два способи створення нейронних мереж. Перший та другий варіанти представленні у прикладах 3.6 та 3.7. У прикладі 3.6 наведено створення моделі LSTM, у прикладі 3.7 – GRU. Моделі складаються з шарів, що ми один за одним додаємо до них. Першим шаром моделей є шар LSTM або GRU, це одразу і вхідний шар, через що, у його описі зазначаймо форму вхідних даних, та основний логічний шар наших мереж. Через те, що прогнозуймо ми лише на наступну добу, нам необхідний всього один шар. У цього шару лише 8 вхідних нейронів, що, як буде видно надалі, достатньо для короткострокового прогнозування. Вихідний, останній шар, складеться для обох моделей складатися лише з одного вихідного нейрону.

Лістинг 3.6 – Формування моделі LSTM

```
lstmModel = tf.keras.models.Sequential([
    tf.keras.layers.LSTM(8, input_shape=xTrain.shape[-2:]),
    tf.keras.layers.LeakyReLU(),
    tf.keras.layers.Dense(1)
])

lstmModel.compile(optimizer=’Nadam’, loss=’mae’)
```

Бібліотека Keras надає змогу додати функцію активації і як параметр шару, і як окремий шар. У обох моделях вже присутня стандартна функція

активації tahc, що описано вище. Для оптимізації моделі ми замінними її на іншу, таку з якою модель у дослідах показує вищу точність і меншу різницю у похибках навчання та валідації. LeakyReLU або випрямлений лінійний блок з витоком є такою функцією активації. Звичайний ReLU на інтервалі $x < 0$ дає на виході нуль, тоді як LeakyReLU має на цьому інтервалі невелике негативне значення, кутовий коефіцієнт близько 0,01. Тобто функція для LeakyReLU має вигляд $f(x) = \alpha x$ при $x < 0$ та $f(x) = x$ при $x \geq 0$, де α – мала константа. У нашому випадку, ця функція активації показує достатньо стабільні результати. Основною різницею між цими способами створення нейронної мережі є способи додавання шарі.

Також, у прикладах 3.6 та 3.7 можна побачити рядок компіляції, саме у ньому зазначається одна з найважливіших для навчання нейронної мережі речей – оптимізатор. Річ у тім, що похибки при навчанні нейронної мережі змінюються залежно від оптимізатора, як і якість навчання нейронної мережі.

Лістинг 3.7 – Формування моделі GRU

```
model = tf.keras.models.Sequential()
model.add(tf.keras.layers.GRU(8,

input_shape=x_train_uni.shape[-2:]))
model.add(tf.keras.layers.LeakyReLU())
model.add(tf.keras.layers.Dense(1))

model.compile(optimizer='Nadam',loss='mae')
```

Дуже часто, на навчанні нейронної мережі не зосереджують достатньої уваги, розглядаючи вибірки даних, довжину навчання і архітектуру моделі більше ніж саме навчання. А між тим, під поняттям «навчання нейронної мережі» ховайтеся обробка цільової функції оптимізатором. Як можна побачити на рисунку 3.5, при зміні оптимізатора, змінюється й похибка, що сигналізує і про зміни процесі навчання. І якщо при використанні оптимізатора «Adagrad» похибка валідації і навчання відрізняються на стільки сильно, хоча і мають тенденцію до зменшення у певний момент, що про будь яку точність такої мережі не має й мови, то при використанні

«RMSprop» або «Nadam» похибки навчання та валідації достатньо близькі одна до одної. Розглянемо детальніше ці три оптимізатори для того, щоб зрозуміти причину такої поведінки. Важливо також зазначити, що оптимізатор Nadam, показує стабільно прийнятні результати як для LSTM мережі, так і для GRU.

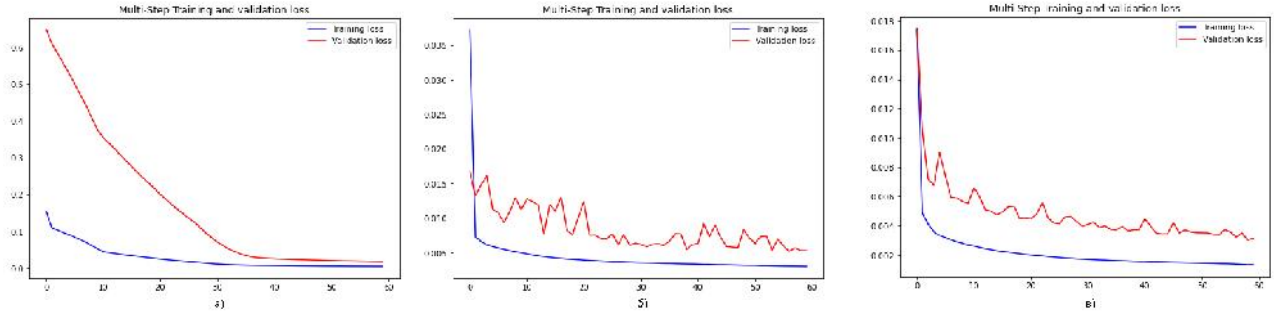

Рисунок 3.5 – Криві втрат при навчанні з різними оптимізаторами для двух нейронних мереж: а) Adagrad; б) RMSprop; в) Nadam

Адаптивний градієнт або Adagrad. Звичайний стохастичний градієнтній спуск та його інерційні варіації не враховують той момент, що деякі ознаки можуть бути вкрай інформативними, але в той же час рідко зустрічатися, наприклад, у незбалансованих вибірках. Хоча варто уточнити, що йдеться не лише про вхідні параметри – такі ж рідкісні ознаки можуть зустрітися і у глибинних уявленнях згорткові мережі, коли вхідні параметри були пропущені кількома шарами. Ідея підходу з адаптивним градієнтом полягає в наступному: необхідно для кожної ознаки підбирати свою швидкість навчання так, щоб мінімізувати ступінь оновлення найпоширеніших ознак і збільшити її для тих, що рідко зустрічаються.

Головний недолік алгоритму полягає в тому, що при його роботі відбувається постійне накопичення квадратів градієнтів у знаменнику, через що кожен новий член, що додається, позитивний. Це, у свою чергу, призводить до того, що коефіцієнт навчання для деяких ознак стає настільки малим, що алгоритм не здатний продовжувати додаткові дослідження поверхні цільової функції. Як вирішення проблеми були запропоновані

AdaDelta, RMSProp та Adam.

Середньоквадратичне поширення або RMSprop це алгоритм оптимізації, розроблений паралельно з AdaDelta і його складовою. Обидва алгоритми були створені для вирішення основної проблеми AdaGrad: безконтрольного накопичення квадратів градієнтів, що зрештою призводило до паралічу процесу навчання. Ідея RMSProp така: замість повної суми оновлень Gt використовуватиметься усереднений з історії квадрат градієнта. Метод нагадує принцип, що використовується в MomentumSGD - метод експоненційно загасаючого середнього, що біжить.

Оцінка адаптивного моменту або Adam це алгоритм оптимізації, що поєднує принципи інерції MomentumSGD та адаптивного оновлення параметрів AdaGrad та його модифікацій. Подібно до того, як Adam по суті є RMSprop з імпульсом, Nadam є Adam з імпульсом Нестерова.

Отже, за результатами практичного дослідження, для поставленої задачі підходять оптимізатори RMSprop, Adam та Nadam. Для навчання остаточних нейронних мереж було обрано оптимізатор Nadam – з ним похибки на навчанні та валідації все ж, як можна побачити на графіках рисунку 3.5 , найменше, що корелює з точністю прогнозування у майбутньому.

3.2.1 Навчання нейронної мережі та оцінка точності прогнозування

Отже, навчимо декілька мереж, та збережемо їх для подальшого аналізу. Код, що використовується для навчання нейронної мережі наведений у прикладі 3.8. Для навчання ми будемо використовувати вже підготовані данні, для кожної нової нейронної мережі, що буде навчатися на наборі, ми будемо додавати зсув, для того, щоб трохи змінити данні. Важливо зазначити, що попередньо данні було перемішано таким чином, щоб вони не повторювались, тож і без додаткового зсуву, вірогідність того, що мережі будуть навчатися на абсолютно ідентичних наборах, у абсолютно ідентичній

послідовності достатньо мала. Також, зазначимо, що оптимальним форматом для рекурентних нейронних мереж є пікетовані данні.

Лістинг 3.8 – Навчання нейронної мережі

```
model.fit(trainData, epochs=EPOCHS,
                      steps_per_epoch=EVALUATION_INTERVAL,
                      validation_data=valData,
validation_steps=50)
```

Емпірично визначено, що достатньою кількістю епох для даних нейронних мереж є 60. Епоха у навчанні нейронної мережі – це проходження усього набору даних у прямому та зворотному напрямку через нейронну мережу усього один раз. Детальніше цей процес описано у другому розділі даної роботи. Достатня кількість кроків у епоху, для даних нейронних мереж, 500 кроків.

Для навчання будемо використовувати два набори даних – курс долара США, та курс євро. Нам доступні данні від 2 вересня 1996 року і по наш день. Кількість даних не перевищує семи тисяч одиниць. Може здаватися, що це достатньо мала кількість даних, але нейронні мережі довгої короткострокової пам’яті здібні навчатися достатньо добре і на невеликих кількостях даних.

Лістинг 3.9 – Розрахунок точності прогнозування валютного курсу нейронною мережею.

```
 for x, y in val_univariate.take(len(x_val_uni)):
     prediction = keras_model.predict(x)[0][0]
     real = round(tf.keras.backend.get_value(y[0]),7)
     predictionVal = float(round((prediction *
trainStd)+trainMean,4))
     realVal = float(round(((real * trainStd)+ trainMean,4))
     infelicity = float(predictionVal - realVal)
     if(infelicity > -0.10):
         if(infelicity < 0.10):
             predicted.append(True)
         else:
             predicted.append(False)
     else:
         predicted.append(False)
```

Для кожного набору даних ми навчимо три нейронні мережі. І оцінимо їх точність. При оцінці точності будемо вважати відхилення у 10 копійок мінімально допустимим, оскільки різниця між обмінним курсом за Національним банком та у обмінниках може складати від 10 до 50 копійок.

У прикладі 3.9 наведено функцію за допомогою якої буде оцінено точність нейронних мереж. Було зроблено припущення, що до допустимої похибки, і результати прогнозування ми будемо оцінювати, спираючись на це припущення Також, для того, щоб відстежити тенденцію поведінки проведемо декілька циклів тестування. Результати тестування представленні у таблиці 3.1.

На цьому кроці важливо визначити основні тенденції у коливанні точності прогнозування для кожної нейронної мережі. Також, важливо оцінити точність нейронних мереж і обрати найкращі для подальшого використання у ансамблях.

Згідно з таблицею 3.1, більшість мереж можуть прогнозувати достатньо точний курс більш ніж у 70% випадків. Але, ця точність не є сталою. Як можна побачити з таблиці 3.1 у деяких випадках точність прогнозування може коливатися від 2,5 % до 7,3%. З метою зменшити такий розрив та підвищити точність об’єднаймо кращі мережі у ансамблі. При формуванні деяких ансамблів такі коливання можуть викликати нестабільність результату самого ансамблю. Згідно до зробленого раніше припущення, різні моделі можуть показати на одних й тих самих наборах даних різну продуктивність, і як бачимо з таблиці 3.1, на наборі даних про курс долара США більш успішна модель LSTM, у той час як мережі навчені на курсі Євро не перевищують в середньому поріг точності у 65%.

Отже, спираючись на результати перевірки точності прогнозування можемо вважати, що ми отримали необхідні за вимогами мережі з точністю більше 80% для даних з меншою кількістю коливань, та з точністю прогнозування вище 60% для даних з великою кількістю коливань. Надалі, для підвищення точності прогнозування будуть розроблені ансамблі.

Таблиця 3.1 – Оцінка точності прогнозування валютного курсу окремими нейронними мережами

ШНМ	Валюта	Точність прогнозування, %										Середня точність, %	Коливання точності, %
		1	2	3	4	5	6	7	8	9	10		
LSTM	Долар США	82,9	84,4	80,3	82,2	82,5	82,2	84,7	84,9	84,0	82,9	83,1	4,6
		82,5	85,5	85,8	87,3	83,2	84,7	83,3	82,5	86,9	83,0	84,47	4,8
		83,2	84,1	84,4	85,1	82,9	85,2	83,3	82,7	86,8	85,8	84,35	4,1
	Євро	64,3	64,1	64,7	62,1	63,1	66,2	63,3	65,5	66,2	63,1	64,26	4,1
		52,1	51,6	49,8	50,7	50,2	49,5	48,8	47,9	52,8	50,9	50,43	4,9
		70,2	66,9	68,1	64,3	65,9	64,7	69,0	64,0	65,0	67,8	66,59	6,2
GRU	Долар США	62,4	62,3	59,3	61,8	62,4	60,2	60,8	60,5	59,0	62,7	61,14	3,7
		75,6	80,7	78,0	74,8	77,0	81,4	75,3	75,3	74,1	79,6	77,18	7,3
		80,0	81,1	81,0	80,0	81,3	80,3	82,1	80,0	81,1	79,6	80,65	2,5
	Євро	64,7	64,5	66,0	64,5	64,5	66,7	63,4	67,6	68,6	65,0	65,55	5,2
		61,6	64,5	65,7	65,0	67,1	68,4	62,6	65,5	64,1	65,2	64,97	6,8
		49,7	48,4	49,8	50,3	47,6	51,7	50,7	50,7	50,5	50,0	49,94	4,1

3.4 Формування та тестування ансамблю нейронних мереж

Звернемося знов до таблиці 3.1. Як можна побачити не усі нейронні мережі показують стабільно високе значення точності, для його підвищення і стабілізації об’єднаймо найбільш успішні мережі, ігноруючи мережі середня точність яких близька до 50%, у вже описані вище види ансамблів – ансамбль конкатенації, ансамбль усереднення і зважений ансамбль.

У прикладі 3.10 наведено приклад створення усередненого ансамблю за допомогою інструментів бібліотеки Keras.

Лістинг 3.10 – Формування усередненого ансамблю

```
models = [model_0,model_1,model_2]
model_input = tf.keras.Input(shape=x_train_uni.shape[-2:])
model_outputs = [model(model_input) for model in models]
ensemble_output = tf.keras.layers.Average()(model_outputs)
ensemble_model = tf.keras.Model(inputs=model_input,
                                outputs=ensemble_output)
```

Також, ансамбль можливо реалізувати за допомогою допоміжний функцій як наведено у прикладі 3.11. У цьому прикладі показане створення зваженого ансамблю.

Лістинг 3.11 – Формування зваженого ансамблю

```
    results = []
    i=0
    for model in models:
        predictionData = model.predict(x)[0][0] * wegthts[i]
        results.append(predictionData)
            i+=1
    result = sum(results)
```

Зазначимо також, що ансамбль формується з попередньо навчених нейронних мереж, їх можна зберегти і завантажити за допомогою стандартних засобів Keras. Також, до ансамблю можливо додати одну й ту саму модель декілька разів.

Отже, для наступної перевірки, для кожного набору даних було

сформовано три ансамблі з найуспішніших навчених моделей. Результати тестування представленні у таблиці 3.2

Таблиця 3.2 - Оцінка точності прогнозування валютного курсу ансамблями нейронних мереж

Вид ансамблю	Валюта	Точність прогнозування, %										Середня точність, %	Коливання
		1	2	3	4	5	6	7	8	9	10		
Конкатиняфція	Долар США	81,8	83,6	83,1	83,5	83,5	83,8	83,6	83,3	83,5	83,7	83,3	2
Середній		84,3	85	85,3	85,1	85	84,6	84,7	84,7	85,1	85	84,9	1
Ваговий		85.7	84.9	86.3	86.0	84.9	85.4	87.3	85.7	85.2	87.9	85,9	3
Конкатиняція	Євро	64	65,4	65,2	65,5	65,9	66,1	66,3	65,9	66	65,7	65,6	2,3
Середній		68,8	68,9	67,5	67,8	67,4	67,6	67,7	67,7	67,9	67,9	67,9	1,5
Ваговий		67,9	67,6	69,7	66,0	66,6	69,8	67,1	64,0	65,7	64,0	66,8	5,8

Як можна побачити з даної таблиці, ансамблі трохи зменшили коливання у точності прогнозування, до того ж усереднений та зважений ансамблі показують тенденцію до збільшення точності прогнозування – якщо, для мережі навченій на даних про долар США середня точність однієї мережі не більша за 84,35% то точність усередненого ансамблю – 84,9%, а зваженого 85,9, що дає нам підстави вважати, що ансамблі здатні збільшити точність прогнозування. Таж сама тенденція відстежується і для другого набору даних. Тож даний метод прогнозування можна використовувати для різних валют.

3.5 Програма прогнозування курсу валют

Отже, коли точність ансамблів перевірена, сформуємо програму, що засновуючись на ансамблі буде прогнозувати курс валют на завтра. Алгоритм роботи цієї програми було описано у другому розділі. Повний лістинг програми знаходиться у додатку Б.

Для формування програми прогнозування було використано усереднений ансамбль – за результатами дослідження продуктивності, що наведені у таблиці 3.2, вони показали в цілому покращення точності для нейронних мереж навчених на обох наборах даних, це стало основною причиною використання цього ансамблю як основного у прогнозуванні.

Як вже було зазначено, вхідними даними для даної програми є назва валюти, у нашому випадку «dollar» або «euro» та дата, на яку необхідно здійснити прогноз. Вихідними значеннями програми є значення курсу на попередні 20 днів, для заданої дати і прогноз. Зробимо декілька прогнозів для двох валют і порівняймо їх з реальними значеннями.

```
Please enter currency: dollar
Please enter date to predict in format D.MM.YYYY: 23.11.2021
--------------------------------------------------
Previous values:
[26.29  26.243 26.19  26.19  26.19  26.102 26.081 26.093 26.058 26.102
 26.102 26.102 26.111 26.337 26.413 26.515 26.443 26.443 26.443 26.569]
--------------------------------------------------
Prediction for 23.11.2021
26.5534
--------------------------------------------------
```

Рисунок 3.6 – Приклад прогнозування для курсу долару США на 22.11.2021

Для прогнозу було обрано дату 23 листопада 2021 року. Реальний курс на данну дату для долара США складав 26,6588 гривень, за Національним банком, прогнозованний курс, як можна побачити на рисунку 3.6 - 26,5534 гривень. Між цми значеннями різниця у 0,4% проценти, або приблизно 10 копыйок.

Зробимо прогнозування для курсу євро на ту ж саму дату. За прогнозом

курс євро 23 листопада 2021 року повинен складати, як можна побачити на рисунку 3.7 - 30,0115 гривень, фактично курс євро на цей день складав 30,0591 гривню. Різниця між прогнозованним курсом та реальним складае 4 копійки, або приблизно 0,15%.

```
Please enter currency: euro
Please enter date to predict in format D.MM.YYYY: 23.11.2021
--------------------------------------------------
Previous values:
[30.49  30.365 30.285 30.285 30.285 30.097 30.203 30.207 30.119 29.914
 29.914 29.914 29.901 30.147 30.027 29.997 29.969 29.969 29.969 29.98 ]
--------------------------------------------------
Prediction for 23.11.2021
30.0115
--------------------------------------------------
```

Рисунок 3.7 – Приклад прогнозування для курсу євро на 22.11.2021

Виходячи з цих данних, а також оцінки точності нейронних мереж та їх ансамблів можна вважати що метод прогнозування показа достатньо точні результати. Обидва анасамблі оказали достатнью точність на реальних данних, враховуючи усі припущеня і реальні коивання курсу від одного банку, до іншого у рамках одного дня, ця точність є допустимою.

ВИСНОВКИ

Об’єктом цього дослідження, як вже було визначено, є прогнозування валютного курсу. А саме, прогнозування валютного курсу за допомогою нейронних мереж. Як вже було зазначено, дана тема широко досліджувалася і раніше, і у даній роботі було прийнято рішення сконцентрувати увагу на комбінуванні різних мереж одного виду у ансамблі з метою підвищення точності короткострокового прогнозування обмінних курсів.

Метою роботи є розробка оптимального методу прогнозування валютного курсую за допомогою нейронних мереж та підвищення точності прогнозування за допомогою об’єднання нейронних мереж у ансамблі нейронних мереж.

У данній роботі було розглянуто основні особливості предметної облості і обрано такі набори данних, які відображують основні тенденції валютного ринку. Висуното основні вимоги до нейронних мереж та ансамблівСтворенно і навченно моделі штучних нейронних мереж на основі LSTM та GRU архітектур. На їх основі було створено ансамблі нейронних мереж. Нейронні мережі було оптимізованно. Було оцінено показники точності як окремих нейронних мереж.

За результатами дослідження, нейронні мережі та їх ансамблі цілком відповідають висунутим вимогам – точність нейрнних мереж для данних з високим рівнем шуму перевищке 60% , а ансамбль нейроннх мереж підвизив точність прогнозування для данних з великою кількістю коливань в середньому на 3%, точніть прогноззування для данних з меншою кількістю коливань складае 85% точності для ансамблю нейронних мереж. Тож, висунуті вимоги і гіпотези покрито і підтвердженно – комбінація декількох рекурентних мереж різної структури у ансамбль підвищує точність прогнозування, а самі рекурентні нейронні мережі показують високу точність прогонозування на достатньо нестабільних наборах данних.

У подвльшому, данну методологію можливо і бажанно розширюіати у двох напрямках. По-перше, до моделей, зараз простих можна ввести багатофакторне прогнозування, тобто при подальшому дослідженні визначити такі фактори формування валютних курсів, що найбільш корелють з їх змінами і внести їх у навчання нейронної мережі. По-друге, можна збільшувати інтервал прогнозування валютного курсу, удосконалюючи нейронні мерещі, що входять у склад ансамбілв.

Данан работа може бути використана у якості короткострокового передбачення курсу валют при невеликих валютних операціях, а також, для відстежування основних тенденцій і спостереження за факторами, що впливають на «сплкски» у валютному курсі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Міщенко В.І. Банківські операції: Підручник /В.І. Міщенко, Н.Г. Слав ’янська, О.Г. Коренєва. — 2-е вид., переробл. і доп. — К. : Знання, 2007. — С. 177 — 205.

2. Соболев В. Математическое моделирование. Задачи прогнозирования валютного курса [Електронний ресурс] / Валерий Соболев – Режим доступу до ресурсу:https://fortrader.org/fortrader_archive/59-fortrader_archive/matematicheskoe-modelirovanie-zadachi-prognozirovaniya-valyutnogo-kursa.html.

3. Chandrasekara V. Forecasting Exchange Rates using Artificial Neural Networks [Електронний ресурс] / V. Chandrasekara, C. Tilakaratne – Режим доступу до ресурсу: https://www.researchgate.net/publication/259772696_Forecasting_Exchange_Rates_using_Artificial_Neural_Networks.

4. Осовский С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского. – М.: Финансы и статистика, 2002. – 344 с.: ил

5. Хайкін С. Нейронні мережі. Повний курс. Друге видання. - М .: Вільямс, 2006. - 1104 с.

6. Harris Drucker, Yann Le Cun. Improving Generalization Performance Using Back-propagation / Harris Drucker, Yann Le Cun // IEEE Transactions on Neural Net-works. - 1992. - Vol.3, No5 - p.991-997.

7. International Finance Tutorial [Електронний ресурс] – Режим доступу до ресурсу: https://www.tutorialspoint.com/international_finance/exchange_rate_forecasts.ht

8. Документация PuzzleLib. Optimizers [Електронний ресурс] – Режим доступу до ресурсу: https://docs.puzzlelib.org/ru/documentation/base/optimizers/Optimizers/.

9. Del Pra M. Time Series Forecasting with Deep Learning and Attention

Mechanism [Електронний ресурс] / Marco Del Pra – Режим доступу до ресурсу: https://towardsdatascience.com/time-series-forecasting-with-deep-learning-and-attention-mechanism-2d001fc871fc.

10. Ломакин Н.И., Максимова О.Н., Экова В.А., Гаврилова О.А., Вагина В.Е. НЕЙРОННЫЕ СЕТИ ДЛЯ ПРОГНОЗИРОВАНИЯ КУРСА ДОЛЛАРА С ИСПОЛЬЗОВАНИЕМ АСТРОЛОГИЧЕСКИХ ЦИКЛИЧЕСКИХ ИНДЕКСОВ ГЮШОН И ГАНЮ // Международный журнал прикладных и фундаментальных исследований. – 2016. – № 6-1. – С. 133-136;

11. Фёдорова Е. ПРОГНОЗИРОВАНИЕ КУРСА ВАЛЮТЫ С ИСПОЛЬЗОВАНИЕМ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ [Електронний ресурс] / Е.А. Фёдорова. – 2013. – Режим доступу до ресурсу: https://auditfin.com/fin/2013/3/2013_III_03_11.pdf.

12. Буй Ч. ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ГИБРИДНЫХ МОДЕЛЯХ ДЛЯ ПРОГНОЗИРОВАНИЯ КУРСА ВАЛЮТЫ [Електронний ресурс] / Ч. Буй, Ф. Пащенко, А. Пащенко – Режим доступу до ресурсу: https://mlsd2020.ipu.ru/proceedings/1607-1609.pdf.

13. Фёдорова Е. Прогнозирование курса валют с помощью нейронніх сетей [Електронний ресурс] / Е. Фёдорова, М. Линкова. – 2013. – Режим доступу до ресурсу: https://cyberleninka.ru/article/n/prognozirovanie-kursa-valyuty-s-pomoschyu-neyronnyh-setey/viewer.

14. Pacelli V. An Artificial Neural Network Model to Forecast Exchange Rates [Електронний ресурс] / V. Pacelli, V. Bevilacqua, M. Azzollini. – 2010. – Режим доступу до ресурсу: https://www.scirp.org/html/4583.html.

15. Wut Hmone Hnin Hlaing. Exchange Rate Forecasting using Ensemble Artificial Neural Network Model [Електронний ресурс] / Wut Hmone Hnin Hlaing. – 2019. – Режим доступу до ресурсу: https://www.researchgate.net/publication/337007492_Exchange_Rate_Forecasting_using_Ensemble_Artificial_Neural_Network_Model.