

ГРИНЬКІВ Андрій Миколайович

**Математичне та програмне забезпечення для
прогнозування слабкоформалізованих подій із
неперервним навчанням / Mathematical Tools and
Software for Weakly Formalized Event Prediction
with Continuous Learning**

спеціальність: 121 - Інженерія програмного забезпечення
освітньо-професійна програма - Інженерія програмного забезпечення

Кваліфікаційна робота

Виконав студент групи ІПЗм-21
А. М. Гриньків

Науковий керівник:
к.е.н., доцент, Л. І. Гончар

Кваліфікаційну роботу
допущено до захисту:

" ___ " _____ 20__ р.

Завідувач кафедри
_____ **А. В. Пукас**

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ПРОЦЕСУ ПРОГНОЗУВАННЯ	
СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ	11
1.1. Цілі та умови прогнозування трудноформалізованих подій.....	11
1.2. Аналіз відомих методів та засобів прогнозування слабкоформалізованих подій	17
1.3. Постановка задачі дослідження	25
Висновки до першого розділу	26
РОЗДІЛ 2 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ДЛЯ	
ПРОГНОЗУВАННЯ СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ ІЗ	
НЕПЕРЕРВНИМ НАВЧАННЯМ	
2.1. Загальна модель неймережевого прогнозування слабкоформалізованих подій з неперервним навчанням.....	27
2.2. Метод неймережевого прогнозування з безперервним навчанням системи та з часовими зрушеннями сигналів.....	32
Висновки до другого розділу.....	38
РОЗДІЛ 3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ДЛЯ	
ПРОГНОЗУВАННЯ СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ ІЗ	
НЕПЕРЕРВНИМ НАВЧАННЯМ	
3.1. Загальні вимоги до програмної архітектури системи.....	39
3.2. Програмна архітектура системи.....	41
3.3. Програмна реалізація системи.....	49
3.4. Експериментальні дослідження з неймережевого прогнозування на прикладі прогнозування подій, пов'язаних з дорожнім рухом.....	54
Висновки до третього розділу	64

ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	67
ДОДАТОК А ЛІСТИНГ ОСНОВНИХ МОДУЛІВ СИСТЕМИ	Помилка!

Закладку не визначено.

ВСТУП

Актуальність теми. Прогнозування подій – актуальне науково-практичне завдання, що становить інтерес у багатьох галузях. Підвищення складності аналізованих процесів та властивим їм подій вимагають дедалі досконаліших інструментів прогнозування. Завдання з управління транспортом, економікою, фінансами, соціальною сферою, складними технічними об'єктами неможливо ефективно вирішувати без отримання точної та своєчасної інформації про найближче майбутнє.

Споживачами прогнозів можуть бути державні органи, комунальні та промислові підприємства, засоби масової інформації, веб-сервіси, автономні технічні засоби, класичні та мобільні програми, окремі люди. Незалежно від цілей і завдань прогнозування, воно виконується в умовах невизначеності ситуації, коли на параметр, впливають прямі і непрямі фактори, що змінюються в часі. Інформація про ці фактори часто не може бути представлена у вигляді набору простих аналітичних моделей: вона закодована у великому масиві даних, а її вилучення та обробка потребують нових, нестандартних підходів.

Зв'язок роботи з науковими програмами, планами, темами

Напрямок виконаних досліджень безпосередньо пов'язаний з науково-дослідним напрямком кафедри “комп'ютерних наук” Західноукраїнського національного університету.

Мета і задачі дослідження

Метою магістерського дослідження є розробка методу та архітектури програмної системи нейромережевого прогнозування слабкоформалізованих подій з безперервним навчанням, що підвищують, що точність прогнозування.

Для вирішення поставленої мети необхідно розв'язати наступні завдання:

- 1) проаналізувати процесу прогнозування слабкоформалізованих подій;
- 2) проаналізувати відомі методи та системи для прогнозування слабкоформалізованих подій;
- 3) розробити метод прогнозування слабкоформалізованих подій з безперервним навчанням та управлінням спрямованістю виклику сигналів з асоціативної пам'яті;
- 4) реалізувати архітектуру програмної системи, що реалізує метод прогнозування слабкоформалізованих подій з безперервним навчанням;
- 5) провести експериментальні дослідження використання розроблених методів та програмної системи.

Об'єкт дослідження – процеси прогнозування слабкоформалізованих подій із безперервним навчанням.

Предмет дослідження – методи та програмні засоби нейромережевого прогнозування слабкоформалізованих подій із безперервним навчанням.

Методи дослідження

В роботі використовувалися методи системного аналізу та синтезу, інтелектуальної обробки даних, сучасна теорія нейромережевого прогнозування, а також методи оптимізації програмних систем.

Наукова новизна одержаних результатів

Запропоновано метод нейромережевого прогнозування слабкоформалізованих подій з безперервним навчанням та управлінням спрямованістю виклику сигналів з асоціативної пам'яті, що дозволяє підвищити точність прогнозу і ґрунтується на рекурентних нейронних мережах.

Практичне значення одержаних результатів

Програмно реалізована архітектура та правила функціонування програмних систем нейромережевого прогнозування з безперервним навчанням.

РОЗДІЛ 1

АНАЛІЗ ПРОЦЕСУ ПРОГНОЗУВАННЯ СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ

1.1. Цілі та умови прогнозування трудноформалізованих подій

Прогнозування подій – актуальна задача, що становить інтерес у багатьох галузях [1-5]. Прогнози необхідні під час управління транспортом, економікою, фінансами, соціальною сферою, складними технічними об'єктами. Вони використовуються при прийнятті рішень на різних рівнях, починаючи від окремих людей та додатків, автоматизованих систем, підприємств, галузей, і закінчуючи органами державної влади та великими корпораціями. Горизонти прогнозів можуть становити від кількох секунд до кількох років. Залежно від цього різняться цілі прогнозування. Приклади цілей:

- підвищення горизонту прогнозування при обмеженнях на помилку;
- збільшення точності прогнозів при обмеженнях на помилку та ресурси, що витрачаються на прогнозування;
- скорочення часу прогнозування за умови забезпечення заданого рівня точності;
- скорочення ресурсів, витрачаються прогнозування, при обмеженнях на помилку;
- надання інформації для прогнозів вищого рівня та інші.

Підвищення складності аналізованих процесів та властивим їм подій вимагають дедалі досконаліших інструментів прогнозування [5–9]. Під подіями, що підлягають прогнозуванню, розуміються прив'язані часу сукупності факторів [10]. У ряді випадків закони прояву подій важко формалізуються та закодовані в об'ємному інформаційному потоці.

Розглянемо умови прогнозування слабкоформалізованих подій з прикладу прогнозування на користь автономних інтелектуальних роботів, і навіть прогнозування новинних інформаційних повідомлень. Ці умови досить повно демонструються наведеними прикладами, проте не обмежуються ними, а мають місце і в інших додатках, де потрібне прогнозування в умовах високої невизначеності.

Так, при прогнозуванні подій на користь інтелектуальних автономних роботів [11-15] закони прояву подій неявно описуються показаннями з датчиків температури, швидкості, тиску, витратою заряду батарей, відстанню до перешкод. Можуть бути дані і від зовнішнього джерела, що передаються робота через канал зв'язку. Наприклад, для прогнозування пробок на міських дорогах з метою планування свого маршруту безпілотний автомобіль може отримувати інформацію про дорожньо-транспортні пригоди, стан доріг, ремонтні роботи, режими роботи світлофорів, а також про поточну завантаженість міських магістралей.

У свою чергу, при аналізі потоків повідомлень новин також доводиться враховувати, що закони прояву подій записані у великих масивах текстових даних, і для передбачення їх майбутнього змісту необхідна глибока обробка лексичного складу.

Умови, які мають місце при прогнозуванні подій, що важко формалізуються, включають такі.

1. Горизонт прогнозів. Експлуатація сучасних робототехнічних виробів чітко виявила проблеми, пов'язані з нездатністю автономних машин пересуватися у суворих та/або непередбачуваних середовищах. Найчастіше операторне керування ними утруднено або неможливо через велику затримку надходження керуючих сигналів. Така затримка може бути обумовлена великою відстанню між роботом і пунктом управління або іншими причинами.

Прогнозування лексичного змісту текстових потоків зокрема може використовуватися для різних цілей. За отриманими даними, можуть коригуватися плани промислових і торгових підприємств, державних установ, окремих людей. На сьогоднішній день така інформація в основному використовується як допоміжна при прогнозуванні фінансово-економічних показників, проте вона може бути корисною і в інших додатках. Наприклад, на основі прогнозування та оцінки майбутнього змісту новин в інтернеті можливе попередження різних негативних соціальних явищ. Для того, щоб прогнози змісту повідомлення новин могли бути своєчасно використані, необхідно, щоб горизонт прогнозу був порівнянний з часом реакції на отримані новини та вжиття заходів.

Використання робототехнічних систем є особливо актуальним в умовах підвищеної небезпеки або шкідливості навколишнього середовища, а також в умовах, де людина не може перебувати з об'єктивних причин. У таких умовах може відбуватися часткова відмова апаратури робота, яку неможливо усунути з людською допомогою. Сенсори, відповідальні отримання інформації із довкілля, можуть виходити з ладу повністю чи частково, видаючи дані з перешкодами. Можуть мати місце пропуски даних.

В умовах зовнішніх перешкод (наприклад, у роботі військового призначення) дані можуть бути спотворені або відсутні. У свою чергу, потоки новинних повідомлень також схильні до різного роду спотворень і перепусток. Для текстових даних, що описують події новин, характерні помилки, починаючи помилками в словах і закінчуючи навмисними вкиданням недостовірних новин. Інформація про новини з'являється із затримкою, сервіси новин можуть бути недоступні з технічних причин. Можуть змінюватися стилістичні особливості текстів, що публікуються, з'являтися і зникати блоки контекстної реклами.

Існуючі методи не завжди здатні виконувати адекватну обробку інформації в розглянутих випадках, проте система прогнозування повинна

бути здатна обробляти інформацію навіть за умов зашумленості, часткової відсутності чи спотворення даних.

3. Врахування взаємозв'язків між елементами оброблюваних даних. Не менш важливою умовою функціонування системи прогнозування є необхідність урахування взаємозв'язків сукупності зовнішніх і внутрішніх факторів, які мають місце в середовищі функціонування об'єкта, в інтересах якого виконуються прогнози.

На прикладі автономного робота до зовнішніх факторів можна віднести такі. Це прохідність місцевості, наявність перешкод чи небезпечних зон, температура навколишнього середовища, присутність рухомих об'єктів та характер їхнього руху. Також можуть мати місце фактори, що періодично з'являються (наприклад, світлові спалахи, вибухи тощо). До внутрішніх чинників відносяться: технічний стан робота (наявність пошкоджень, обладнання, що відмовило), заряд батарей (обсяг палива), поточне додаткове навантаження (озброєння тощо), інші. Перелік факторів, що впливають на робота, залежить від його призначення та можливостей. Чим більше цих факторів буде враховано, тим точніше прогноз і вища ефективність робота. Сучасні методи не в змозі врахувати всі необхідні фактори, або їх облік вимагає неприйнятно великих обчислювальних витрат.

При прогнозуванні новин важливо враховувати взаємозв'язки між окремими елементами інформаційного блоку, що обробляється. У текстових даних цими елементами є окремі слова. Взаємозв'язки слів можуть нести в собі інформацію про майбутні події, тому необхідні методи, які здатні обробити та асоціативно пов'язати такі елементи.

Система прогнозування повинна бути здатна обробляти різноманітні дані, що несуть інформацію про фактори, які мають значення в даний момент, виявляти приховані внутрішні закономірності між ними та виконувати прогнози, ґрунтуючись на цій інформації.

4. Облік змін у законах прояву подій. Однією з ключових проблем створення перспективних автономних інтелектуальних машин є наділення їх

можливістю роботи в умовах присутності динамічних перешкод та захищеної місцевості. Природа об'єкта та характер зміни становища таких перешкод впливають на ступінь його передбачуваності. Наявність додаткових об'єктів та їх взаємодія з метою, що рухається, істотно ускладнюють прогноз. Найбільш складною і важкопередбачуваною категорією динамічних об'єктів для інтелектуального робота є інші роботи, що у області його видимості. Вони не тільки рухаються важкопередбачуваною траєкторією, але й націлені на взаємодію з об'єктами навколишнього світу, що робить завдання прогнозування подій подвійно нетривіальною. Проблема колективної взаємодії може бути відносно легко вирішена за наявності групи роботів загального керуючого центру. Однак у ряді випадків такого центру немає, і роботи належать до різних чи навіть ворожих груп. У таких ситуаціях розробники систем автономної навігації стикаються із суттєвими труднощами. Незважаючи на складність проблеми, її вирішення є надзвичайно важливим, оскільки майбутнє лежить саме за кооперативними робототехнічними системами.

При прогнозуванні змісту блоків новин у багатьох випадках є набір тематик, які охоплює оброблене інформаційне повідомлення [51]. Змінним у часі є як склад тем, і займана ними частка, і навіть тематична спрямованість усередині кожної з тем. Теми можуть бути неявно пов'язані один з одним, що також необхідно враховувати під час прогнозів. Наприклад, за новинами про ядерні випробування в Північній Кореї часто йдуть повідомлення про погіршення інвестиційного клімату в країнах-сусідах.

З наведених вище прикладів випливає, що система прогнозування повинна не тільки забезпечувати точні прогнози, але й одночасно з цим враховувати зміни в факторах, що впливають на майбутні події.

5. Оперативність виконання прогнозів.

Невід'ємною рисою перспективних інтелектуальних машин є оперативність ухвалення рішень. Оскільки середовище, в якому функціонує робот, є складним і динамічним, не виключені ситуації, в яких на ухвалення

рішення відведено дуже обмежений час. Події, подані в повідомленнях новин, також можуть розвиватися дуже стрімко і вимагають швидкої реакції. Це особливо помітно в новинах, пов'язаних із надзвичайними подіями та ринком цінних паперів. Однак існуючі підходи до прогнозування або обчислювально складні, або вимагають перенавчання при змінах прогнозованого середовища. В обох випадках між виникненням нового фактора і прийняттям рішення проходить значний час, що робить існуючі методи непридатними до прогнозування подій, що важко формуються. Система прогнозування має бути досить оперативною, щоб надавати актуальні прогнози.

б. Обмеженість апаратних ресурсів до виконання прогнозування.

Апаратні та програмні ресурси, що виділяються на прогнозування, не завжди дозволяють своєчасно отримувати цікаві для прогнозів. Крім того факту, що бортовий комп'ютер робота завантажений багатьма завданнями, не пов'язаними з плануванням поведінки, його можливості можуть бути обмежені умовами функціонування. Наприклад, для роверів-планетоходів потрібна спеціальна апаратура, стійка до космічної радіації. З цієї причини бортові комп'ютери навіть сучасних планетоходів у десятки разів слабші за звичайні побутові обчислювачі.

При прогнозуванні подій новин необхідний облік широкого контексту. Взаємопов'язаність подій із різних сфер життя призводить до того, що для точного передбачення потрібно обробляти одночасно велику кількість текстових блоків із різних тематичних сфер. При цьому загальний обсяг такої інформації експонентно збільшується. Потрібно як облік самих слів, а й зв'язків з-поміж них, що у своє чергу також багаторазово ускладнює обробку. У разі застосування відомих методів обробки природної мови на апаратних потужностях стандартних комп'ютерів становить серйозні труднощі.

З цього випливає, що система прогнозування подій має бути не лише достатньо інтелектуальною та адаптивною, а й підходити для системи з обмеженими ресурсами.

Неодмінною умовою прогнозування подій у багатьох випадках є висока шкода у разі недостатньої точності та своєчасності прогнозів. Зокрема, робототехнічні системи відрізняються високою вартістю, а усунути проблеми, що виникли через неадекватність прогнозів і, відповідно, обраного маршруту, на місці неможливо. Ушкодження обладнання, не кажучи вже про аварії, високою ціною за некоректне планування дій. У свою чергу, за некоректним прогнозуванням новинних подій можуть послідувати невірні управлінські рішення, що ведуть до великих фінансових втрат або різноманітних надзвичайних ситуацій. У зв'язку з цим від системи прогнозування подій потрібна як оперативність, а й висока точність.

Вищевикладені умови прогнозування подій тією чи іншою мірою відносяться і до інших областей прогнозування подій, що важко формуються. Вимоги до величини горизонтів прогнозів, робота в умовах спотворень та пропусків у даних, помилок програмного забезпечення та алгоритмів попередньої обробки, зовнішніх та внутрішніх факторів, що впливають на об'єкти прогнозування, роблять отримання інформації про майбутні події нетривіальним завданням. У багатьох випадках необхідно забезпечити швидкість прийняття рішень та безперервність навчання системи прогнозування за дуже обмежених апаратних можливостей використовуваної ЕОМ. На даний момент розроблено широке коло підходів до прогнозування. Проте вони багато в чому не враховують весь спектр розглянутих умов прогнозування подій.

1.2. Аналіз відомих методів та засобів прогнозування слабоформалізованих подій

Вибір методу прогнозування параметрів обумовлений специфікою завдання, особливостями середовища, в якому функціонує система

прогнозування, вимогами точності, швидкодії, а також програмно-апаратними можливостями обладнання, на якому розгорнута система. Неослабний інтерес до вивчення різних аспектів прогнозування подій знайшов свій відбиток у численних дослідженнях.

Усю сукупність методів прогнозування подій за рівнем формалізації можна поділити на інтуїтивні та формалізовані методи (рисунок 1.1). До інтуїтивних методів належать методи індивідуальних та колективних експертних оцінок. Вони застосовуються тоді, коли об'єкт прогнозування або занадто простий, або настільки складний, що аналітично врахувати вплив багатьох факторів практично неможливо.

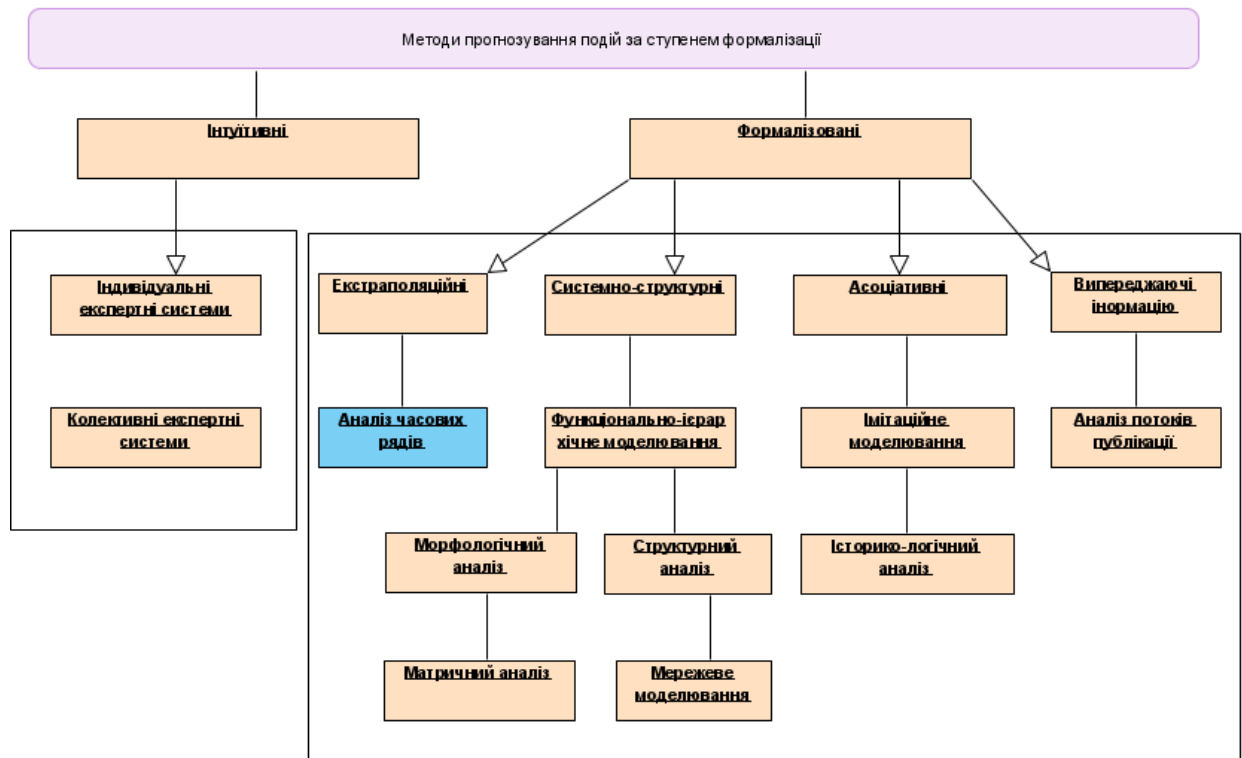


Рис. 1.1. Класифікація методів прогнозування подій за ступенем формалізації

У свою чергу формалізовані методи діляться на екстраполяційні, системно-структурні, асоціативні методи, а також методи випереджальної інформації. Системно-структурні методи включають функціонально-

ієрархічне моделювання, морфологічний аналіз, матричний аналіз, мережеве моделювання і структурний аналіз. Асоціативні методи поділяються на імітаційне моделювання та історико-логічний аналіз, а до групи методів випереджальної інформації входять такі методи, як аналіз потоків публікацій, оцінка значущості винаходів та аналіз патентної інформації [12]. Екстраполяційні методи ґрунтуються на аналізі часових рядів - упорядкованих у часі наборів вимірювань тих чи інших характеристик об'єкта, що досліджується» [13-15].

Існуючі на сьогоднішній день методи прогнозування часових рядів, поділяються на традиційні методи, методи на основі нейронних мереж та методи на основі нечіткої логіки. Структуру цих методів представлена на рисунку 1.2.

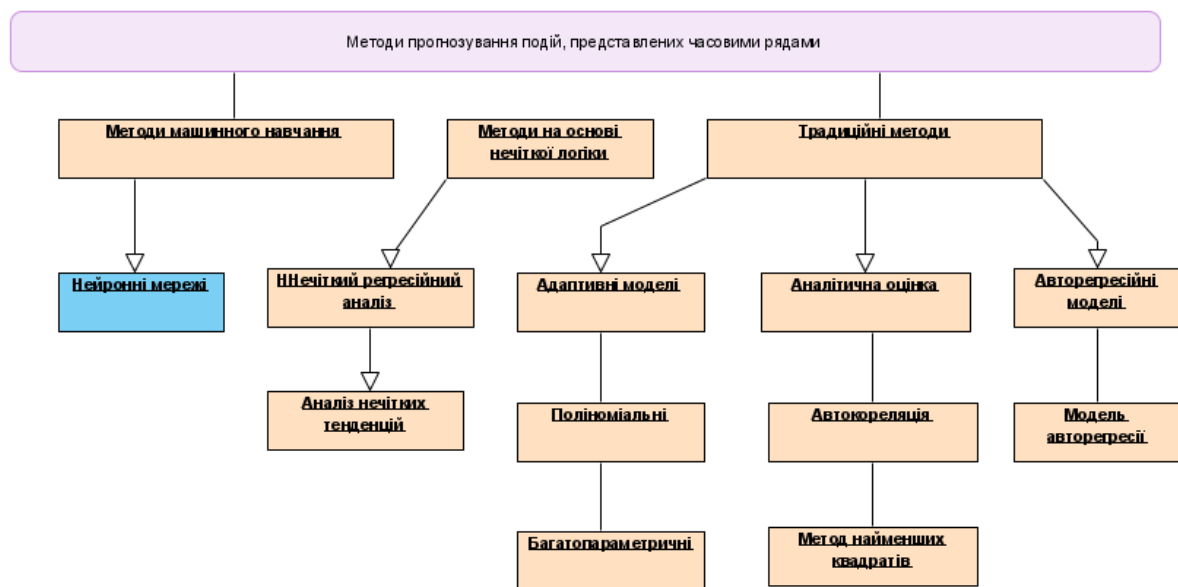


Рис. 1.2. Класифікація методів прогнозування часових рядів

Традиційні методи прогнозування часових рядів полягають у визначенні основних аналітичних параметрів часового ряду, що впливають на його динаміку, та екстраполяції на основі відомих попередніх та справжніх значень. Виділяють такі основні групи методів:

- згладжування;

- аналітична оцінка невинпадкової складової;
- авторегресійні моделі;
- адаптивні моделі;
- методи прогнозування систем часових рядів.

Згладжування часових рядів є одними з найпростіших методів. Виділяють згладжування простої ковзної середньої, зваженої ковзної середньої, експоненціальне згладжування та їх різновиду [11].

Аналітична оцінка невинпадкової складової часового ряду [13, 15] включає:

– автокореляцію рівнів, коли визначається кореляційна залежність між послідовними рівнями часового ряду. Автокореляція кількісно вимірюється за допомогою коефіцієнта кореляції між рівнями вихідного часового ряду та рівнями цього ряду, зрушеними на кілька кроків назад у часі. Коефіцієнти автокореляції характеризують тісноту лінійного зв'язку поточного та попереднього рівнів ряду. По знаку коефіцієнта автокореляції не можна дійти висновку про зростаючої чи спадної тенденції у рівнях ряду. Якщо найвищим виявився коефіцієнт автокореляції першого порядку, досліджуваний ряд містить лише тренд. Якщо найвищим виявився коефіцієнт автокореляції порядку, ряд містить коливання з періодом. Якщо всі коефіцієнти автокореляції виявилися незначними, або ряд не містить тренду і періодичних коливань, або ряд містить сильну нелінійну тенденцію;

- Прогнозування по лінії тренду. Воно здійснюється на основі кривих зростання, коли вибирається крива та оцінюються її параметри, а потім на їх основі виконується прогноз;

– прогнозування періодичної компоненти, яка може бути оцінена як різниця між значеннями, що спостерігаються, і значеннями, згладженими з використанням відповідних ковзних середніх.

Модель авторегресії – ковзного середнього та її різновиду (AR, ARMA, ARIMA та ін.):

– модель авторегресії (AR) – модель часового ряду, у якій значення лінійно залежить від попередніх значень цього ряду;

- модель авторегресії - ковзного середнього (ARMA) використовується для аналізу стаціонарних часових рядів. Вона узагальнює у собі простіші моделі авторегресії та ковзного середнього;

– модель авторегресії – інтегрованого ковзного середнього (ARIMA) – модифікація алгоритму ARMA для нестаціонарних часових рядів.

Адаптивне прогнозування передбачає, що у процесі прогнозування здійснюється облік нової інформації та коригування прогнозу. Адаптивні моделі можуть бути поліноміальними та багатопараметричними [18].

При прогнозуванні в деяких випадках доводиться враховувати взаємозв'язки різних елементів, стан кожного з яких описується окремим параметром. Сукупності значень цих параметрів, що реєструються в дискретні моменти часу, зветься систем часових рядів. Для прогнозування таких систем існують такі методи:

- трендова модель системи часових рядів - для кожного ряду, що входить в систему, визначається наявність тренду. Якщо тренд спостерігається, то відповідних часових рядів будується кілька конкуруючих парних моделей, у тому числі вибирається оптимальна. Якщо ж тренд не виявлений, то часові ряди згладжуються ковзною середньою;

– спільна гармонійна модель будується для часових рядів, у яких виявлено циклічну складову;

– модель векторної авторегресії – це модель, що одночасно описує поведінку кількох спільно залежних часових рядів через зміну власних попередніх значень та значень інших спільно залежних рядів.

Як правило, статистичні моделі можна застосовувати для отримання короткострокових прогнозів. Відповідно до цього підходу, коли змінюються закони динаміки часових рядів, потрібно будувати нові моделі, адекватні поточній ситуації. Ця особливість робить такі методи малоприматними для прогнозування подій в умовах високої динамічності і невизначеності

параметра. Ці методи не здатні реагувати на зміни зовнішньої ситуації з достатньою оперативністю. Тому на сьогодні найбільш перспективним підходом є використання евристичних моделей на основі накопиченого досвіду.

Методи нечіткої логіки є одним з підходів, що активно розвиваються зараз, до прогнозування часових рядів. Представлення часових рядів у класі нечітких часових рядів ґрунтується на припущенні, що можлива лінгвістична інтерпретація значень часового ряду, заснована на теорії нечітких множин. Така інтерпретація може виконуватися як на основі абстрактних лінгвістичних оцінок, так і із залученням експертів..

Серед існуючих підходів найбільш гнучким та інтелектуальним є підхід, що ґрунтується на машинному навчанні [21-24]. Серед них особливий потенціал мають штучні нейронні мережі. Вони знайшли широке застосування у роботизованих системах. Корисні властивості нейронних мереж: нелінійність, відображення вхідної інформації у вихідну, адаптивність, очевидність відповіді, контекстна інформація, масштабованість, відмовостійкість, однаковість аналізу та проектування, аналогія з нейробіологією. Як приклад успішної практичної реалізації системи, що виконує локалізацію та розпізнавання об'єктів на зображенні, можна навести архітектуру YOLO..

Рішення на основі нейронних мереж можна класифікувати за способом втілення: на апаратні та програмно-апаратні. Апаратні ґрунтуються на застосуванні нейронних чіпів.

Відомий проект «Spinnaker» (скорочення від Spiking Neural Network Architecture) та однойменна архітектура нейроморфного комп'ютера, що розвивається в Advanced Processor Technologies Research Group (APT). SpiNNaker спрямований на створення масово паралельного мільйонно-ядерного комп'ютера, архітектура між'єднань якого натхненна характеристиками зв'язності мозку ссавців та який підходить для моделювання великомасштабних нейронних мереж у біологічному

реальному часу. Зокрема, його архітектура дозволяє передавати дуже велику кількість маленьких пакетів даних, кожен з яких явно передає джерело та неявний час одного нейронного потенціалу дії або сплеску.

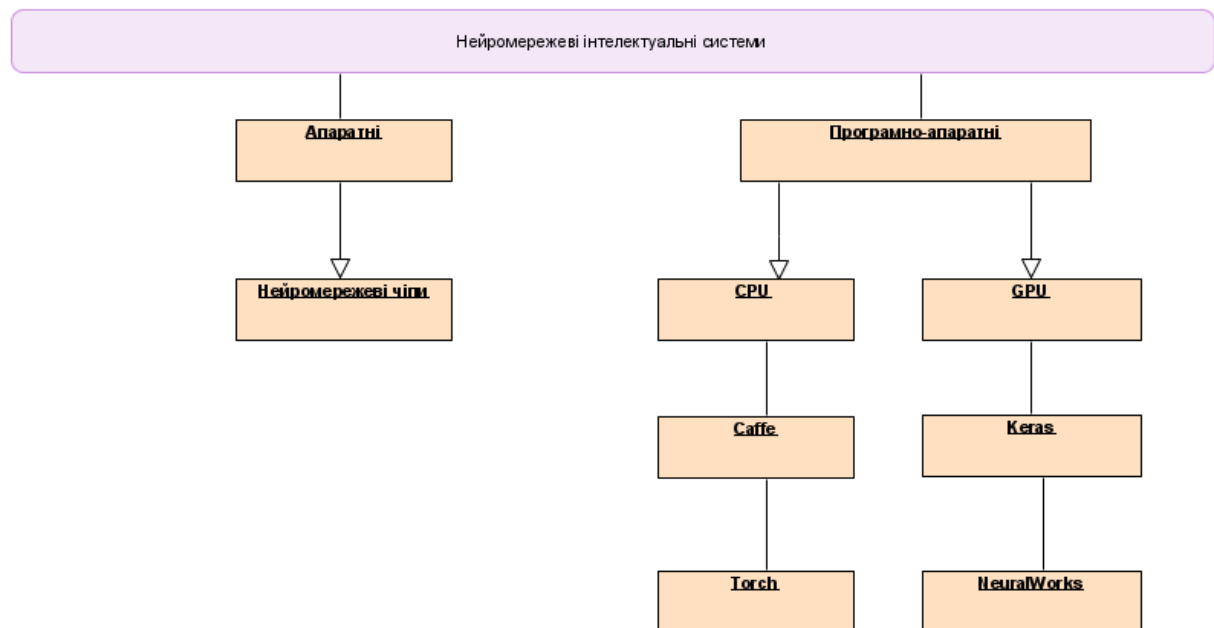


Рис. 1.3. Класифікація способів реалізації нейромережевих інтелектуальних систем

Апаратний підхід до реалізації нейромережевих архітектур в даний час не представлений масово і є скоріше об'єктом експериментів. Переважна більшість робіт пов'язана з програмно-апаратним підходом, і ці дослідження спрямовані на застосування нейромережевих методів для вирішення різних прикладних завдань.

Для типових нейромережевих архітектур розроблено ряд програмних пакетів, що містять набори функцій емуляції нейронних мереж. Оскільки нейронні мережі по суті є паралельним розподіленим процесором, їх навчання знайшли широке застосування як центральні (CPU), а й графічні процесори (GPU). Приклади нейромережевих пакетів: NeuralWorks, Caffe, Keras, Tensorflow, Torch, NeuroShell.

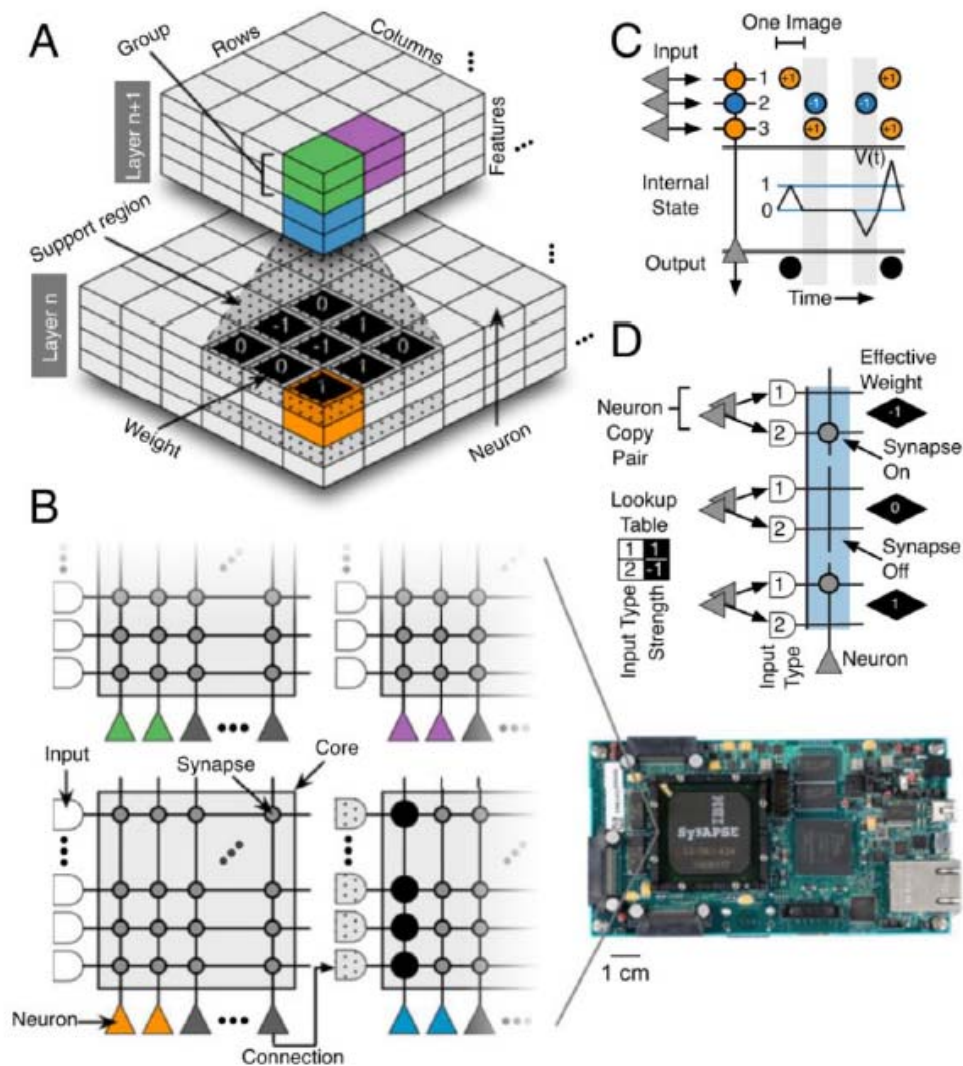


Рис. 1.4. Згортова нейронна мережа, реалізована на базі нейрочіпа TrueNorth

Рекурентні нейронні мережі, що розробляються в останні роки, з керованими елементами, які є самонавчальними, мають значний потенціал для вирішення завдання прогнозування подій. На відміну від відомих аналогів, такі мережі можуть наділитись лінійними, спіральними, петлевими та іншими логічними структурами. Вони можуть паралельно обробляти різномірні дані, при цьому за рахунок наділення шарів нейронної мережі різними конфігураціями можливе управління просторово-часовим зв'язуванням та асоціативним викликом інформації з пам'яті таких мереж. Необхідно розробити моделі, методи та архітектури програмних систем

прогнозування з безперервним навчанням, які можна буде застосовувати у складі інтелектуальних систем прогнозування часових рядів в умовах високої невизначеності ситуації та присутності великої кількості змінних факторів.

1.3. Постановка задачі дослідження

Аналіз досліджуваного процесу та існуючих методів прогнозування часових рядів дозволяє стверджувати необхідність удосконалення цих методів. Потрібний пошук нових моделей, методів та архітектур програмних систем прогнозування тслабкоформалізованих подій з безперервним навчанням. Відомі методи та підходи не повною мірою застосовні для прогнозування часових рядів, оскільки ці методи не враховують наявні можливості та умови такого прогнозування.

Метою магістерського дослідження є розробка методу та архітектури програмної системи нейромережевого прогнозування слабокформалізованих подій з безперервним навчанням, що підвищують, що точність прогнозування.

Для вирішення поставленої мети необхідно розв'язати наступні завдання:

- 1) проаналізувати процесу прогнозування слабокформалізованих подій;
- 2) проаналізувати відомі методи та системи для прогнозування слабокформалізованих подій;
- 3) розробити метод прогнозування слабокформалізованих подій з безперервним навчанням та управлінням спрямованістю виклику сигналів з асоціативної пам'яті;
- 4) реалізувати архітектуру програмної систем, що реалізує метод прогнозування слабокформалізованих подій з безперервним навчанням;

5) провести експериментальні дослідження використання розроблених методів та програмної системи.

При розробці методу нейромережевого прогнозування пропонується застосувати клас рекурентних нейронних мереж з керованими елементами, що дозволить використовувати їх можливості щодо наділення шарів різними логічними структурами та задіяти їх потенціал з управління асоціативним викликом інформації з їхньої пам'яті.

Розробка нових методів та архітектури програмної системи нейромережевого прогнозування з безперервним навчанням дозволить розширити можливості щодо прогнозування часових рядів зі складною динамікою, підвищить інтелектуальність систем, що їх використовують, і підвищить ефективність цих систем.

Висновки до першого розділу

1. Проведено аналіз відомих методів та систем прогнозування подій, за результатами якого було встановлено, що вони не задовольняють умовам, які мають місце при функціонуванні часових рядів зі складною динамікою. Показано, що існуючі підходи не враховують ряд особливостей, властивих таким часовим рядам.

2. Здійснено постановку задачі магістерського дослідження із врахуванням особливостей функціонування інтелектуалізованих систем.

РОЗДІЛ 2

МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ДЛЯ ПРОГНОЗУВАННЯ СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ ІЗ НЕПЕРЕРВНИМ НАВЧАННЯМ

2.1. Загальна модель нейронного прогнозування слабкоформалізованих подій з неперервним навчанням

З формальної точки зору прогнозування подій за часовим рядом

$$X = \{x_1, x_2, \dots, x_T\}, \quad (2.1)$$

на горизонт $T_{\text{упр}}$ зводиться до пошуку часового ряду

$$\{x_{T+1}, x_{T+2}, x_{T+\tau}, \dots, x_{T+T_{\text{упр}}}\}, \quad (2.2)$$

який описує майбутні події, де $\tau = \overline{1T_{\text{упр}}}$. Система прогнозування реалізує функцію S від вихідного ряду X , що дозволяє отримати часовий ряд X_p , який відрізняється від реального X'_p на величину менш заданої помилки E :

$$X_p = S(X), \quad (2.3)$$

$$\text{Error}(X_p, X'_p) < E, \quad (2.4)$$

Можуть прогнозуватись різні події. Наприклад, для безпілотних транспортних засобів це можуть бути ряди у вигляді значень швидкостей

руху на дорогах у фіксовані моменти часу, обсягів дорожнього трафіку чи завантаженості доріг. Ці часові ряди з часом доповнюються новими значеннями, інформація оновлюється, і прогноз, що видається, потребує коригування з урахуванням знову отриманих значень.

При цьому можуть змінюватися закони поведінки рядів, які потрібно враховувати для отримання точних прогнозів подій. Це потребує постійного навчання систем прогнозування та відсутності переривань цього навчання при отриманні прогнозів. Відомі рішення не задовольняють цим вимогам, що суттєво позначається на точності прогнозів, що формуються. Пропонується нова нейромережева модель прогнозування, що дозволяє усунути існуючу суперечність та підвищити точність прогнозування різних подій. Ця модель нейромережевого прогнозування подій з безперервним навчанням представлена на рисунку 2.1

Згідно рисунку 2.1 до складу системи прогнозування з безперервним навчанням входять:

- 1) блок попередньої обробки даних;
- 2) перший екземпляр нейронної мережі;
- 3) блок зовнішньої обробки №1;
- 4) другий екземпляр нейронної мережі;
- 5) блок зовнішньої обробки №2;
- 5) блок управління прогнозуванням.

Блок попередньої призначений перетворення вхідних даних у послідовність сукупностей одиничних образів – формат, сприйманий нейронною мережею, і навіть визначення ефективного способу кодування даних.

Перший екземпляр нейронної мережі призначений для обробки даних та безперервного навчання (тобто побудови в собі просторово-часової моделі подій, що відбуваються у зовнішньому середовищі та відображені у значеннях часового ряду) та розпізнавання цих даних.

Блок зовнішньої обробки №1 призначається для розкодування результатів розпізнавання, одержуваних з рекурентної нейронної мережі. Процедури розкодування є зворотною використовується в блоці зовнішньої обробки.

Другий екземпляр нейронної мережі, призначений для отримання з блоку управління прогнозуванням просторово-часової моделі, приведення свого стану у відповідність до цієї просторово-часової моделі шляхом присвоєння відповідним елементам отриманих значень, обробки скопійованих даних у собі з формуванням прогнозу на вказаний обрій та видачею результатів прогнозування на виході.

Блок зовнішньої обробки №2 за своєю структурою та виконуваним операціям аналогічний блоку зовнішньої обробки №1, за винятком, призначений для розкодування результатів прогнозів.

Блок управління прогнозуванням призначений для здійснення загального контролю над системою прогнозування, і зокрема над екземплярами нейронних мереж першого та другого типів.

Об'ємними стрілками позначені потоки даних, що опрацьовуються. Головна ідея, покладена в основу запропонованої моделі системи прогнозування, полягає у використанні двох екземплярів рекурентних нейронних мереж, перший з яких безперервно працює в режимі навчання, а в другому за потребою виконується прогнозування з урахуванням навченої просторово-часової моделі, скопійованої з першої мережі. Така схема дозволяє, по-перше, проводити навчання та прогнозування в реальному часі, по-друге, швидко без перенавчання виконувати оновлення прогнозів, і по-третє, не спотворювати навчену просторово-часову модель через зміну режимів роботи.

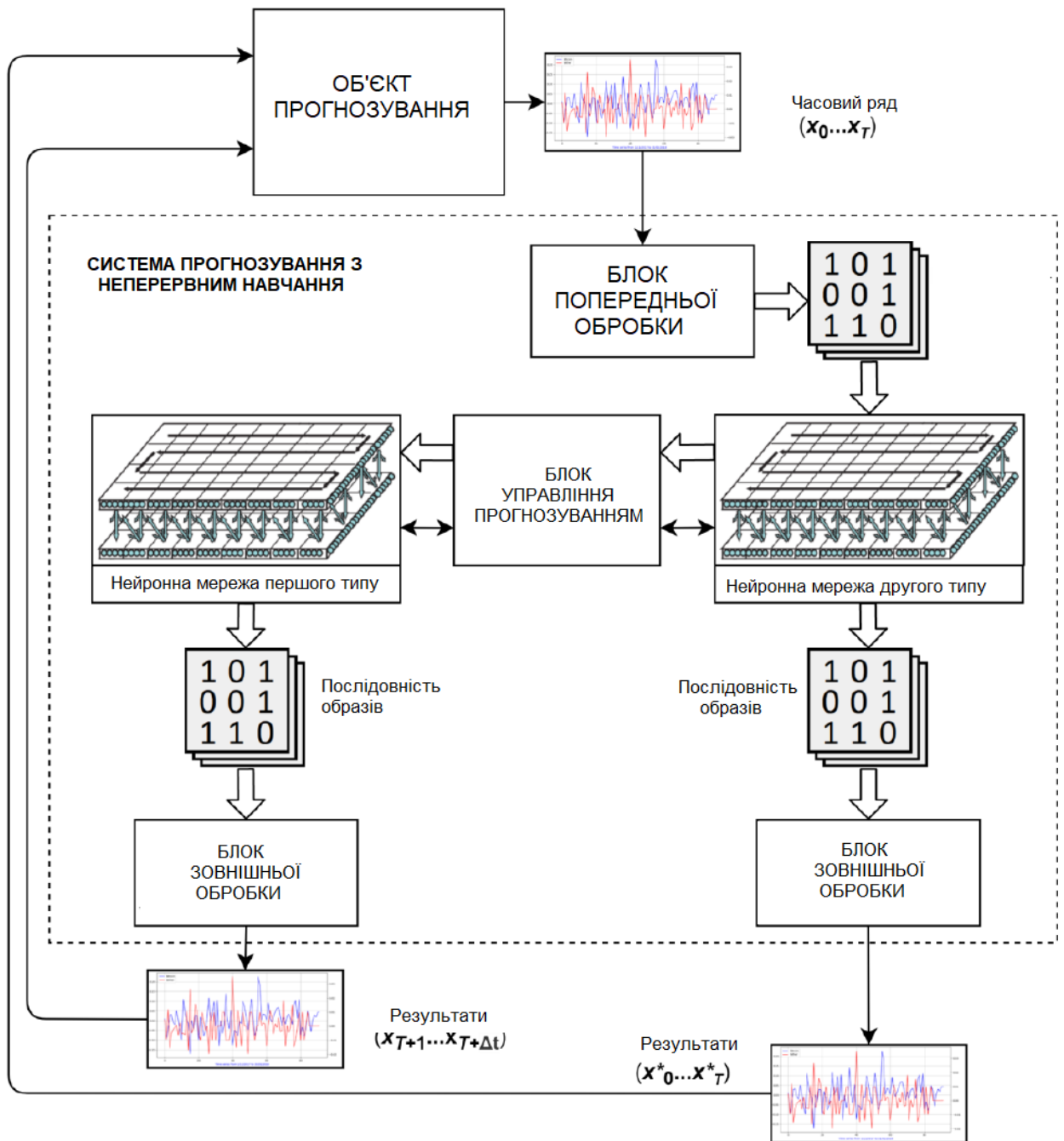


Рис. 2.1. Загальна модель системи нейромережевого прогнозування подій, що слабо формалізуються, з безперервним навчанням.

Вибір рекурентних нейронних мереж як основа системи прогнозування пояснюється тим, що такі архітектури призначені для обробки часових послідовностей довільної довжини, і це робить їх найбільш придатними для прогнозування подій в умовах високої невизначеності.

Наявність зворотних зв'язків дозволяє здійснювати глибшу просторово-часову обробку інформації та виконувати прогнози на великі горизонти, ніж звичайними нейронними мережам прямого поширення. Вибір таких мереж обґрунтовується їх широкими функціональними можливостями з керованої асоціативної обробки сигналів.

Нижче наведено послідовність кроків, що описує роботу запропонованої моделі прогнозування з безперервним навчанням.

Крок 1. Елементи часового ряду, що обробляється, на рисунку 2.1 позначеного x_0, x_1, \dots, x_T , надходять до блоку попередньої обробки. Там вони перетворюються на формат, що сприймається нейронною мережею, і надходять у рекурентну нейронну мережу.

Крок 2. Послідовність закодованих даних обробляється мережею. На елементах мережі відбувається формування просторово-часової моделі. При цьому здійснюється розпізнавання та часткове відновлення елементів вибірки.

Крок 3. Блок управління прогнозуванням виконує копіювання станів елементів рекурентної мережі першого типу в рекурентну мережу другого типу та запускає на ній обробку скопійованих даних. У мережі другого типу реалізується асоціативний виклик прогнозних значень часового ряду.

Крок 4. Здійснюється зняття інформації з виходу мережі та розкодування даних у блоці зовнішньої обробки №1. Отримання результатів розпізнавання вхідних даних нейронною мережею ($x^*_0 \dots x^*_T$). Зняття інформації з виходу та розкодування їх у блоці обробки №2. Отримання результатів прогнозування ($X_{T+1}, \dots, X_{T+\Delta t}$).

У рамках запропонованої моделі можуть реалізовуватися різні способи прогнозування подій, що слабо формалізуються, з безперервним навчанням.

2.2. Метод нейромережевого прогнозування з безперервним навчанням системи та з часовими зрушеннями сигналів

Модель нейромережевого прогнозування подій із безперервним навчанням (рисунок 2.1) може реалізовуватися, виходячи з різних завдань та умов. Її основна ідея полягає в наявності двох паралельно працюючих екземплярів нейронної мережі, перший з яких безперервно навчається, а другий за необхідністю виконує прогнози, ґрунтуючись на сформованій в першій просторово-часовій моделі процесу, що спостерігається. Виходячи з вимог до системи прогнозування з безперервним навчанням, вона може реалізовуватися різними методами.

Розглянемо метод нейромережевого прогнозування з безперервним навчанням із часовими зрушеннями сигналів. В основу цього методу покладено рекурентні нейронні мережі з керованими елементами. Такі нейронні мережі є структурами, що складаються з двох шарів нейронів, двох блоків одиничних затримок, двох блоків динамічних синапсів і блоку управління. Нейрони можуть перебувати у трьох станах: очікування, збудження та рефрактерності (несприйнятливості).

Спочатку всі нейрони перебувають у стані очікування. Коли сумарний заряд на вході нейрона перевищує деякий поріг збудження, нейрон збуджується і генерує одиночний імпульс (сигнал), після чого перетворюється на стан несприйнятливості вхідних сигналів.

У цьому стані нейрон не може накопичувати заряд та збуджуватись. Час на рефрактерність нейронів більше часу сумарної затримки одиночних імпульсів у двошаровому контурі рекурентної нейронної мережі.

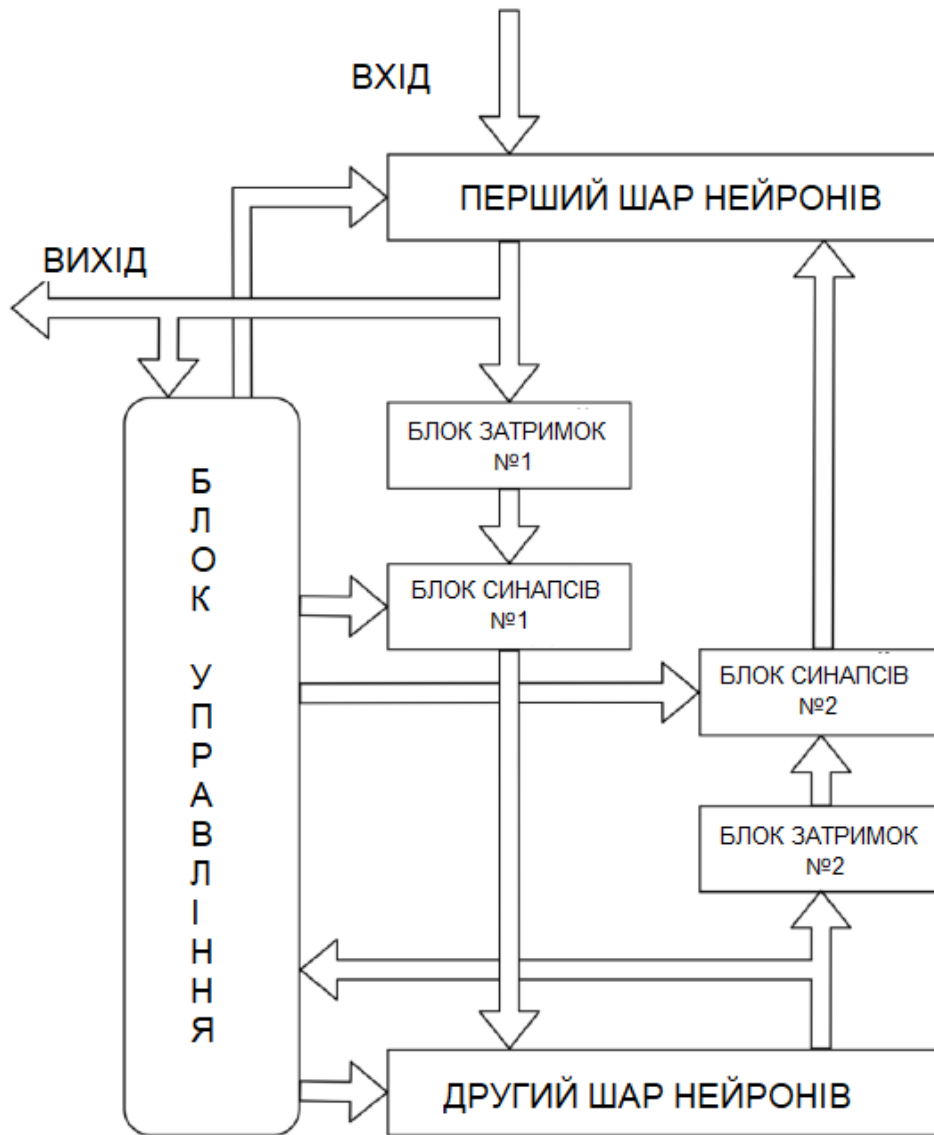


Рис. 2.2. Структурна схема рекурентної нейронної мережі з керованими елементами

Нейронна мережа оперує з даними, представленими як сукупностей одиничних імпульсів збуджених нейронів, званих сукупностями одиничних образів. Згенеровані нейронами імпульси затримуються в блоці одиничних затримок і через блок динамічних синапсів передаються нейрони іншого шару. Блок пов'язує синапс кожен нейрон одного шару з усіма нейронами іншого шару. Амплітуда одиничного образу на виході кожного синапсу, що зв'язує i -й нейрон з j -м нейроном на момент часу t , дорівнює амплітуді

вхідного одиничного образу, помноженої на вагу $w_{ij}(t)$. Ваги визначаються за формулою:

$$w_{ij}(t) = k_{ij}(t) \times \beta(r_{ij}(t)) \times \eta(r_{ij}(t)) \quad (2.5)$$

де $k_{ij}(t)$ – ваговий коефіцієнт;

$\beta(r_{ij}(t))$ - функція ослаблення розбіжних одиничних образів;

$\eta(r_{ij}(t))$ функція ослаблення схожих одиничних образів;

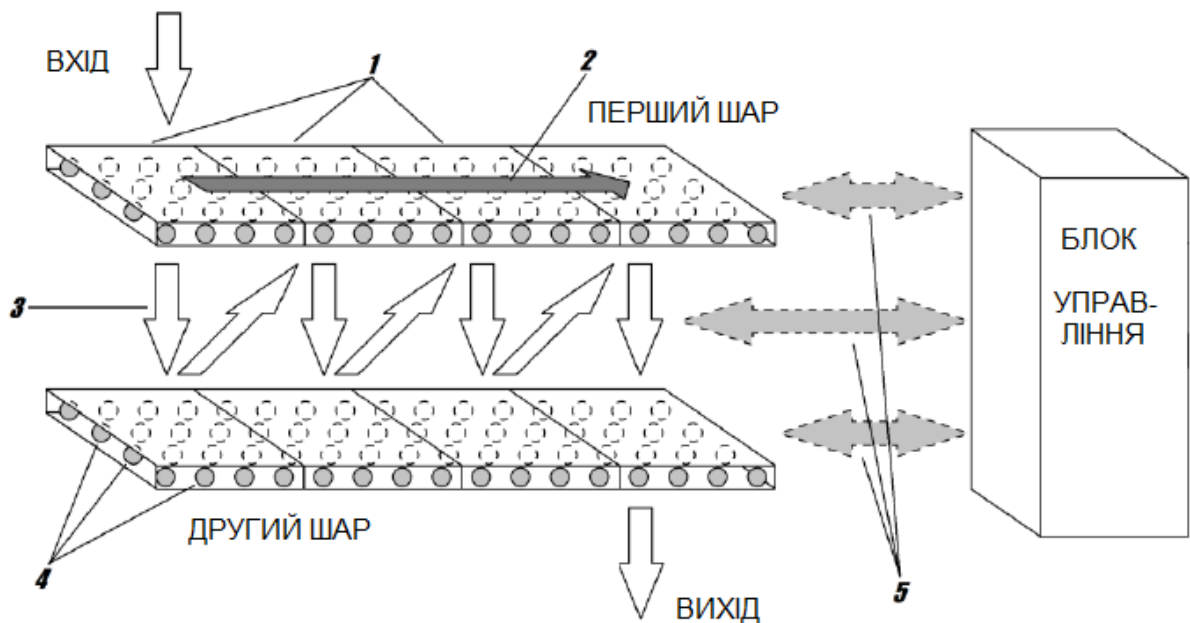


Рис. 2.3. Рекурентна нейронна мережа з лінійною структурою шарів: 1 – логічні поля, куди діляться шари мережі; 2, 3 – напрями руху образів вздовж шарів та між шарами відповідно; 4 - нейрони, 5 - керуючі впливу блоку управління

Вагові коефіцієнти $k_{ij}(t)$ змінюються в залежності від впливу на відповідні синапси імпульсів, що проходять через них, і є елементами довготривалої асоціативної пам'яті. При проходженні по синапсу імпульс знімає з нього інформацію про попередні впливи і залишає інформацію про

свій вплив за допомогою зміни вагового коефіцієнта. Залежно від станів i -го та j -го нейронів ваговий коефіцієнт змінюється за такими правилами:

$$k_{ij}(t) = th(\gamma * g_{ij}(t)) \quad (2.6)$$

$$g_{ij}(t) = g_{ij}(t - \Delta t) + \Delta g_{ij}(t) \quad (2.7)$$

$$\Delta g_{ij}(t) = \begin{cases} g+, \text{ якщо } i - \text{й нейрон згенерував сигнал} \\ g-, \text{ якщо } i - \text{й нейрон не генерував сигнал,} \\ 0, \text{ у всіх інших випадках} \end{cases} \quad (2.8)$$

де $g_{ij}(t)$, $g_{ij}(t - \Delta t)$ – передісторія навчання на моменти часу t і Δt відповідно, $g_{ij}(0) = 0$;

γ - позитивний коефіцієнт;

$g-$, $g+$ – деякі негативне та позитивне значення відповідно.

Функції $\beta(r_{ij}(t))$ ослаблення розбіжних одиничних образів залежать від умовної віддаленості $r_{ij}(t)$ зв'язуваних синапсом нейронів:

$$\beta(r_{ij}(t)) = \frac{1}{1 + \alpha^h \sqrt{r_{ij}(t)}} \quad (2.9)$$

де α , h – позитивні коефіцієнти.

При передачі одиничних імпульсів між шарами блок управління визначає стан шару, що приймає. Зауважимо, що рекурентні мережі, що використовуються, за рахунок реалізації просторових зрушень сигналів при передачі від одного шару до іншого наділяються логічними структурами. Зрушення здійснюються шляхом зміни відповідних умовних відстаней $r_{ij}(t)$. Завдяки зсувам шари нейронної мережі розбиваються на логічні поля однакового розміру. Якщо логічні поля мають розмір $d \times q$, то відповідні розміри шарів L і M визначаються з виразів:

$$L = d \times D, M = q \times B \quad (2.10)$$

де D, B - позитивні цілі числа.

За рахунок зрушень сигналів при їх передачі від шару до шару шари рекурентної нейронної мережі можуть конфігурувати різні логічні структури: лінійні, спіральні, петлеві. На рисунку 2.3 представлена нейронна мережа такого типу з лінійною структурою шарів.

Вибір рекурентної нейронної мережі з керованими елементами як модельної основи запропонованого методу пояснюється такими причинами. На відміну від відомих методів, мережі даної архітектури наділені прозорою логічною структурою, що розширює їх можливості щодо управління асоціативним викликом інформації. Їх шари можуть бути налаштовані в різні структури, при цьому дані будуть просуватися від входу до виходу різними маршрутами.

Структура рекурентної нейронної мережі з керованими елементами допускає паралельну обробку кількох нейромережевих каналів. Саме ця властивість використана у запропонованому методі нейромережевого прогнозування з тимчасовими зрушеннями.

Розглянемо докладніше запропонований метод. Його виконання описується наступним алгоритмом (рисунок 2.4).

Крок 1. Введення даних у систему прогнозування, розбиття їх на два канали: перший (поточний) та другий (затриманий на заданий час). Кодування даних у послідовність образів.

Крок 2. Асоціативна обробка даних рекурентної нейронної мережі першого типу. Формування в мережі просторово-часової моделі подій, що спостерігаються.

Крок 3. Зчитування інформації про стан рекурентної нейронної мережі першого типу до блоку управління прогнозуванням. Копіювання станів синапсів рекурентної нейронної мережі першого типу в рекурентну нейронну

мережу другого типу, а також першого (поточного) каналу рекурентної нейронної мережі першого типу до другого (затриманого) каналу.

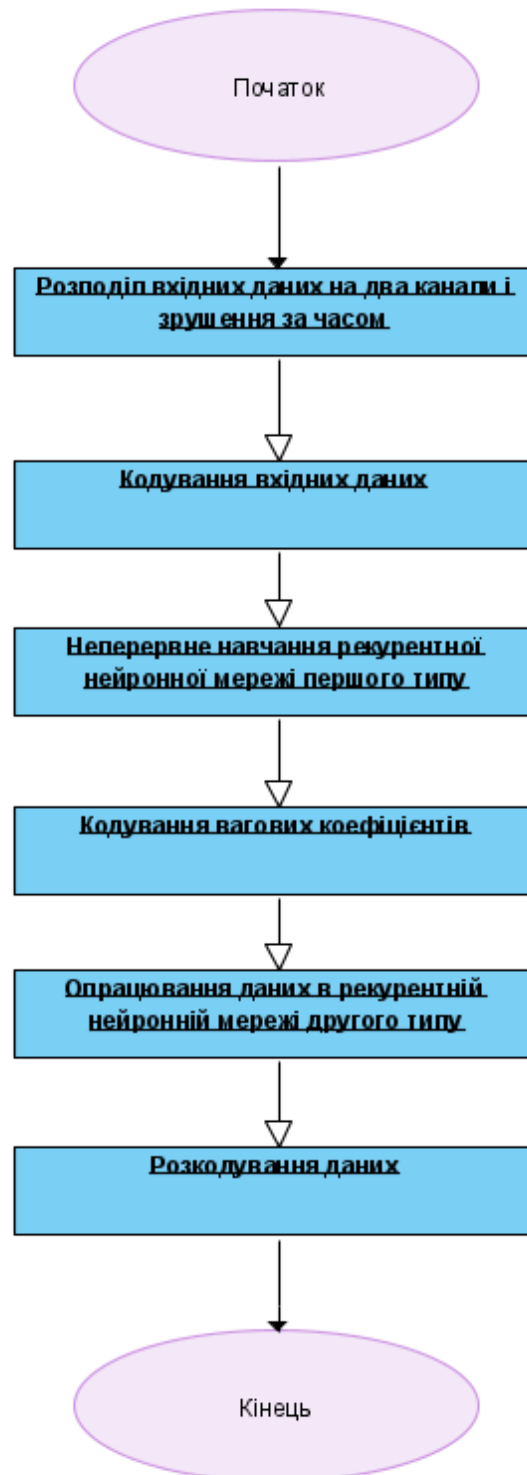


Рис. 2.4. Блок-схема, що описує виконання методу прогнозування з безперервним навчанням системи з часовими зсувами сигналів.

Крок 4. Запуск та реалізація процесу обробки даних у рекурентної нейронної мережі другого типу з посиленням асоціативного виклику у першому каналі.

Крок 5. Зворотне перетворення результатів прогнозів у рекурентної нейронної мережі другого типу з формату одиночних образів у відповідні дані вихідного формату.

Висновки до другого розділу

1. В рамках цього розділу було розроблено узагальнену модель системи нейромережевого прогнозування слабоформалізованих подій з безперервним навчанням, що відрізняється своєю структурою та правилами обробки сигналів, що забезпечують оперативне прогнозування з урахуванням змін у законах прояву подій.

2. У рамках розробленої моделі запропоновано метод нейромережевого прогнозування з безперервним навчанням: з часовими зрушеннями сигналів, що відрізняються новими правилами прогнозування та управління асоціативним викликом інформації з нейромережевої пам'яті і забезпечує високу точність прогнозів, що одержуються, для слабоформалізованих подій.

РОЗДІЛ 3

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ДЛЯ ПРОГНОЗУВАННЯ СЛАБКОФОРМАЛІЗОВАНИХ ПОДІЙ ІЗ НЕПЕРЕРВНИМ НАВЧАННЯМ

3.1. Загальні вимоги до програмної архітектури системи

Особливості програмної архітектури, що реалізує методи прогнозування часових рядів, багато в чому визначають їхню ефективність. Невдала архітектура знижує ефект використання методів, і навпаки, хороша архітектура дозволяє розширити можливості методів, покладених у її основу. Першим кроком розробки архітектури є формулювання вимог до неї.

Для програмної системи важливо як реалізовувати повною мірою заявлений функціонал, а й забезпечувати безперебійне функціонування протягом багато часу. Чим простіше програмна архітектура, тим вище її надійність і стійкість до відмов, тим нижче ймовірність виникнення помилок. Надійність одна із основних вимог, що висувуються до програмної архітектури. Особливо це стосується багатопотокових додатків, де ризик виникнення помилок, що важко налагоджуються, особливо великий. Важливо забезпечити, з одного боку, відсутність зайвих елементів у архітектурі. З іншого боку, мають бути передбачені засоби моніторингу стану програмної системи, які можна було б задіяти за потреби.

Програмна реалізація має забезпечувати як автоматичне, і ручне управління процесом її функціонування. Незважаючи на те, що впроваджена система не потребує участі оператора у своїй роботі, мають бути передбачені основні функції для ручного управління, такі як зупинка, запуск, перезапуск, оновлення прогнозу. Крім того, на етапі дослідження можливостей розроблюваної системи та її налагодження необхідні засоби налаштування параметрів та режимів роботи програмного забезпечення.

Як було зазначено у 2 розділ, для кожного типу додатків існують свої завдання та обмеження, через що необхідні різні методи прогнозування на користь таких додатків. Області застосування прогнозування часових рядів відрізняються розв'язуваними ними завданнями, середовищем функціонування системи прогнозування, вимогами до точності та горизонту прогнозів, апаратними можливостями комп'ютера, який покладено виконання прогнозів.

У другому розділі запропоновано узагальнену модель і два методи прогнозування з безперервним навчанням. Перший передбачає використання двох нейромережових каналів і тимчасових зрушень сигналів, а другий – одного каналу і прискорення внутрішнього часу рекурентної нейронної мережі. При цьому другий метод включає правила управління спрямованістю виклику сигналів з асоціативної нейромережової пам'яті.

Програмна архітектура має підтримувати можливість використання будь-якого методу. Незалежно від конкретизації, необхідно підтримувати два екземпляри нейронних мереж ідентичної конфігурації. Вони повинні функціонувати в паралельних потоках за допомогою асинхронної взаємодії між блоком управління прогнозуванням, та типами рекурентних нейронних мереж.

Наступною ключовою вимогою є можливість системи наводити вхідні дані, що надходять на обробку в програмну систему прогнозування, у формат нейронних мереж, і виконувати зворотні операції на виході. Тобто архітектура має підтримувати механізми кодування та декодування даних нейромережової обробки.

Відповідно до функціонування рекурентних нейронних мереж визначається їх власними внутрішніми блоками управління, а роботу системи загалом контролює блок управління прогнозуванням. Усі блоки керування мають набір параметрів, що визначаються користувачем при ініціалізації системи або самим блоком у процесі роботи системи. Особливості роботи рекурентних нейронних мереж полягає в тому, що не вдається формалізувати

добір значень цих параметрів, тому для ефективного використання програмної системи необхідні механізми, що дозволяють контролювати та задавати такі параметри.

Таким чином, вимоги до структури програмної архітектури, що розробляється, включають наступні пункти:

- наявність засобів моніторингу стану програмної системи;
- наявність автоматичного та ручного управління програмною системою;

наявність двох примірників рекурентних нейронних мереж та блоку управління прогнозуванням, що працюють в окремих потоках;

- можливість реалізації методів нейромережевого прогнозування з часовими зрушеннями сигналів і без часових зрушень з керуванням спрямованістю виклику сигналів з асоціативної пам'яті;

- наявність механізмів прийому вхідних даних у вихідному форматі та кодування/розкодування їх у формат, що сприймається нейронною мережею;

наявність засобів для контролю та редагування параметрів, пов'язаних з функціонуванням рекурентних нейронних мереж та програмної системи в цілому.

3.2. Програмна архітектура системи

Пропонована архітектура програмної системи нейромережевого прогнозування з безперервним навчанням представлена на рисунку 3.1, а на рисунку 3.2 для неї наведена діаграма класів UML.

Архітектура включає такі програмні модулі:

- 1) екземпляр нейронної мережі першого типу рекурентної нейронної мережі;
- 2) екземпляр нейронної мережі другого типу рекурентної нейронної мережі;

- 3) модуль попередньої обробки вхідних даних;
- 4) модуль загальної обробки вихідних даних №1;
- 5) модуль загальної обробки вихідних даних №2;
- 6) модуль читання/запису параметрів №1;
- 7) модуль читання/запису параметрів №2;
- 8) графічний інтерфейс, що складається з кнопочкових панелей та графічних модулів для кожного екземпляра рекурентної нейронної мережі.

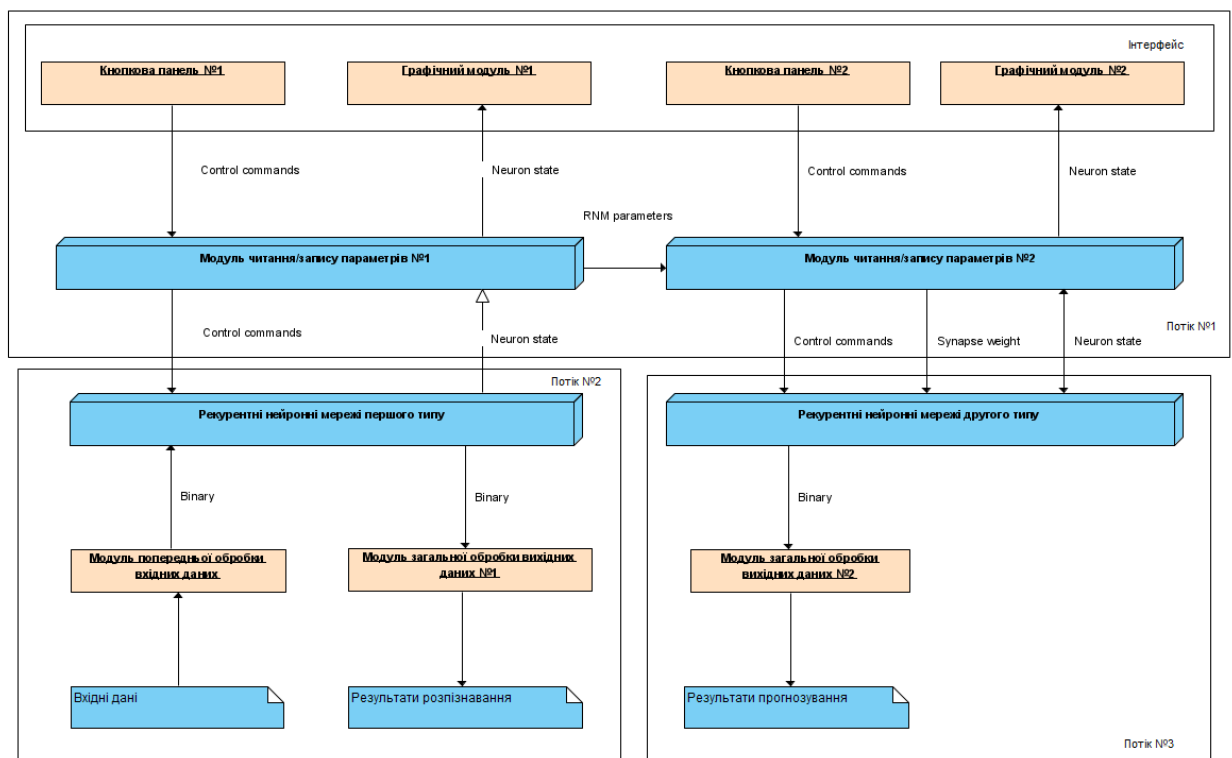


Рис. 3.1. Архітектура програмної системи нейромережевого прогнозування з безперервним навчанням

На рисунку 3.1 Control commands – керуючі команди, {RnnParams} – програмна структура з інформацією про параметри функціонування рекурентної нейронної мережі, [Binary], [Neuron State], [Synapse Weight] – масиви з бінарними даними, станами нейронів та вагами синапсів відповідно.

Примірник рекурентної нейронної мережі першого типу є реалізацією абстрактної нейромережевої програмної структури, що описує функціонал

рекурентних нейронних мереж з керованими елементами. Оскільки така мережа призначена для навчання, у цьому примірнику передбачається можливість зміни синаптичних ваг мережі. Запуск екземпляра рекурентної нейронної мережі здійснюється в окремому потоці. Обмін даними з іншими програмними модулями системи прогнозування здійснюється із застосуванням одного з механізмів міжструмової взаємодії, таких як черга. У процесі функціонування системи рекурентної нейронної мережі першого типу пов'язані з іншими модулями в такий спосіб.

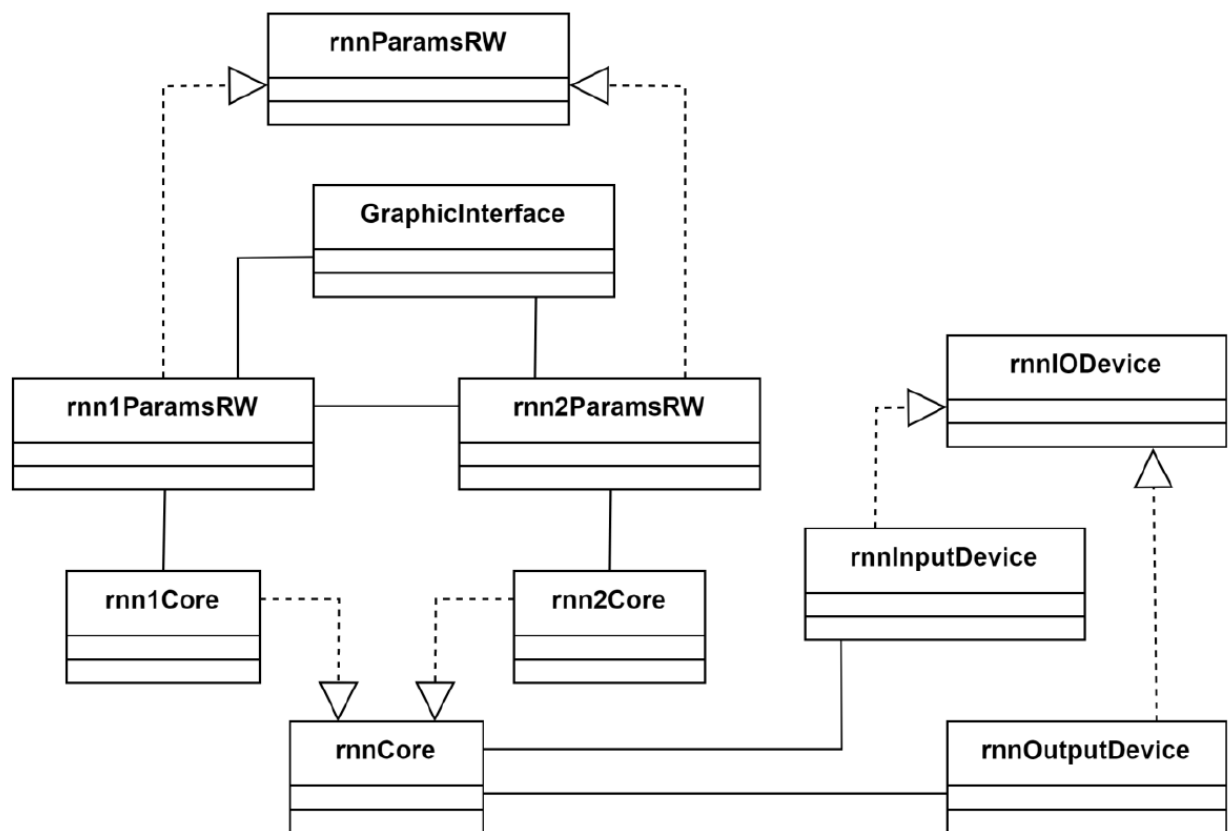


Рис. 3.2. Діаграма класів UML для паралельної архітектури програмної системи нейромережевого прогнозування з безперервним навчанням

З модуля попередньої обробки він отримує закодовані у формат дані про значення часового ряду у вигляді бінарних масивів даних ([Binary]). З модуля читання-запису параметрів №1 на нього надходять керуючі команди

(Control commands), а також передаються параметри функціонування рекурентної нейронної мережі, як структури {RnnParams}.

У зворотний бік надходить інформація про стан шарів рекурентної нейронної мережі у вигляді масиву станів нейронів [Neuron State], якщо включений режим їх відображення на інтерфейсі користувача. При отриманні команди на прогноз туди ж передається просторово-тимчасова модель рекурентної нейронної мережі у вигляді сукупності станів її нейронів (Neuron State) і ваг синапсів (Synapse Weight). Вихідні дані, що є результатами розпізнавання на шарах рекурентної нейронної мережі, передаються модуль зовнішньої обробки №1.

Екземпляр нейромережі другого типу – це реалізація абстрактної нейромережевої програмної структури за аналогією з першим типом. На відміну від першого, цей екземпляр не наділяється функціоналом, пов'язаним із навчанням. Як і в попередньому випадку, виконання рекурентної нейронної мережі винесено в окремий потік («потік №3»), а інформаційний обмін здійснюється із застосуванням механізмів міжструмової взаємодії. За аналогією з першим типом, екземпляр рекурентної нейронної мережі другого типу здійснює інформаційний обмін із модулем читання-запису параметрів №2. На вхід модуля рекурентної нейронної мережі другого типу передаються управляючі команди та скопійована просторово-тимчасова модель рекурентної нейронної мережі першого типу у вигляді масивів станів нейронів та ваг синапсів. Вихідні дані рекурентної нейронної мережі другого типу у вигляді бінарних даних, що кодують прогнозні значення, передаються в модуль зовнішньої обробки №2. При включеному режимі візуалізації станів шарів рекурентної нейронної мережі цього примірника також передаються стану нейронів.

На діаграмі класів UML (рисунок 3.1) абстрактний клас нейромережевого ядра rnnCore, яке реалізації рекурентної нейронної мережі – rnn1Core і rnn2Core відповідно.

Модуль попередньої обробки вхідних даних реалізує абстрактний клас кодувальника та відповідальний за виконання кодування. Він отримує на вхід довільні дані із зовнішнього середовища, а на виході відповідно до заданих правил формує закодоване подання у бінарному вигляді, яке передається на обробку в рекурентну нейронну мережу першого типу.

Модулі зовнішньої обробки вихідних даних №1 і №2 за аналогією з модулем попередньої обробки є реалізаціями абстрактного класу кодувальника, з тією відмінністю, що замість кодування даних формат рекурентної нейронної мережі виконують процедуру декодування. Модулі зовнішньої обробки отримують на вхід закодовані бінарні дані з виходів рекурентної нейронної мережі двох типів, а на виході формують елементи даних у вихідному форматі.

На рисунку 3.2 абстрактний клас кодувальника позначений `rnnIODevice`, а класи модулів попередньої та зовнішньої обробки – `rnnInputDevice` та `rnnOutputDevice` відповідно.

Модуль читання/запису параметрів №1 реалізує абстрактний клас, який відповідає за зберігання, редагування та передачу в рекурентній нейронній мережі параметрів, пов'язаних з керуванням цим екземпляром нейронної мережі та програмної системи загалом. Усі основні потоки даних у програмній системі проходять через цей блок. З кнопкової панелі графічного інтерфейсу на модуль читання/запису параметрів №1 надходять управляючі команди, що надсилаються з інтерфейсу. Інформація про стани шарів із рекурентної нейронної мережі першого типу передається на графічний модуль. Через модуль читання/запису параметрів №1 з рекурентної нейронної мережі першого типу на модуль читання/запису параметрів №2 передається скопійована просторово-часова модель. Крім того, модуль читання-запису параметрів №1 має власне діалогове вікно, призначене для ручного редагування його параметрів.

Модуль читання/запису параметрів №2 є реалізацією того ж класу, що і модуль читання/запису №1. Відмінністю першого модуля є відсутність

параметрів, що з управлінням системою загалом. Через модуль читання-запису параметрів №2 із модуля №1 на рекурентну нейронну мережу другого типу передається просторово-тимчасова модель рекурентної нейронної мережі першого типу. За аналогією з модулем №1, цей модуль має діалогове вікно, що викликається, для редагування своїх параметрів.

На діаграмі класів абстрактний клас модуля читання-запису параметрів наведено під ім'ям `rnnParamsRW`, а модулі читання-запису №1 та №2 – під іменами `rnn1ParamsRW` та `rnn2ParamsRW` відповідно.

Графічний інтерфейс включає програмно реалізовані кнопкові панелі №1 і №2, призначені для віддачі команд з управління системою прогнозування, і графічні модулі №1 і №2, на які надходить з відповідних модулів читання/запису параметрів інформація про стани шарів нейронних мереж. Графічні модулі виконують візуалізацію станів шарів рекурентної нейронної мережі двох типів на головному вікні програми. Графічний інтерфейс на діаграмі класів зветься `GraphicInterface`.

Модулі читання-запису параметрів №1, №2 та графічний інтерфейс складають блок управління прогнозуванням, який працює окремо від рекурентної нейронної мережі двох типів в потоці («потік №1»).

Функціонування програмної архітектури системи прогнозування може бути умовно поділено на три етапи, кожен із яких має свої особливості. Розглянемо детально кожен із етапів.

Етап 1. Ініціалізація системи.

Запуск програми починається з отримання останніх актуальних налаштувань модулями читання/запису параметрів №1 та №2. Якщо реалізація програмної системи передбачає зберігання налаштувань у зовнішньому файлі або іншому джерелі, то модулі ініціалізуються з використанням цієї інформації. Інакше використовуються стандартні налаштування.

Виходячи з отриманої інформації, визначаються параметри нейронних мереж (розміри шарів та логічних полів, параметри синапсів, нейронів,

конфігурація нейромережових шарів тощо), модулів обробки (визначається джерело даних для обробки: файл, tcp-сокет і т.д., формат даних, спосіб кодування), а також різні системні налаштування. Створюються екземпляри нейромереж, організуються канали міжструмової взаємодії. Примірники містяться в окремі потоки і запускаються.

Етап 2. Прогнозування з безперервним навчанням.

Прогнозування із безперервним навчанням є основною фазою роботи програмної системи. Ця фаза виконується у багатопотоковому режимі. На рисунку 3.3 є 3 потоки. Перший – основний, у якому працює блок управління прогнозуванням. У другому та третьому потоках функціонують рекурентні нейронні мережі відповідно. Для передачі між потоками використовується у випадку 9 каналів, призначення яких розкрито у таблиці 3.1.

Таблиця 3.1

Канали передачі в програмній архітектурі нейромережевого прогнозування з безперервним навчанням

Ідентифікатор	Виконувана функція
Канал 1	Передача параметрів функціонування рекурентної нейронної мережі другого типу із блоку управління прогнозуванням.
Канал 2	Передача параметрів функціонування рекурентної нейронної мережі першого типу із блоку управління прогнозуванням.
Канал 3	Передача просторово-часової моделі та параметрів прогнозування з блоку управління прогнозуванням у рекурентної нейронної мережі другого типу
Канал 4	Передача станів шарів рекурентної нейронної мережі другого типу блок управління прогнозуванням для візуалізації
Канал 5	Передача станів шарів рекурентної нейронної мережі першого типу блок управління прогнозуванням для візуалізації
Канал 6	Передача команди виконання прогнозу з блоку управління прогнозуванням в шарів рекурентну нейронну мережу першого типу
Канал 7	Передача просторово-часової моделі з рекурентної нейронної мережі першого типу до блоку управління прогнозуванням
Канал 8	Передача команди на завершення роботи програмної системи
Канал 9	Передача команди на завершення роботи програмної системи другого типу

2.2. Навчання. Процедура навчання є циклічною та здійснюється в рекурентній нейронній мережі першого типу потактово протягом усього часу виконання Етапу 2. Вона передбачає отримання даних із зовнішнього середовища та перетворення їх у блоці передобробки у необхідний формат. Далі відбувається передача даних у рекурентній нейронній мережі першого типу та нейромережева обробка. У процесі проходження даних через рекурентну нейронну мережу першого типу формується просторово-часова модель. На виході рекурентній нейронній мережі першого типу є результати розпізнавання даних нейронною мережею. Вони передаються блок обробки №1, та вихід системи. На кожному такті навчання рекурентної нейронної мережі першого типу перевіряє наявність команди на прогноз. Якщо така команда надійшла, рекурентна нейронна мережа першого типу передає просторово-часову модель блок управління прогнозуванням. Також перевіряється наявність команди на завершення роботи системи. Якщо така команда надійшла, виконання потоку рекурентної нейронної мережі першого типу завершується.

2.3. Прогнозування. Виконання прогнозу починається з отримання рекурентної нейронній мережі другого типу просторово-часової моделі, що передається з блоку управління прогнозуванням. Рекурентна нейронна мережа другого типу шляхом присвоєння своїм нейронам і синапсам відповідних значень просторово-часової моделі приводить свій стан у відповідність до рекурентної нейронній мережі першого типу. Крім цього, отримує з блоку управління прогнозуванням порядок та спосіб виконання прогнозів, визначений нею за станом шарів. Потім виконується обробка скопійованих даних рекурентної нейронній мережі другого типу з формуванням прогнозних даних. Ці дані передаються блок обробки №2, на виході якого є прогнозні дані у вихідному форматі. За аналогією з першою при отриманні команди на завершення роботи потік рекурентної нейронній мережі другого типу завершується.

2.4. Візуалізація процесу обробки. Процес візуалізації включає потактову передачу станів шарів рекурентної нейронної мережі першого та другого типів в блок управління прогнозуванням і відображення цих даних на графічному інтерфейсі. Цей процес є опціональним і виконується на розсуд користувача.

Етап 3. Завершення роботи. Завершення роботи починається з подання блоком управління прогнозуванням відповідних команд на рекурентні нейронні мережі. При цьому відбувається завершення роботи потоків №2 та №3. Знищуються екземпляри блоків обробки даних. Останні актуальні налаштування системи, які зберігаються в блоках читання/запису параметрів №1 та №2, зберігаються на зовнішньому носії, якщо така функція передбачена реалізацією. Після цього відбувається звільнення ресурсів системи та виконання програми завершується.

3.3. Програмна реалізація системи

Програмна система нейромережевого прогнозування, що реалізує запропоновані моделі та методи та використовувалася в експериментах, реалізована мовою програмування C++ з використанням фреймворку Qt версії 5.

Розроблена програмна система є конкретизацією архітектур, викладених у попередніх параграфах. Файлова структура представлена на рисунку 3.3. Розглянемо докладніше реалізацію кожного з елементів програмної реалізації.

Перший і другий екземпляри рекурентної мережі є екземплярами класів `rnn1Core` та `rnn2Core` відповідно, які у свою чергу реалізують клас `rnnCore`. Частина функціоналу нейромережевих ядер та допоміжних функцій винесені в окремі файли `rnnCore_WCoeff`, `rnnCore_Structures`, `rnnCore_Support`, `rnnCore_Synapses`.

Клас `rnnCore` визначає основну логіку функціонування нейромережевого ядра. У файлах `rnnCore_Structures` винесено оголошення та реалізація основних елементів нейронної мережі, таких як нейрон, синапс, логічне поле тощо. Клас `rnnCore_Synapses` містить опис блоків динамічних синапсів РНС, в клас `rnnCore_WCoeff` винесені змінні та функції, пов'язані з навчанням мережі. Клас `rnnCore_Support` надає допоміжні функції, такі як визначення індексу нейрона в масиві за його координатами на шарах мережі тощо.

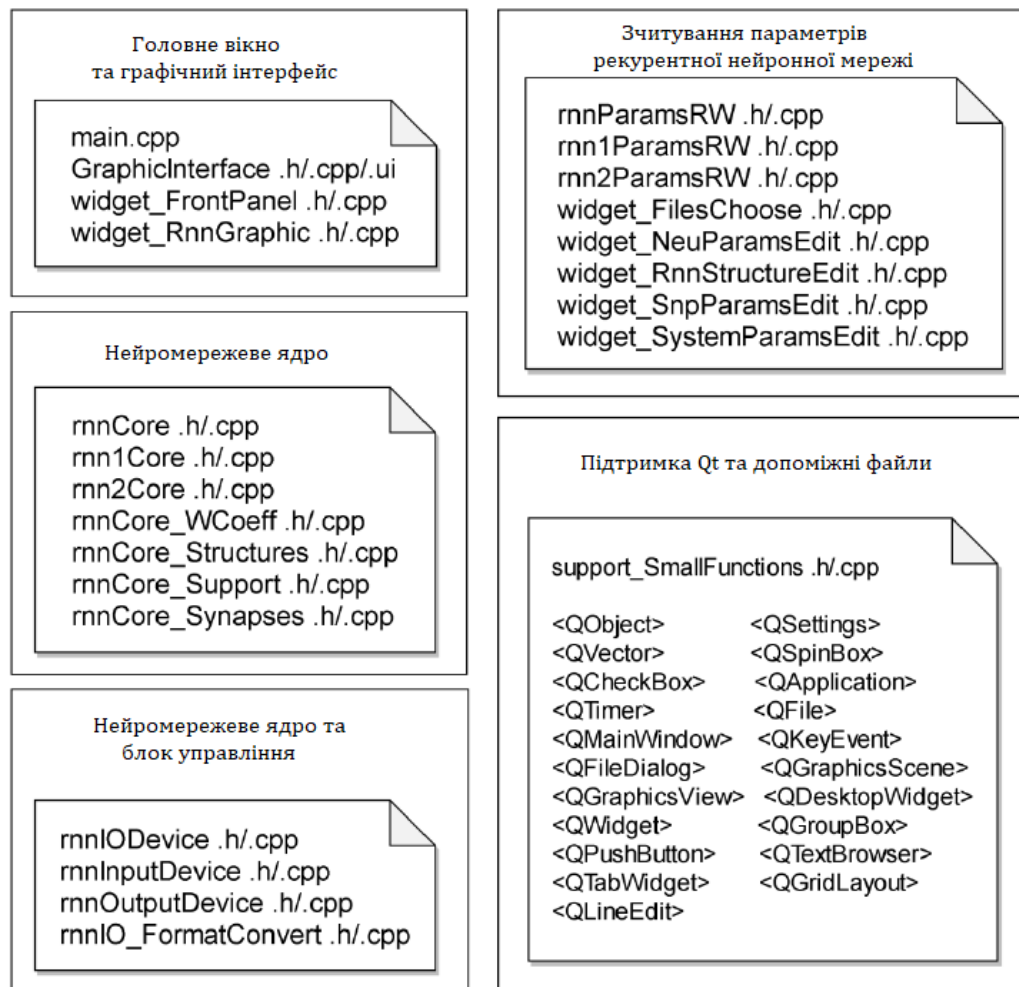


Рис. 3.5. Файлова структура програмного забезпечення, що реалізує методи нейромережевого прогнозування з безперервним навчанням

Модулі читання/запису параметрів №1 та №2 містять основні налаштування, пов'язані з функціонуванням рекурентної нейронної мережі двох типів відповідно, а також загальні налаштування програмної системи.

Абстрактна логіка, пов'язана з ними, описана у класі `rnnParamsRW`, а реалізації – у класах `rnn1ParamsRW` та `rnn2ParamsRW`. Налаштування редагуються через діалогове вікно, яке викликається відповідною кнопкою з панелі графічного інтерфейсу. Його елементи описані у класах `widget_FilesChoose`, `widget_NeuParamsEdit`, `widget_RnnStructureEdit`, `widget_SnpParamsEdit`, `widget_SystemParam`.

Модуль передобробки вхідних даних включає емулятор пристрою введення, описаний у класі `rnnInputDevice`, а модуль постобробки – відповідно пристрій виведення `rnnOutputDevice`. Дані класи успадковуються від абстрактного класу вводу-виводу `rnnIODevice`, на основі якого можуть бути реалізовані пристрої для різних типів вхідних даних. У існуючому ПЗ реалізовано пристрій для читання даних із зовнішніх файлів. Крім пристрою введення, модулі перед- та постобробки містить функції, відповідальні за конвертування даних із файлів у формат РНС. Ці функції реалізовані у класі `rnnIO_FormatConvert`.

Графічний інтерфейс, реалізований у класі `GraphicInterface`, відображено на рисунках 3.6-3.8. На цих скріншотах представлені стани графічного інтерфейсу у різних режимах роботи програмної системи. Графічний інтерфейс (рис. 3.6) включає кнопку панель з кнопками налаштування параметрів, запуску, зупинки, очищення пам'яті нейронної мережі, команди на прогноз, а також перемикача режимів функціонування («навчання» – «прогнозування»). Кнопкова панель реалізована у класі `widget_FrontPanel`. Вікно для службових повідомлень інформує користувача про статус процесів, що протікають усередині системи, повідомляє про помилки у разі виникнення.

Графічний модуль відображає нейромережеві шари. Натиснувши на відповідну вкладку, можна отримати візуальне відображення стану першого або другого шару мережі. Реалізація графічного модуля перебуває у класі `RnnGraphic`.

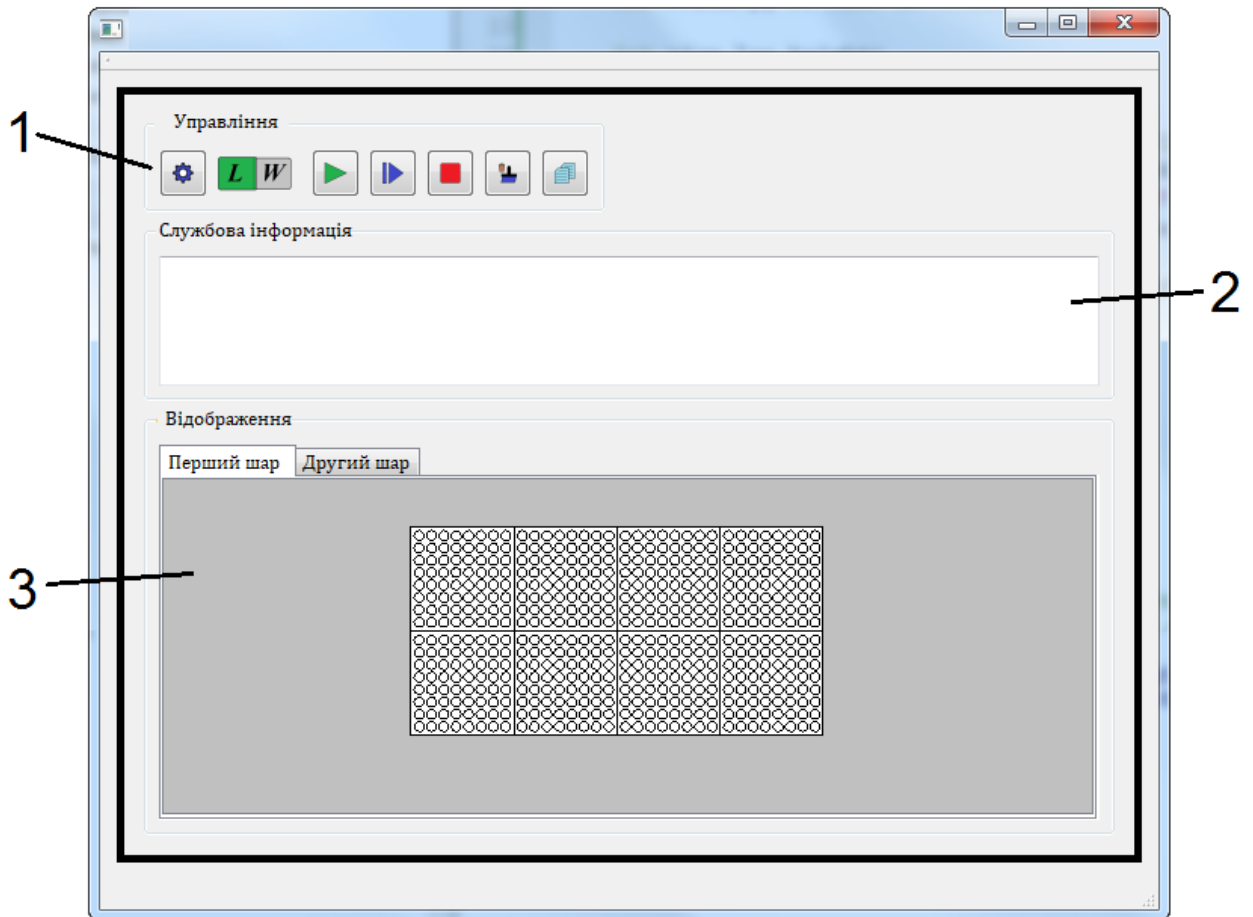


Рис. 3.6. Інтерфейс програмної системи нейромережевого прогнозування (у режимі навчання).

На рисунку 3.6 - 1 – кнопкова панель, 2 – вікно для системних повідомлень, 3 – графічний модуль для відтворення шарів нейронної мережі

При виконанні прогнозу на інтерфейсі відображаються графічні елементи, пов'язані з другим екземпляром (рисунок 3.7). На рисунку 3.8 продемонстровано стан графічного інтерфейсу під час прогнозування. Видно, що на шарах рекурентної нейронної мережі здійснюється обробка даних, закодованих у вигляді сукупностей збуджених нейронів.

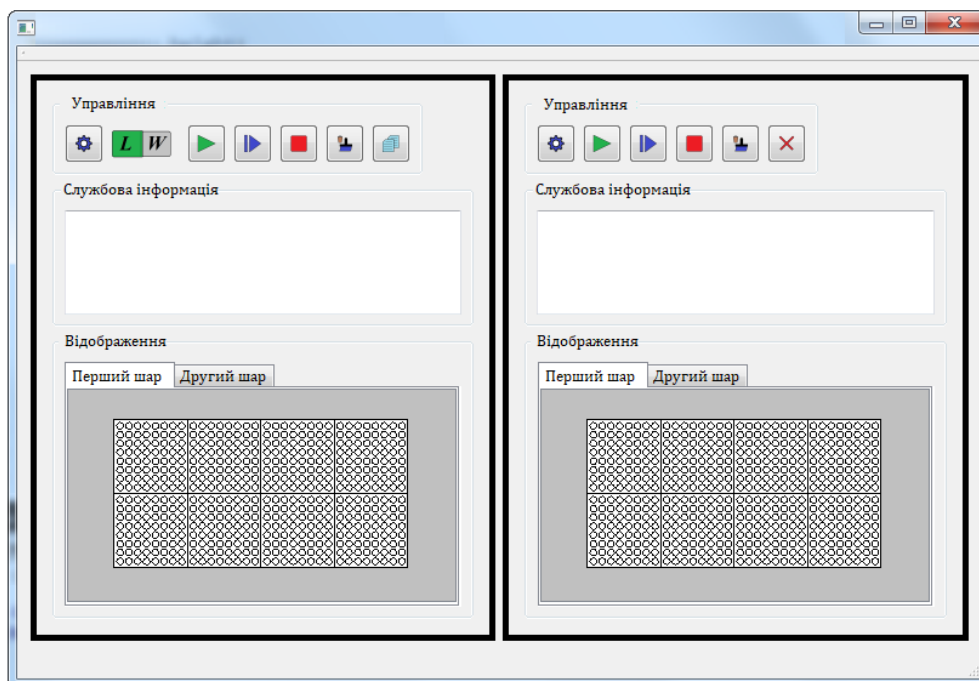


Рис. 3.7. Інтерфейс програмної системи неймережевого прогнозування (у режимі навчання та прогнозування).

Ліворуч на рисунку 3.7 – панель для мережі першого типу, праворуч – другого

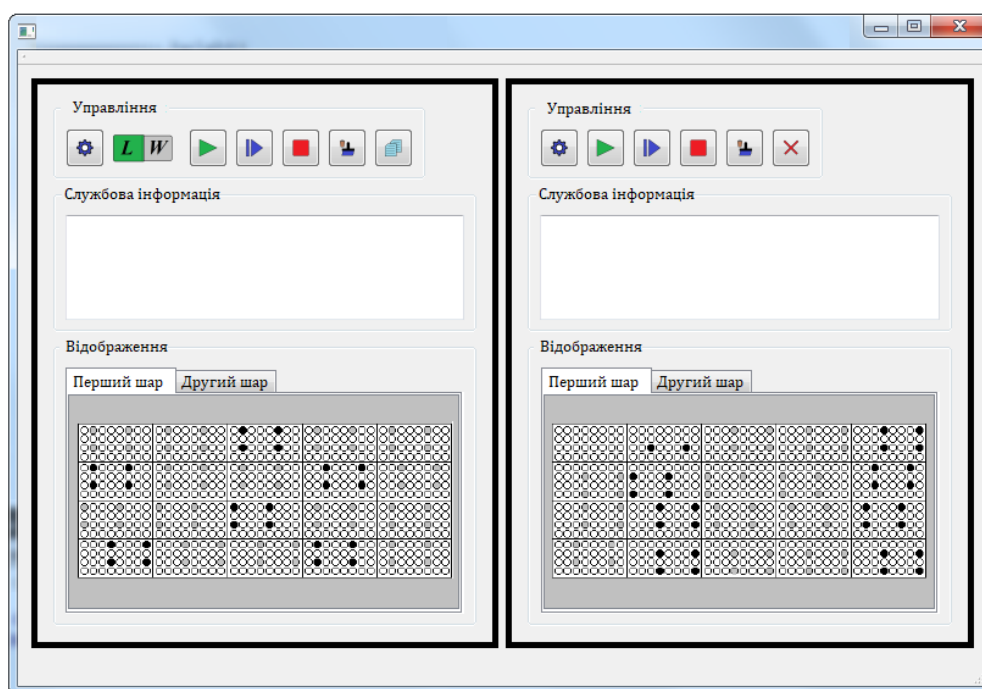


Рис. 3.8. Функціонування програмної системи неймережевого прогнозування з безперервним навчанням.

На шарах білими кружками позначені нейрони в стані очікування, сірими – рефракторні нейрони, а чорними кружками – збуджені нейрони.

Графічний інтерфейс, рекурентні нейронні мережі двох типів запущені в трьох різних потоках, за рахунок чого забезпечується, по-перше, відсутність «зависання» інтерфейсу при виконанні процесів в окремих екземплярах, а по-друге, незалежність та паралельність роботи екземплярів. Взаємодію між потоками забезпечують сигнально-слотові з'єднання.

3.4. Експериментальні дослідження з нейромережевого прогнозування на прикладі прогнозування подій, пов'язаних з дорожнім рухом

Для перевірки працездатності запропонованих методів та програмних архітектур було проведено низку експериментальних досліджень. Як вихідні дані для прогнозування використовувалося три набори даних. Перший набір збирався автором самостійно, другий та третій є загальнодоступними.

Перший набір зібраний з інтернет-сервісу, який інформує в реальному часі про транспортну ситуацію в м. Тернопіль. Сервіс надавав інформацію про середні швидкості руху на поточний момент у конкретній точці, а також давав прогноз для неї наступного дня за допомогою вбудованих у нього алгоритмів. Було вибрано 8 точок, в яких у період з 14 по 27 вересня 2022 року в моменти часу {00:00, 03:00, 06:00, 09:00, 12:00, 15:00, 18:00, 21:00} реєструвалася інформація про реальну швидкість руху та прогноз, що надається сервісом на один день. Таким чином, для навчання та тестування була сформована база даних з $8 \times 9 \times 14 \times 8 \times 2 = 1792$ значень.

Другий і третій набори – дані, які збирає лабораторія дослідження транспортних даних (TDRL). Цей набір містить інформацію про дорожню ситуацію, отриману з датчиків. Дані знімаються із 4500 датчиків, інтервал

запису становить 30 секунд. Дані представлені у бінарному форматі. Файли другого набору мають розширення «.v30» та містять інформацію про обсяг трафіку, записану в 1-байтному форматі. Файли третього набору з розширенням ".С30" містять дані про зайнятість (оссурансу - відсоток часу, протягом якого точка на дорозі зайнята транспортними засобами) у 2-байтному вигляді.

Другий набір даних містить інформацію про завантаженість доріг у період з 12 по 24 вересня 2022 року з детекторів №№ 17, 22, 24, 52, 62, 83.

Третій набір містить дані щодо обсягу трафіку в період з 12 по 24 вересня 2022 року року з детекторів №№ 51, 56, 61, 77, 80, 90, 453.

Експерименти щодо визначення точності прогнозування швидкостей руху. У першій серії експериментів ставилося завдання визначення точності прогнозування швидкостей руху в Тернополі із застосуванням моделей та методів прогнозування з безперервним навчанням. Для цього дані першого набору були розбиті на дві частини. Перші 12 днів використовувалися для навчання нейронної мережі, а під прогнозування залишалися дані за останні 2 дні. Інтервал прогнозів також дорівнював двом дням.

Наявні дані спочатку кодувалися у формат, що сприймається нейронною мережею. Діапазон швидкостей руху розбивався на 8 піддіапазонів, кожному з яких на логічних полях ставилися у відповідність певні нейрони. Приклад кодування щодо розміру 8×4 нейронів логічних полів наведено в Таблиці 3.1.

Для вивчення можливостей прогнозування методом з часовими зрушеннями були сформовані дві ідентичні мережі з числом нейронів у кожному шарі, що дорівнює 1920 одиниць. Шари мереж були поділені на логічні поля розміром 8×8 нейронів. Кількість таких полів у кожному рядку становила 6, а кожному стовпці – 5. Кожне логічне поле ділилося на два підполя.

Таблиця 3.1

Кодування піддіапазонів швидкостей руху на логічних полях розміром
8×4 нейронів

Швидкість км/год.	Матриця логічного поля
[0-5]	1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
[5-10]	0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0
[10-15]	0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0
[15-20]	0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0
[20-25]	0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0
[25-30]	0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0
[30-35]	0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0
[35-40]	0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1

При навчанні мережі зрушений часовий ряд кодованих швидкостей подавався в перше підполі вхідного поля, а поточний ряд часовий - у другому підполі. Кодування швидкостей руху відображено у Таблиці 3.1. Наявність у

На рисунку 3.9, 1 – напрямок руху вздовж шару; 2, 3 – збуджені нейрони (чорні вузли) у першому та другому підполях відповідно; 4, 5 – нейрони в станах очікування (білі вузли) та несприйнятливості (сірі вузли) відповідно.

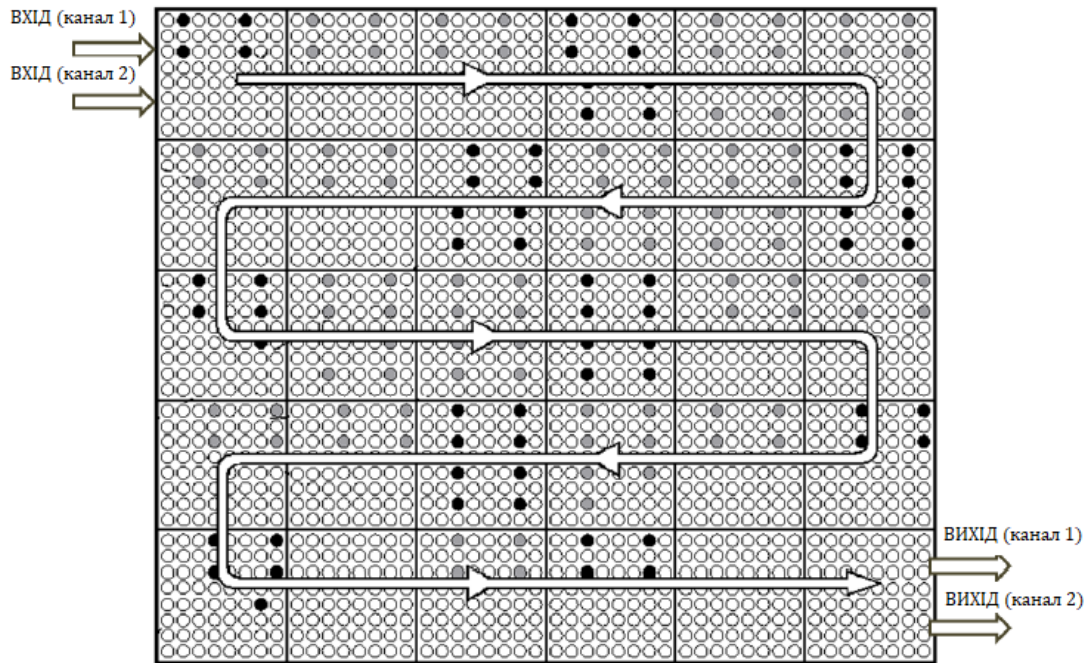


Рис. 3.10. Приклад стану першого шару рекурентної нейронної мережі під час навчання

Також було досліджено можливість прогнозування швидкостей руху методом без часових зрушень. Кількість нейронів у кожному шарі мережі у цих експериментах становило 960 одиниць. Кожен шар цих мереж був поділений на 30 логічних полів розміром 8×4 нейрона. Відмінність цих мереж від попередньої версії полягала лише в зменшених розмірах логічних полів. Схема просування за шарами мережі аналогічна попередньому прикладу. На рекурентній мережі першого типу також було покладено завдання безперервного навчання, зв'язування оброблюваних сигналів у просторі та часі, зберігання їх на елементах мережі. Для прогнозування стан мережі також копіювався в другий тип. Після цього при прямому

прогнозуванні значно прискорилося обробка сигналів. Приклад стану першого шару для мережі другого типу як прогнозування показаний на рисунку 3.11.



Рис. 3.11. Стан першого шару рекурентної нейронної мережі другого типу під час прогнозування методом без часових зрушень

Помилка прогнозування визначалася з використанням метрик середньої абсолютної помилки (MAE), середньоквадратичної помилки (RMSE) та середньої абсолютної процентної помилки (MAPE). У таблиці 3.2 наведено результати прогнозування.

Графіки на рисунках 3.12 та 3.13 демонструють отримані результати прогнозування порівняно з реальними швидкостями руху. Результати прогнозування показані суцільними лініями, а реальні дані – пунктирними.

Як видно з таблиці 3.2, обидва методи характеризуються високими показниками для умов, що розглядаються. Дещо краще з точки зору MAE та MAPE відзначається метод прогнозування з мчасовими зрушеннями (перший метод). Однак перший метод важче реалізувати, ніж другий.

Крім того, другий метод (без часових зрушень) за рахунок прискорення обробки інформації в дозволяє отримувати прогнози швидше, ніж перший метод.

Таблиця 3.2

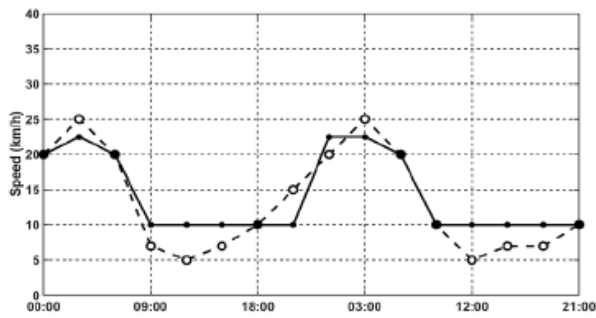
Показники ефективності запропонованих методів

Точки спостереження	Метод прогнозування з часовими зрушеннями			Метод прогнозування без часових зрушень		
	MAE (км/год)	MAPE (%)	RMSE (км/год)	MAE (км/год)	MAPE (%)	RMSE (км/год)
Вулиця 1	3.31	29.10	5.01	3.47	27.20	4.47
Вулиця 2	4.28	29.10	5.44	4.75	29.80	5.69
Вулиця 3	2.16	27.30	2.85	2.39	26.70	2.95
Вулиця 4	2.22	18.00	3.89	2.69	20.70	4.06
Вулиця 5	1.72	11.50	2.61	1.88	12.00	2.69
Вулиця 6	5.97	44.20	9.15	5.87	39.00	8.02
Вулиця 7	1.88	9.60	3.06	3.44	15.80	4.15
Вулиця 8	1.63	16.60	2.77	1.94	17.50	2.88
Середнє	2.89	23.20	4.35	3.30	23.60	4.36

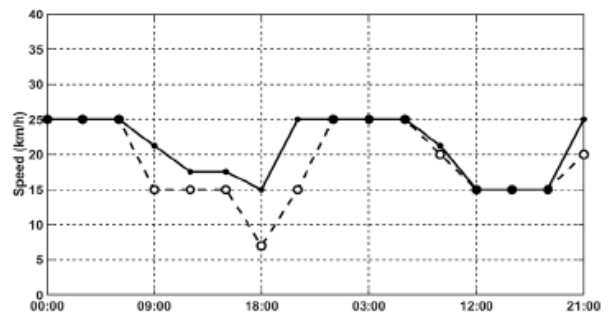
Використовуючи ті ж вихідні дані, було оцінено ефективність прогнозування транспортних потоків з використанням інтернет-сервісу Google Maps, а також моделі ARIMA та нейронної мережі LSTM. Результати порівняння запропонованих методів із цими підходами наведено у таблиці 3.3.

Аналіз таблиці 3.3 показує, що запропоновані методи у метриках MAE, MAPE, RMSE перевершують відомі рішення. Зокрема, виграти запропонованого методу з часовими зрушеннями порівняно з методом LSTM становлять: 11.1% MAPE, 31.5% MAE, 15.5% RMSE.

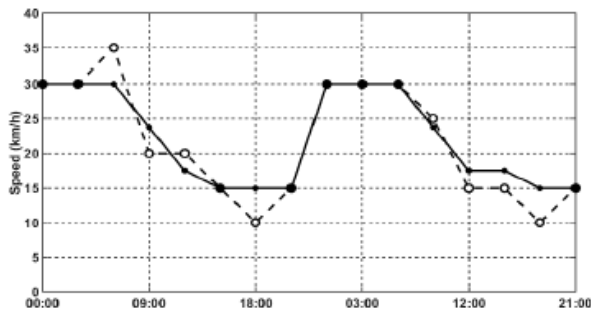
Вулиця 1



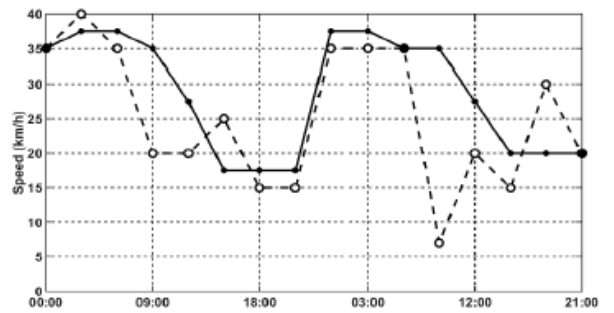
Вулиця 2



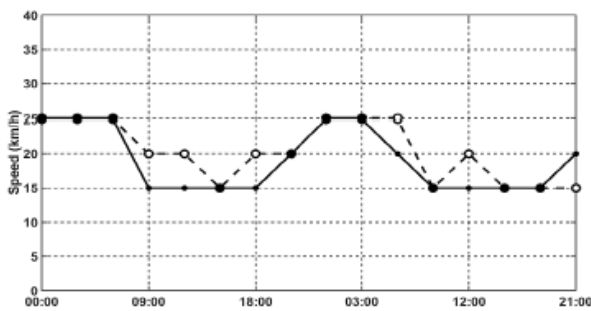
Вулиця 3



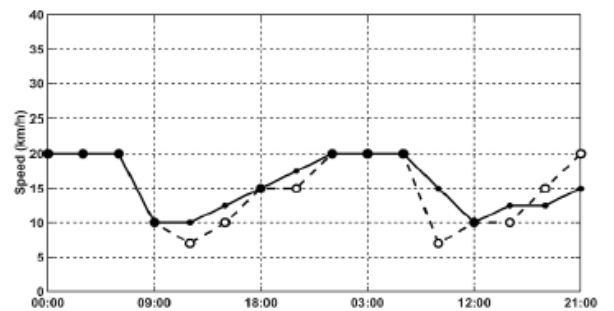
Вулиця 4



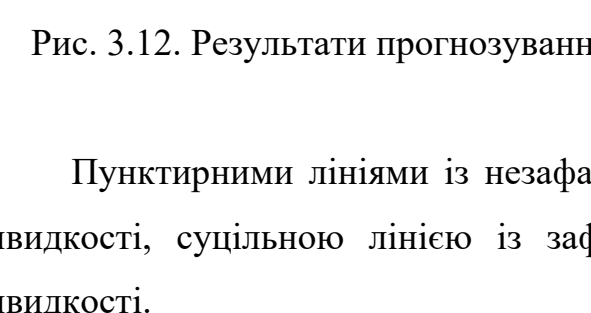
Вулиця 5



Вулиця 6



Вулиця 7



Вулиця 8

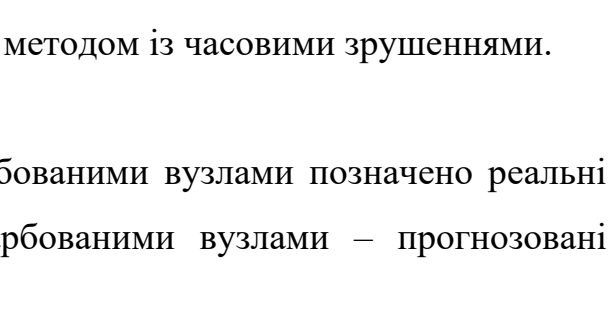
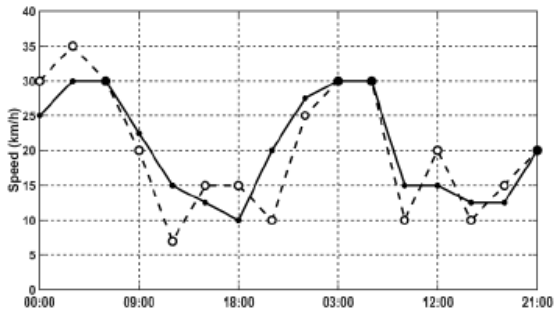


Рис. 3.12. Результати прогнозування методом із часовими зрушеннями.

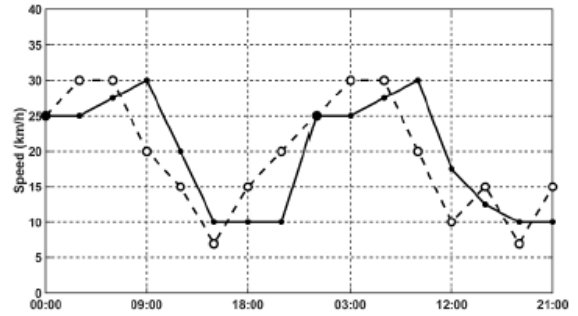
Пунктирними лініями із незафарбованими вузлами позначено реальні швидкості, суцільною лінією із зафарбованими вузлами – прогнозовані швидкості.

Таким чином, до основних переваг запропонованих методів можна віднести:

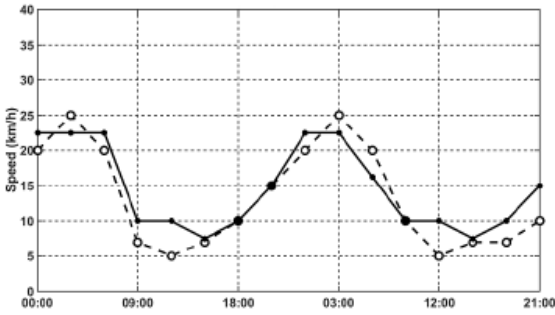
- 1) вищу точність прогнозування проти відомих рішень;
- 2) відсутність необхідності попередньої підготовки та перепідготовки моделі, що дає можливість працювати в режимі реального часу.



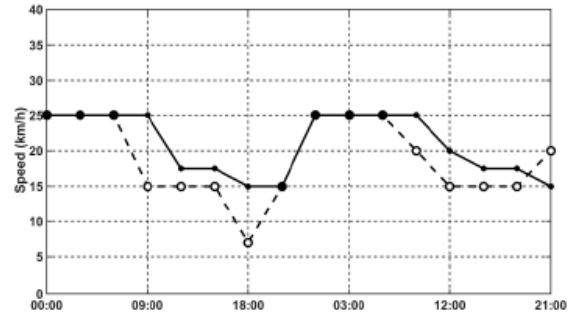
Вулиця 1



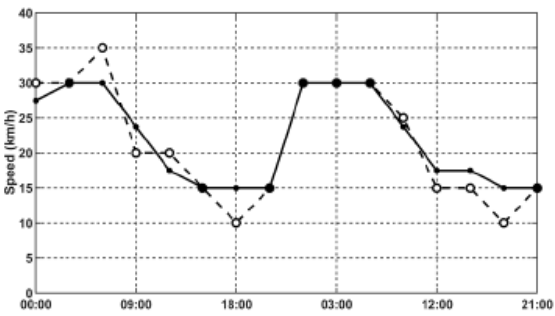
Вулиця 2



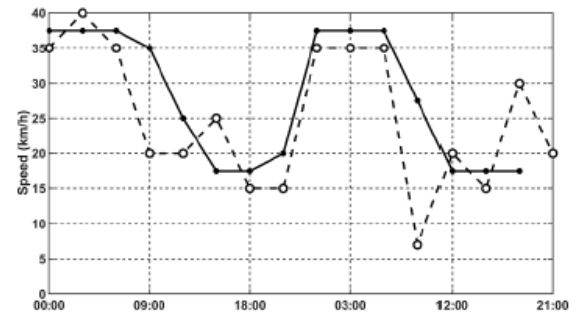
Вулиця 3



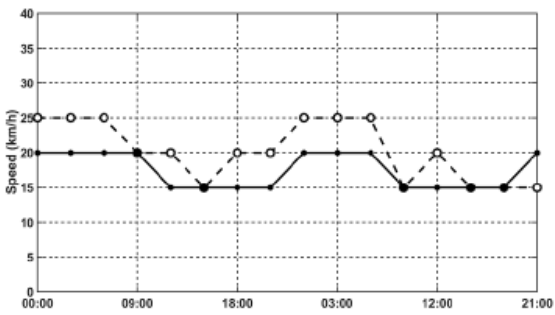
Вулиця 4



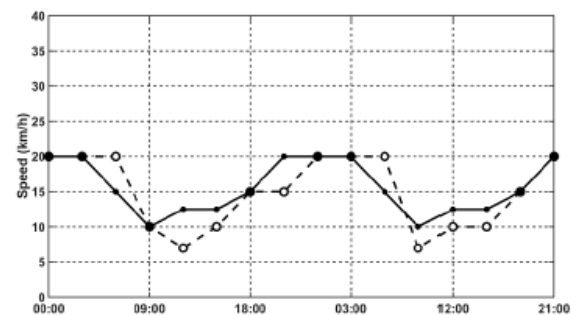
Вулиця 5



Вулиця 6



Вулиця 7



Вулиця 8

Рис. 3.13. Результати прогнозування шляхом без часових зрушень

Експерименти щодо визначення точності прогнозування завантаженості доріг. Третя серія експериментів проводилася на третьому наборі даних і мала на меті визначення оптимальної конфігурації

нейромережевих шарів для прогнозування подій. Рекурентні нейронні мережі з керованими елементами мають можливість наділення шарів різними конфігураціями: лінійними (рис. 3.14 А), петлевими (рис. 3.14 Б), спіральними з постійним радіусом (рис. 3.14 В), що сходяться (рис. 3.14 Г) і розбіжними рис. 3.14 Д) спіралями.

У цій серії експериментів за інших рівних параметрів нейронної мережі оцінювалися можливості прогнозування рекурентними нейронними мережами з керованими елементами. Оцінюючи точності використовувалися самі метрики, що у попередніх експериментах.

Дані з третього набору за 12-23 вересня 2022 року використовувалися для навчання мережі, тоді як прогнозування виконувалося на інтервал, що дорівнює 1 добі (24 вересня 2022 року). Інтервал становив 3 години, а кількість прогнозованих значень – 8.

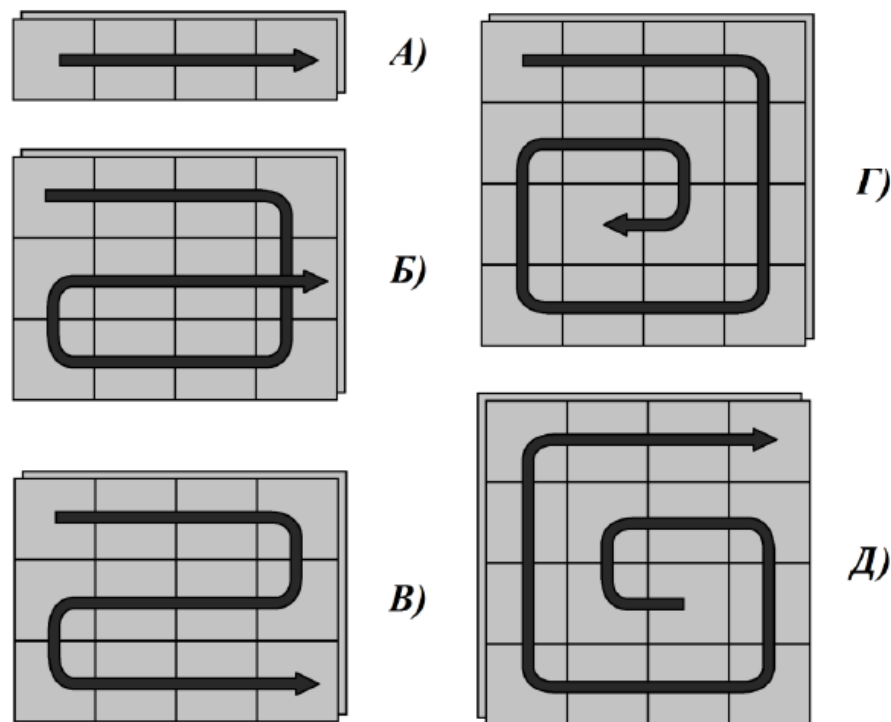


Рис. 3.14. Результати прогнозування шляхом без часових зрушень

Результати показують, що всі п'ять конфігурацій продемонстрували досить високу точність прогнозування обсягу дорожнього руху. Вона вища за

точність ARIMA. Найкращі результати характерні спіральним структурам, які виявилися кращими, ніж структури лінійні.

Спостережуваний результат пояснимо перевагою спіральних нейромережових конфігурацій з погляду щільності асоціативного зв'язування викликаних сигналів з тими сукупностями образів, які вже знаходяться на шарах нейронної мережі. У той самий час лінійна структура може забезпечити відносно щільне зв'язування лише найближчих образів, а вплив інших одиничних образів у міру видалення стає дуже малим.

Висновки до третього розділу

1. Наведено файлову структуру програмної системи нейромережового прогнозування, що реалізує запропоновані методи та архітектури. Розкрито особливості функціонування програмного забезпечення, а також представлені графічні інтерфейси програмної системи на різних етапах роботи.

2. Здійснено моделювання розроблених моделей та методів прогнозування слабкоформалізованих подій стосовно завдань прогнозування транспортних потоків.

3. Описано практичні рекомендації щодо підвищення точності та використання програмних систем нейромережового прогнозування слабкоформалізованих подій з безперервним навчанням, що забезпечують підвищення точності прогнозів за рахунок визначення найбільш ефективної конфігурації нейромережових шарів стосовно задачі прогнозування подій і розробки нових правил вибору методу та архітектури в залежності від умов, яких функціонує система прогнозування.

ВИСНОВКИ

В процесі виконання магістерської роботи отримано наступні наукові та практичні результати:

1. Проведено аналіз відомих методів та систем прогнозування подій, за результатами якого було встановлено, що вони не задовольняють умовам, які мають місце при функціонуванні часових рядів зі складною динамікою. Показано, що існуючі підходи не враховують ряд особливостей, властивих таким часовим рядам.

2. Здійснено постановку задачі магістерського дослідження із врахуванням особливостей функціонування інтелектуалізованих систем.

3. Розроблено узагальнену модель системи нейромережевого прогнозування слабкоформалізованих подій з безперервним навчанням, що відрізняється своєю структурою та правилами обробки сигналів, що забезпечують оперативне прогнозування з урахуванням змін у законах прояву подій.

4. У рамках розробленої моделі запропоновано метод нейромережевого прогнозування з безперервним навчанням: з часовими зрушеннями сигналів, що відрізняються новими правилами прогнозування та управління асоціативним викликом інформації з нейромережевої пам'яті і забезпечує високу точність прогнозів, що одержуються, для слабкоформалізованих подій.

5. Наведено файлову структуру програмної системи нейромережевого прогнозування, що реалізує запропоновані методи та архітектури. Розкрито особливості функціонування програмного забезпечення, а також представлені графічні інтерфейси програмної системи на різних етапах роботи.

6. Здійснено моделювання розроблених моделей та методів прогнозування слабкоформалізованих подій стосовно завдань прогнозування транспортних потоків.

7. Описано практичні рекомендації щодо підвищення точності та використання програмних систем нейромережевого прогнозування слабкоформалізованих подій з безперервним навчанням, що забезпечують підвищення точності прогнозів за рахунок визначення найбільш ефективної конфігурації нейромережевих шарів стосовно задачі прогнозування подій і розробки нових правил вибору методу та архітектури в залежності від умов, яких функціонує система прогнозування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мінаєв Ю. М., Філімонова О. Ю. Розв'язання прикладних інженерних задач в нейронних мережах : навч.-метод. посібник. Ч. 1. Теоретичні основи штучних нейронних мереж та головні передумови для розв'язування прикладних задач. Київ : Національний авіаційний університет, 2003. 75 с.
2. Надеран С. В. Розпізнавання будівель на супутникових зображеннях надвисокої роздільної здатності за допомогою нейронних мереж : автореф. дис. ... канд. техн. наук : 05.12.23. Київ, 2015. 18 с.
3. Олійник А. О., Субботін С. О., Олійник О. О. Еволюційні обчислення та програмування. Запоріжжя : ЗНТУ, 2010. 324 с.
4. Системи штучного інтелекту: нечітка логіка, нейронні мережі, нечіткі нейронні мережі, генетичний алгоритм : монографія / В. П. Лисенко, В. М. Решетюк, В. М. Штепа та ін. Київ : НУБіП України, 2014. 332 с.
5. Тимошук П. В. Штучні нейронні мережі : навчальний посібник. Львів : Видавництво Львівської політехніки, 2011. 444 с.
6. Cecchi S., Spinsante S., Terenzi A., Orcioni A. A Smart Sensor-Based Measurement System for Advanced Bee Hive Monitoring. *Sensors*. 2020. Vol. 20 (9), Article 2726. DOI: <https://doi.org/10.3390/s20092726>.
7. Classification of general audio data for content-based retrieval / D. Li, I. K. Sethi, N. Dimitrova [et al.] *Pattern Recognition Letters*. 2001. № 22(5). P. 533–544.
8. Geron A. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. Sebastopol : O'Reilly, 2017. 861 p.
9. Hu W., Lv J., Liu D., Chen Y. Unsupervised Feature Learning for Heart Sounds Classification Using Autoencoder. *Journal of Physics: Conference Series*. 2018. Vol. 1004. P. 1–9. DOI: <https://doi.org/10.1088/1742-6596/1004/1/012002>.