УДК 681.3.06

І.А. Терейковський,
Національний технічний університет України
“Київський політехнічний інститут”

КРИТЕРІЇ ВИБОРУ АРХІТЕКТУРИ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ РОЗВ’ЯЗАННЯ ЗАДАЧ З ЗАХИСТУ ІНФОРМАЦІЇ

Запропоновано методи і критерії оптимізації архітектури нейронної мережі, призначеної для розв’язання актуальних задач з захисту інформації в комп’ютерних системах та мережах. Наведено приклади оптимізації архітектури нейронної мережі, призначеної для виявлення комп’ютерних вірусів.

Ключові слова: архітектура нейронної мережі, захист інформації, комп’ютерний вірус, нейронна мережа, оптимізація.

In article are offered methods and criteria to optimization of the architecture neuronetworks, intended for decision of the actual problems of protection to information in computer system and set. Cite an instance determinations of the optimum architectures neuronetworks, intended for revealing computer virus.

Key words: architecture to neuronetworks, protection to information, computer virus, neuronetworks, optimization.

Вступ

Упродовж останніх 10–15 років прогрес галузі захисту інформації в комп’ютерних системах та мережах безпосередньо пов’язаний з використанням різноманітних теорій штучного інтелекту, серед яких особливе місце займає теорія штучних нейронних мереж (НМ). У більшості відомих прикладів НМ використовуються з метою:

- аналізу параметрів підзахисної системи з метою розпізнавання атак;
- аналізу параметрів підзахисної системи з метою розпізнавання вразливостей;
- управління параметрами системи захисту;
- реалізації паралельних обчислень з метою зменшення терміну формування та застосування захисних заходів.

Результати досліджень [1, 2] вказують на те, що в загальнопоширених методиках використання НМ в засобах захисту інформації питання визначення оптимальної архітектури мережі далекі від свого вирішення. Разом з тим доведено, що ефективність використання НМ безпосередньо залежить від того, наскільки вид та параметри її архітектури відповідають умовам поставленої задачі [1, 2]. Отже, розроблення методики оптимізації архітектури НМ відносно умов задач захисту інформації може доволі позитивно вплинути на рівень безпеки комп’ютерних систем та мереж.

Вимоги до архітектури

Аналіз сучасних публікацій показує, що для забезпечення ефективності використання НМ їх архітектура повинна відповідати деяким вимогам, які характеризують:

1) навчальні дані;
2) процес навчання;
3) обчислювальні потужності;
4) вихідну інформацію;
5) технічну реалізацію;
6) сферу застосування.

Розглянемо вказані вимоги детальніше.

1. Основними характеристиками навчальних даних є:

- Кількість параметрів, що характеризують навчальний приклад. У деяких випадках, наприклад, під час аналізу текстової інформації, кількість параметрів не може бути визначена апріорно. Крім того, для різних навчальних прикладів кількість вхідних параметрів може відрізнятись.
- Вид параметрів – дискретний (символьний) чи безперервний (числовий).
- Обсяг навчальних прикладів.
- Наявність помилок (шуму) в навчальних прикладах.
- Наявність кореляції навчальних прикладів.
- Можливість та необхідність попередньої обробки вхідних даних з метою їх нормалізації та видалення шуму.
- Можливість відображення в навчальній вибірці всіх аспектів процесу, що моделюється..
- Пропорційність навчальних прикладів, що відповідають різним аспектам процесу, що моделюється..

2. Процес навчання характеризується:

- Терміном навчання.
- Необхідністю представлення в навчальних даних очікуваного вихідного сигналу НМ. Таким чином визначається можливий тип навчання – з вчителем або без вчителя.
- Можливістю автоматизації процесу навчання, яка визначається кількістю та важливістю емпіричних параметрів. Вказана можливість багато в чому визначає умови застосування НМ. Мережі в яких процес навчання не автоматизовано, можна використовувати тільки в лабораторних умовах.
- Можливістю донавчання на експлуатації.
- Вимогами до якості навчання, яке звичайно оцінюють за величиною максимальної та середньої помилки розпізнавання навчальних та тестових навчальних даних. Тестові дані повинні не значно відрізнятись від навчальних.
- Можливістю навчання НМ в лабораторних умовах. Наприклад, в лабораторних умовах потенційно можливо навчити НМ розпізнавати мережеві атаки певного типу. До того ж неможливо навчити НМ класифікувати електронні листи відповідно інтересам конкретного користувача. Доцільність навчання в лабораторних умовах пояснюється потребами оптимального механізму створення та оновлення бази знань НМ.
- Незмінністю виходу мережі для різних прикладів з однаковими параметрами.

3. На практиці вимоги до обчислювальних потужностей визначаються:

- Максимальною кількістю прикладів (обсяг пам'яті), яку може запам'ятати мережа для досягнення необхідної достовірності прийняття рішення.
- Достовірністю прийняття рішення, що характеризується допустимими величинами максимальної та середньої помилки мережі на реальних даних, котрі в загальному випадку можуть виходити за межі множини навчальних даних.
- Можливістю екстраполяції результатів навчання за межі навчальних прикладів.

4. Вимоги до вихідної інформації НМ визначаються

- Виглядом, в якому має бути подана вихідна інформація. Наприклад, при розпізнаванні вірусів може виникнути необхідність не тільки визначення ситуації типу “несправність в програмному забезпеченні комп’ютерної мережі”, але й розрахунку ймовірності цієї ситуації або графічного відображення таких ситуацій на площину, яке дозволить провести остаточну класифікацію користувачеві.
- Необхідністю визначення вербальних залежностей між вхідною та вихідною інформацією.

5. Обмеження реалізації НМ стосуються:

- Швидкості прийняття рішення.
- Можливості інтеграції в наявні засоби захисту.
- Обсягу програмної реалізації.

6. Сфера застосування визначає:

- Вид задач, які розв'язуватимемо в засобах захисту за допомогою НМ. Сьогодні доведена висока ефективність використання НМ для класифікації та кластеризації образів. Трохи гіршою вважається ефективність застосування під час розв'язання задач управління та для змістовного аналізу інформації природною мовою. У перспективних дослідженнях вказується на доцільність застосування НМ з метою реалізації паралельних розрахунків в комп'ютерних системах, що дозволить значно підвищити їх стійкість від перенавантаження.
- Можливість пристосувати мережу до автономного функціонування.

Характеристика задачі оптимізації

Розглянемо процес використання НМ у засобах захисту як складну систему, стан якої можна оцінювати за деякою множиною критеріїв

$$F = \{\alpha_i f_i\}_n, \tag{1}$$

де f_i – значення *i*-го критерію, α_i – ваговий коефіцієнт *i*-го критерію, *n* – кількість критеріїв.

У такому разі система оцінки ефективності НМ за набором критеріїв, котрі відповідають перерахованим вимогам, описується так:

$$F = \{\{\alpha_i f_i\}\}_6, \tag{2}$$

де $\{f_1\}$– критерії навчальних даних, $\{f_2\}$– критерії процесу навчання НМ, $\{f_3\}$ – критерії обчислювальних потужностей, $\{f_4\}$ – критерії вихідної інформації, $\{f_5\}$ – критерії технічної реалізації, $\{f_6\}$– критерії сфери застосування.

Своєю чергою

$$\{f_1\} = \{f_{1,1}, f_{1,2}, f_{1,3}, f_{1,4}, f_{1,5}, f_{1,6}, f_{1,7}\}, \tag{3}$$

де $f_{1,1}$ – можливість навчатись на прикладах з апріорно невизначеною кількістю параметрів, $f_{1,2}$ – пристосованість до виду параметрів, $f_{1,3}$ – можливість навчання на зашумлених даних, $f_{1,4}$ – можливість навчання на прикладах, що корелюються, $f_{1,5}$ – необхідність попередньої обробки навчальних даних, $f_{1,6}$ – можливість навчання на прикладах, які не відображають всіх аспектів процесу, $f_{1,7}$ – пропорційність навчальних прикладів.

$$\{f_2\} = \{f_{2,1}, f_{2,2}, f_{2,3}, f_{2,4}, f_{2,5}, f_{2,6}, f_{2,7}\}, \tag{4}$$

де $f_{2,1}$ – тривалість терміну навчання, $f_{2,2}$ – наявність у навчальних прикладах вихідного сигналу, $f_{2,3}$ – автоматизація навчання, $f_{2,4}$ – до навчання на експлуатації, $f_{2,5}$ – якість навчання, $f_{2,6}$ – навчання в лабораторних умовах, $f_{2,7}$ – незмінність виходу в навчальних прикладах.

$$\{f_3\} = \{f_{3,1}, f_{3,2}, f_{3,3}, f_{3,4}\}, \tag{5}$$

де $f_{3,1}$ – обсяг пам'яті, $f_{3,2}$ – максимальна помилка класифікації, $f_{3,3}$ – середня помилка класифікації, $f_{3,4}$ – можливість екстраполяції результатів навчання.

$$\{f_4\} = \{f_{4,1}, f_{4,2}, f_{4,3}\}, \tag{6}$$

де $f_{4,1}$ – можливість представлення рішення у вигляді ймовірності, $f_{4,2}$ – можливість представлення рішення у графічному вигляді, $f_{4,3}$ – можливість вербалізації рішення.

$$\{f_5\} = \{f_{5,1}, f_{5,2}, f_{5,3}\}, \tag{7}$$

де $f_{5,1}$ – швидкість прийняття рішення, $f_{5,2}$ – пристосованість до інтеграції, $f_{5,3}$ – обсяг програмного забезпечення.

$$\{f_6\} = \{f_{6,1}, f_{6,2}, f_{6,3}, f_{6,4}, f_{6,5}\}, \tag{8}$$

де $f_{6,1}$ – пристосованість до класифікації образів, $f_{6,2}$ – пристосованість до кластеризації образів, $f_{6,3}$ – можливість аналізу інформації природною мовою, $f_{6,4}$ – пристосованість до визначення управлінських рішень, $f_{6,5}$ – автономність функціонування.

Після підстановки (3)–(8) в (2) отримаємо загальний вираз для оцінки ефективності НМ

$$F\left\{\left\{f_{1,i}\right\}_7,\left\{f_{2,i}\right\}_7,\left\{f_{3,i}\right\}_4,\left\{f_{4,i}\right\}_3,\left\{f_{5,i}\right\}_3,\left\{f_{6,i}\right\}\right\}. \quad (9)$$

При знаходженні значення від критеріїв доцільно привести їх до безрозмірного вигляду

$$\overline{f}_i = \frac{f_i^{max} - f_i}{f_i^{max} - f_i^{min}}, \quad (10)$$

де $\overline{f}_i$ – безрозмірне значення критерію f_i, f_i^{max} та f_i^{min} – максимальне та мінімальне значення критерію f_i.

Визначення критеріїв оптимізації дозволяє записати задачу оптимізації архітектури НМ у такому вигляді:

$$\begin{cases} F \to max, \\ \overline{f}_i^{min} \le \overline{f}_i \le \overline{f}_i^{max}, \end{cases} \quad (11)$$

де $\overline{f}_i^{min}, \overline{f}_i^{max}$ – мінімальна та максимальна допустима величина безрозмірного значення *i*-го критерію опитмізації.

Зазначимо, що розв'язком (11) має бути визначення оптимального виду архітектури НМ та визначення оптимальних параметрів цієї мережі. Відповідно вираз (11) потрібно переписати так:

$$\begin{cases} F(a_j) \to max \\ \overline{f}_i^{min}(a_j) \le \overline{f}_i(a_j) \le \overline{f}_i^{max}(a_j) \end{cases}, \quad (12)$$

де a_j – *j*-та архітектура НМ, $\overline{f}_i(a_j)$ – безрозмірне значення *i*-го критерію для *j*-ї архітектури, $\overline{f}_i^{min}(a_j), \overline{f}_i^{max}(a_j)$ – мінімальна та максимальна допустима величина безрозмірного значення критерію для *j*-ї архітектури.

Врахуємо в (12) обмеженість множини відомих архітектур НМ

$$\begin{cases} F(a_j) \to max \\ \overline{f}_i^{min}(a_j) \le \overline{f}_i(a_j) \le \overline{f}_i^{max}(a_j), \\ a_j \in A,\ j = 1,2,...,L \end{cases} \quad (13)$$

де A – множина допустимих архітектур НМ, L – кількість допустимих архітектур.

Окреслимо множину допустимих рішень (13). Сьогодні відома велика кількість видів архітектур НМ, однак найбільшого поширення набули: багатошаровий перспетрон (БШП), мережа радіальної базисної функції (РБФ), ймовірнісна мережа (PNN), мережа адаптивної резонансної теорії (АРТ), топографічна карта Кохонена (ТК) та асоціативні мережі (АНМ). Структурні моделі цих мереж показані на рис. 1–6. Перераховані архітектури вже стали класичними, а відповідно [4] більшість вдалих нейромережевих рішень реалізовано за рахунок адаптації їх параметрів до поставлених практичних задач.

Зазначимо, що перераховані НМ мають порівняно стабільну топологію, яка може не принципово та/або доволі повільно змінюватись під час навчання. Наприклад, під час багатоітераційного навчання БШП методом "нейронний газ" в його структуру можна додавати нові сховані нейрони. Прикладами не принципової зміни топології можуть слугувати РБФ та PNN, в яких кількість схованих нейронів співвідноситься з кількістю навчальних прикладів. Також не принципово змінюється топологія АРТ, архітектура якої передбачає можливість додати під час навчання в шар розпізнавання новий нейрон. Стабільність топології не дозволяє побудувати мережу, в якій кількість вхідних параметрів та обсяг пам'яті можуть змінюватись під час функціонування, що звужує сферу їх застосування в задачах захисту інформації. Це вказує на необхідність застосування НМ з динамічною топологією, які менше досліджені та відомі, хоча і більше схожі зі своїм біологічним прототипом, що вказує на потенційно високі можливості. Кількість вхідних, вихідних

та схованих нейронів, зв'язки між ними, а також кількість нейронних шарів таких мереж визначається безпосередньо під час їх навчання та функціонування. Прикладом НМ з динамічною структурою є створена для класифікації тексту семантична нейронна мережа (СНМ). Основною перевагою СНМ є можливість розпізнавання образів, що характеризуються необмеженою кількістю детермінованих параметрів [5].

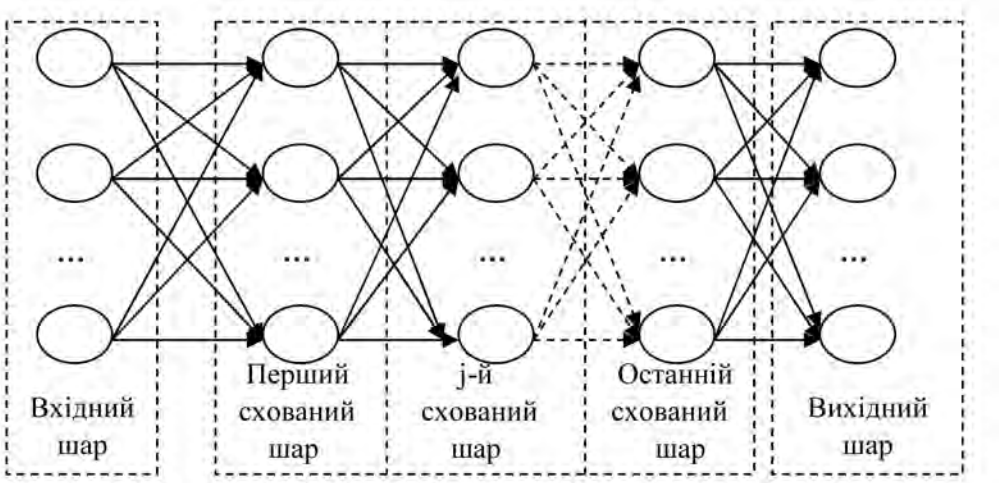

Рис. 1. Структура БШП

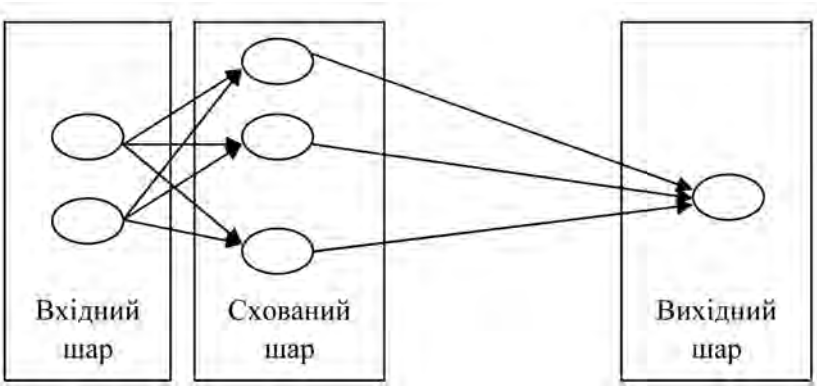

Рис. 2. Спрощена структура РБФ

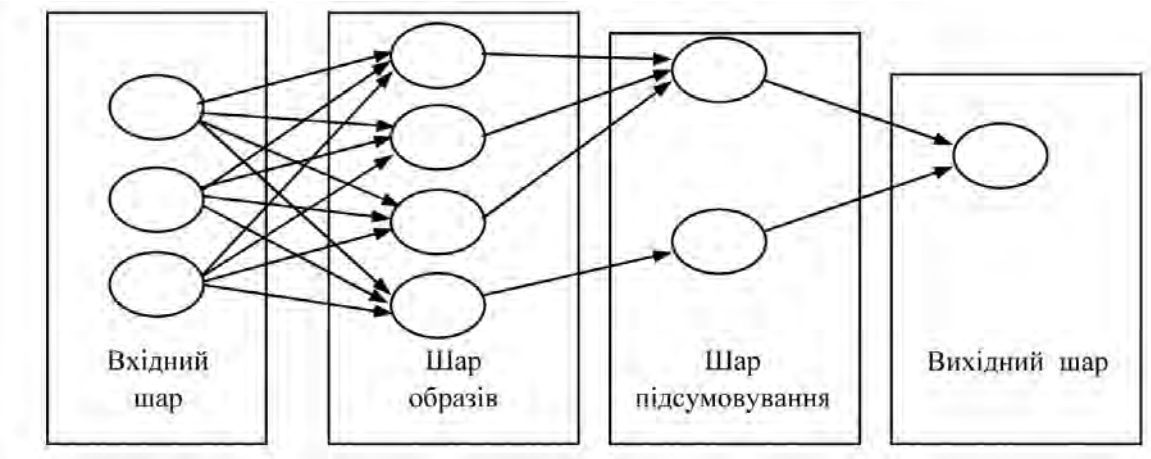

Рис. 3. Структура PNN

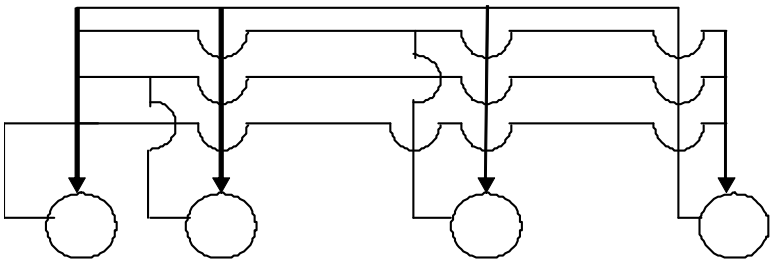

Рис. 4. Структура АНМ Хопфілда

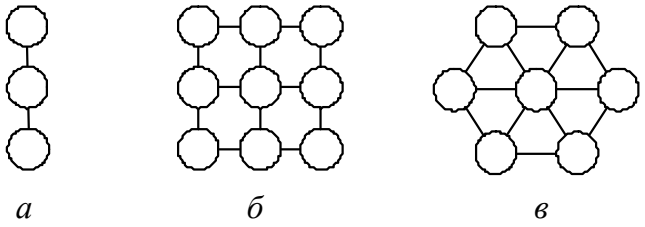

Рис. 5. Структура карти Кохонена з лінійною (а), квадратною (б) та гексагональною(в)сіткою зв'язків

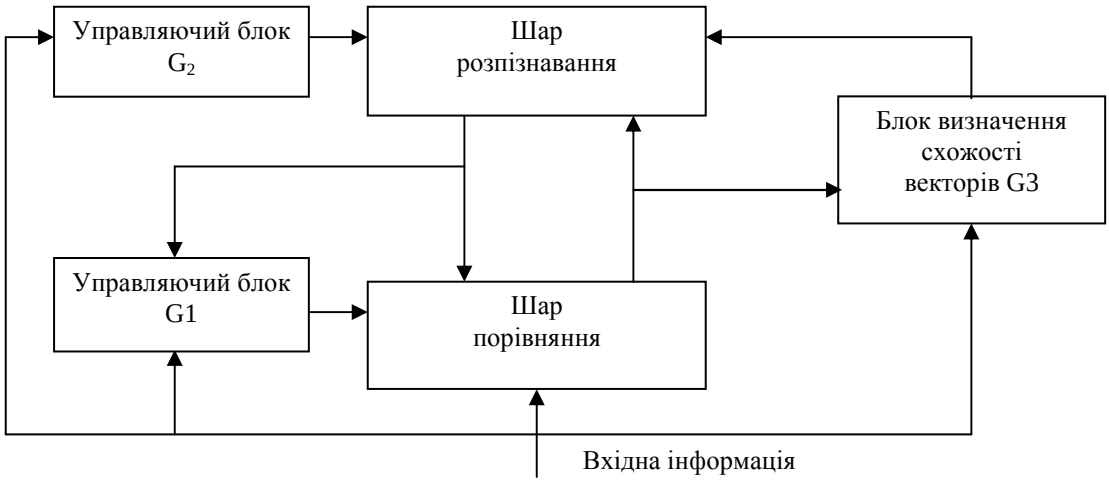

Рис. 6. Блок-схема АРТ-1

Отже, множина допустимих архітектур набуде такого вигляду:

$$A = \{БШП, РБФ, PNN, ТК, АРТ, АНМ, СНМ\}. \quad (14)$$

Відповідно (13) зміниться так:

$$\begin{cases} F(a_j) \to max \\ \overline{f}_i^{min}(a_j) \le \overline{f}_i(a_j) \le \overline{f}_i^{max}(a_j). \\ a_j \in A,\ j = 1,2...,7 \end{cases} \quad (15)$$

У результаті порівняльного аналізу допустимих архітектур НМ у першому наближенні визначено величини критеріїв оптимізації (3)–(8). Вказані величини оцінок, виставлені за дискретною трибальною шкалою величини, наведені в табл. 1. Критерій f_i=1, якщо він повністю забезпечується в архітектурі, f_i=0 – якщо забезпечується частково і f_i=-1 – якщо не забезпечується.

Деталізуємо постановку задачі (15), врахувавши в ній багатокритеріальний характер оптимізації

$$\begin{cases} \{\{\alpha_i f_i(a_j)\}\}_6 \to max \\ \overline{f}_i^{min}(a_j) \le \overline{f}_i(a_j) \le \overline{f}_i^{max}(a_j), \\ a_j \in A,\ j = 1,2...,7 \end{cases} \quad (15)$$

де $f_i(a_j)$ – значення *i*-го критерію для *j*-ї архітектури.

Таблиця 1

Величини критеріїв оптимізації

№	Архітектура НМ						
	БШП	РБФ	ТК	АРТ	СНМ	PNN	АНМ
$f_{1,1}$	-1	-1	-1	-1	1	-1	-1
$f_{1,2}$	-1	-1	-1	-1	1	-1	-1
$f_{1,3}$	1	-1	-1	0	1	-1	-1
$f_{1,4}$	1	0	1	-1	1	0	-1
$f_{1,5}$	1	1	1	1	1	1	-1
$f_{1,6}$	-1	1	1	-1	-1	1	0
$f_{1,7}$	1	-1	-1	-1	-1	-1	0
$f_{2,1}$	-1	0	1	1	0	1	1
$f_{2,2}$	1	1	-1	-1	-1	1	1
$f_{2,3}$	1	-1	0	1	1	1	0
$f_{2,4}$	0	1	1	1	1	1	0
$f_{2,5}$	1	0	0	1	1	1	1
$f_{2,6}$	1	1	0	0	1	1	0
$f_{2,7}$	0	1	0	1	1	1	1
$f_{3,1}$	1	-1	-1	-1	0	-1	0
$f_{3,2}$	1	-1	-1	-1	0	-1	0
$f_{3,3}$	1	-1	-1	-1	0	-1	1
$f_{3,4}$	1	1	0	1	1	1	0
$f_{4,1}$	0	0	-1	-1	-1	1	0
$f_{4,2}$	-1	-1	1	-1	-1	-1	-1
$f_{4,3}$	1	0	-1	-1	-1	0	-1
$f_{5,1}$	1	1	1	1	0	1	-1
$f_{5,2}$	1	1	1	1	0	1	1
$f_{5,3}$	-1	1	-1	0	-1	-1	0
$f_{6,1}$	1	1	1	1	0	1	1
$f_{6,2}$	-1	-1	1	0	0	-1	-1
$f_{6,3}$	-1	-1	1	0	1	0	-1
$f_{6,4}$	-1	-1	1	-1	-1	-1	1
$f_{6,5}$	-1	-1	-1	1	1	-1	-1

У виразах (1), (2), (15) ваговий коефіцієнт *i*-го критерію α_i можна інтерпретувати з погляду важливості відповідної вимоги в поставленій задачі захисту інформації. Наприклад, у задачі розпізнавання комп'ютерних вірусів розв'язок традиційно представляється у вигляді ймовірності, а вимога графічного представлення не висувається. Відповідно ваговий коефіцієнт $\alpha_{4,1}$ для критерію $f_{4,1}$ дорівнює 1, а ваговий коефіцієнт $\alpha_{4,2}$ для критерію $f_{4,2}$ дорівнює 0. Очевидно, що вагові коефіцієнти критеріїв потрібно розраховувати для поставленої задачі захисту інформації індивідуально. Можливо застосувати метод експертних оцінок. Також відзначимо, що результатом оптимізації може бути не один, а декілька видів архітектур.

Розробка методики оптимізації

Проведені дослідження та результати [2] дозволяють запропонувати трьохетапну методику оптимізації архітектури НМ. На першому етапі потрібно визначити номенклатуру оптимальних видів архітектур, на другому етапі оптимізувати параметри архітектур, а на третьому – в результаті порівняльних експериментів визначити, яка з НМ з оптимальною архітектурою та оптимізованими параметрами найповніше відповідає вимогам поставленої задачі.

Методика визначення оптимального виду архітектури складається з таких етапів:

1. Сформулювати постановку задачі в практичному аспекті, вказати необхідність, мету та задачі застосування НМ.

2. Використовуючи (10) та дані табл. 1, розрахувати безрозмірні значення критеріїв оптимізації (3)–(9).

3. Врахувавши особливості поставленої задачі захисту, розрахувати вагові коефіцієнти кожного з критеріїв оптимізації.

4. На основі (15) визначити оптимальний тип (типи) моделі НМ.

Розглянемо етап оптимізації архітектури НМ. Результати аналізу НМ типу БШП, РБФ, PNN, АРТ, ТК, АНМ та СНМ. вказують на те, що їх розробляють за однаковим алгоритмом, блок-схема якого показана на рис. 7.

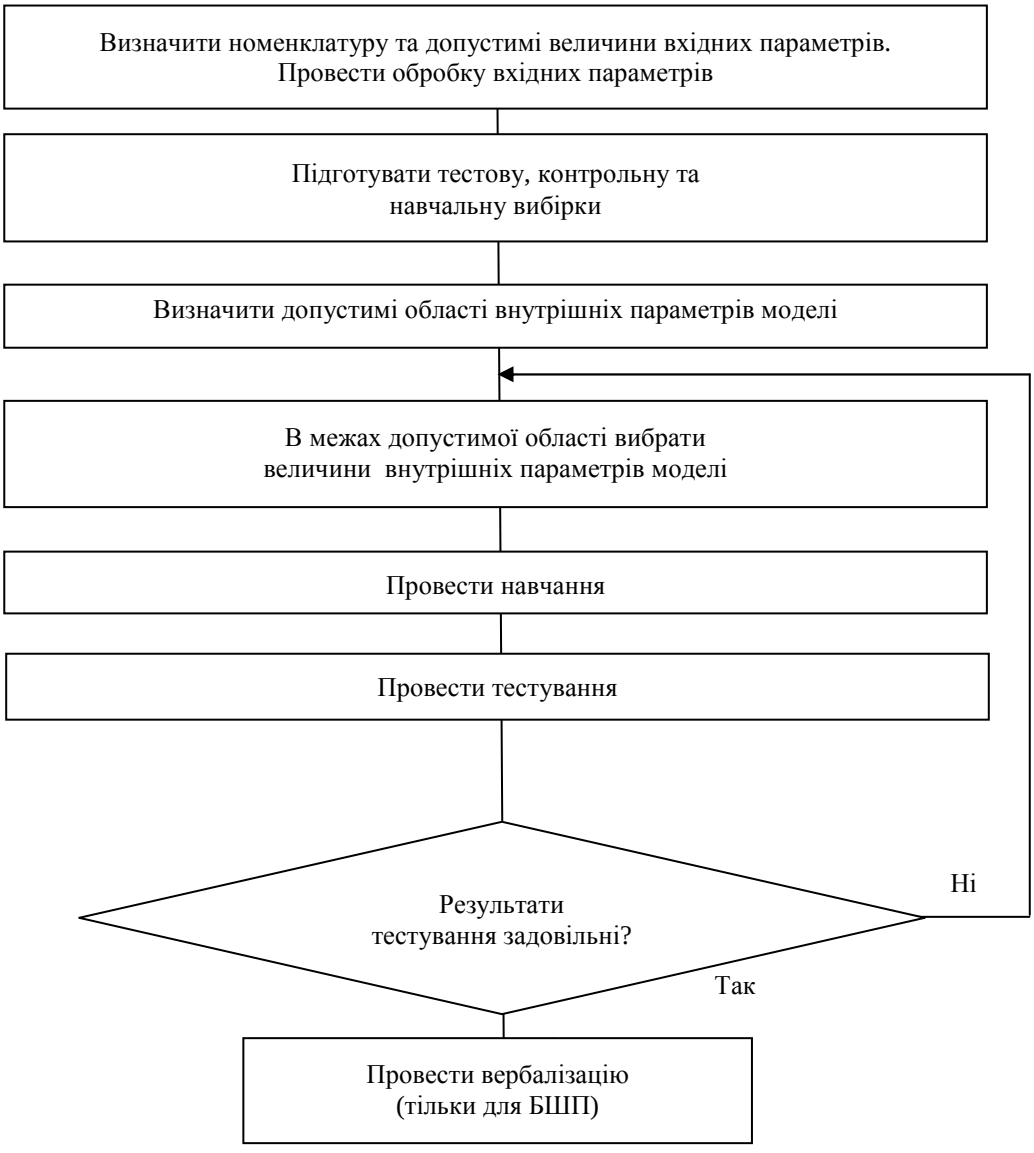

Рис. 7. Алгоритм розробки моделі НМ

Використання цього алгоритму та результатів аналізу актуальних задач ЗІ, проведеного в першому розділі, дозволяють сформувати методику оптимізації параметрів архітектури НМ обраного виду до умов поставленої задачі захисту:

1. Підготовка вхідної інформації полягає в визначенні та обробці контрольованих параметрів захисту до виду, прийнятного для НМ. Процес визначення контрольованих параметрів базується на умовах задачі захисту. Його основним результатом є номенклатура та вид (бінарний та/або неперервний) вхідних та вихідних параметрів НМ. Для НМ типу ТК визначається номенклатура тільки вхідних параметрів. Обробка параметрів, як правило, полягає в їх нормалізації на інтервалі [-1,1].
2. Підготовка навчальної, контрольної та тестової вибірки – полягає в формуванні множини прикладів вхідних та вихідних параметрів НМ, яке реалізується на основі аналізу експлуатаційних та експериментальних даних.
3. Розрахунок параметрів моделі – реалізується на основі власного для кожного із типів НМ математичного та методологічного забезпечення. При першій реалізації цього етапу із допустимого діапазону обираються середні значення параметрів моделі. У разі наступних реалізацій параметри уточнюються з позицій збільшення точності моделювання.
4. Проведення навчання НМ – реалізується на основі методики навчання даного типу моделі.
5. Перевірка точності моделі на контрольних даних. У разі незадовільної точності потрібно повернутись до третього етапу та уточнити параметри моделі.
6. Перевірка точності моделі на тестових даних. Якщо точність незадовільна, то необхідно уточнити постановку задачі (змінити номенклатуру та методику обробки вхідних параметрів, перевірити коректність навчальних, контрольних та тестових прикладів) та повернутись до третього етапу. Якщо після декількох спроб досягти заданої точності не вдається, то варто відмовитись від використання цього типу моделі.
7. Вербалізація моделі – реалізується тільки для БШП.

Задача виявлення скриптових комп'ютерних вірусів

Будемо вважати, що НМ застосовується в антивірусному сканері з метою розпізнавання скриптових вірусів та троянів на основі аналізу програмного коду піддослідних файлів. У цьому разі номенклатура вхідних параметрів НМ повинна відображати здатність скриптових вірусів до саморозповсюдження та характерні спільні ознаки скриптових вірусів та троянів. У проведених дослідженнях акцентувалося на розпізнаванні поштових скриптових вірусів та макровірусів, призначених для зараження документів MS Office і написаних спорідненими мовами VBA та VBScript, які дають змогу працювати з файловою системою, встановлювати мережеві з'єднання, маніпулювати процесами і потоками, здійснювати виклик функцій API ОС та запускати зовнішні програми. Засобами розповсюдження макровірусу в такому разі можуть бути об'єкти відповідних бібліотек, функції API ОС та програмні додатки. Крім того, вхідні параметрів НМ повинні враховувати характерні ознаки скриптових вірусів, які можливо розділити на групи: автоматизації запуску, ігнорування помилок, маскування та деструктивних функцій. Приблизний перелік ознак наведений в табл. 2. Безпосередньо вхідними параметрами будуть фрагменти коду (назви функцій, параметрів, об'єктів, бібліотек, методів та властивостей об'єктів), що відповідають ознакам скриптових вірусів та макровірусів. Вхідні параметри можуть мати два значення: 1 – якщо ознака присутня та -1 в протилежному випадку. Кількість вхідних параметрів дорівнюватиме кількості відповідних фрагментів.

Визначаючи оптимальний вид архітектури НМ, акцентувалась вимога забезпечення максимальної достовірності класифікації та забезпечення максимального обсягу пам'яті мережі. Термін навчання мережі може бути достатньо тривалим, а кількість комбінацій вхідних параметрів, які характеризують скриптовий вірус чи троян, принципово обмежена. Для збільшення гнучкості системи розпізнавання визначено, що вихід НМ повинен містити ймовірнісну оцінку класифікації скрипта. Вербалізація НМ не потрібна. Відповідно до вказаних вимог та розробленої методики оптимальним видом архітектури є БШП.

Навчальну та тестову вибірки було сформовано на основі аналізу сигнатур вірусів, що входять до складу баз даних антивірусних програм. Крім того, як до навчальної, так і до тестової вибірки входять скрипти без ознак вірусів (безпечні скрипти). Статистичним матеріалом були використані сигнатури скриптових вірусів, що входять до складу бази даних антивірусних пакетів та частково представлені в [2]. Вхідні параметри БШП були отримані шляхом аналізу цих сигнатур. Кількість вхідних параметрів – 105. Сформовано навчальну вибірку з 560 прикладів. Половина прикладів відповідала скриптовим вірусам, а інша половина – безпечним скриптам. . Для вхідних елементів вибрано лінійну функцію активації, а для схованих елементів – сигмоїдальну. Прийнято, що кількість схованих шарів нейронів дорівнює 1. Кількість схованих нейронів розраховувалась відповідно [2, 3, 4], вважаючи що кількість бібліотечних образів БШП повинна в 1,5–2 рази перевищувати обсяг навчальної вибірки, який своєю чергою в 10 разів повинен перевищувати розмірність вхідного сигналу. Вихідний шар БШП складається з одного елемента. Враховуючи вказані передумови, визначено, що оптимальна кількість схованих нейронів L=5. Кількість прикладів, яку може запам'ятати БШП, $1262<P<6310$.

Таблиця 2

Характерні спільні ознаки скриптових вірусів

Назва групи	Перелік ознак
Автоматизації запуску	Використання автомакросів
Ігнорування помилок	Використання операторів ігнорування помилок та переходу на певний рядок програмного коду після виникнення помилок
Маскування вірусу	Шифрування/дешифрування макросу, захист макросу паролем, відключення захисту від макровірусів, блокування та перевизначення кодів клавіш, зміна шрифту макросів, запис/зчитування інформації в буфер обміну, відключення редактору VBA, знищення панелі інструментів для роботи з макросами та шаблонами, ігнорування повідомлень програмного середовища, поліморфізм макровірусу, використання функцій, що порушують функціонування антивіруса, знищення або перейменування файлу та модулю з вірусом, знищення процедури з вірусом
Деструктивні функції	Форматування жорстких дисків, модифікація та знищення файлів, встановлення паролів на файли, встановлення мережевих з'єднань, доступ до поштових клієнтів

Після навчання БШП були пред'явлені 40 тестових прикладів, отриманих за допомогою зміни деяких параметрів навчальних прикладів. Всі тестові приклади розпізнані правильно. До того ж максимальна похибка виходу БШП не перевищувала 10 %. Можна вважати, що цей приклад підтверджує ефективність запропонованої методики оптимізації.

Висновки

Запропоновано підхід та розроблена багатокритеріальна методика оптимізації архітектури нейронної мережі, призначеної для розв’язання задач захисту інформації в комп’ютерних системах та мережах.

Основні перспективи подальших розвідок у цьому напрямку полягають у розробленні методики визначення вагових коефіцієнтів критеріїв оптимізації та вдосконаленні загальної методики оптимізації з метою врахування оптимізаційних обмежень та використання перспективних типів архітектур НМ.

1. Хорошко В. А. Методы и средства защиты информации / В. А. Хорошко, А. А. Чекатов. – К.: Юниор, 2003. – 504 с. 2. Терейковський І. Нейронні мережі в засобах захисту комп’ютерної інформації / І. Терейковський. – К.: ПоліграфКонсалтинг. 2007. – 209 с. 3. Ежов А. А. Нейрокомпьютинг и его применения в экономике и бизнесе / А. А. Ежов, С. А. Шумский. – М. : МИФИ, 1998. – 224 с. 4. Хайкин С. Нейронные сети: полный курс, 2-е изд., испр. / Хайкин С.; пер. с англ. Н. Н. Куссуль. – М.: Вильямс, 2006. – 1104 с. 5. Шуклін Д. Є. Моделі семантичних нейронних мереж та їх застосування в системах штучного інтелекту: 05.13.23.. // Дис. …канд. техн. наук. – Харків, 2003. – 196 с.