

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Штучна імунна система для обробки та
класифікації даних

(тема)

Виконав:

студент II курсу, групи СПМ-22-5
Прудіус М.А.
(прізвище, ініціали)

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

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

Освітня програма Системне програмування
(повна назва освітньої програми)

Керівник: ст. викл. Фомічов О.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

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

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

Освітня програма _____ Системне програмування _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Прудіусу Микиті Андрійовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Штучна імунна система для обробки та класифікації даних _____

затверджена наказом по університету від “ 01 ” квітня 2024 р. № 257 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 15 червня 2024 р.

3. Вхідні дані до роботи _____

штучна імунна система

обробка даних

класифікація даних

розпізнавання об'єктів на зображеннях

радіально-базисна мережа

4. Перелік питань, що потрібно опрацювати у роботі _____

Аналіз існуючих методів та моделей

Огляд штучних імунних систем

Розробка алгоритму ідентифікації

Програмні засоби ідентифікації об'єктів на зображеннях з використанням ШС

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 13 слайдів

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання та аналіз літератури	01.04.2024 – 06.04.2024	
2	Огляд існуючих рішень та методів	07.04.2024 – 12.04.2024	
3	Розробка моделі	13.04.2024 – 18.04.2024	
4	Вибір програмних засобів	19.04.2024 – 25.04.2024	
5	Програмна реалізація	26.04.2024 – 02.05.2024	
6	Аналіз отриманих результатів	03.05.2024 – 16.05.2024	
7	Оформлення записки	17.05.2024 – 14.06.2024	
8	Представлення роботи в ЕК	15.06.2024 – 15.06.2024	

Дата видачі завдання 01 квітня 2024 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

ст. викл. Фомічов О.О.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 60 с., 24 рис., 1 табл., 2 дод., 14 джерел.

ШТУЧНА ІМУННА СИСТЕМА, РАДІАЛЬНО-БАЗИСНА МЕРЕЖА, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, РОЗПІЗНАВАННЯ.

Метою кваліфікаційної роботи є аналіз моделей та методів обробки та класифікації даних з використанням штучних імунних систем.

У ході виконання кваліфікаційної роботи проведено аналіз існуючих моделей та методів обробки та класифікації даних штучними імунними системами. Досліджені існуючі методи розпізнавання символів на зображеннях; проведено огляд алгоритмів штучних імунних систем; розроблена модель деревовидної штучної імунної системи для розпізнавання символів. Отримані результати показали доцільність використання розробленої моделі.

ABSTRACT

Master's thesis: 60 pages, 24 figures, 1 tables, 2 appendices, 14 sources.

ARTIFICIAL IMMUNE SYSTEM, RADIAL-BASIC NETWORK,
SOFTWARE, RECOGNITION.

The major goal of this thesis the analysis of models and methods of data processing and classification using artificial immune systems.

In order to the qualification work, an analysis of existing models and methods of data processing and classification by artificial immune systems was carried out. The existing methods of character recognition on images were studied; a review of algorithms of artificial immune systems was carried out; developed a model of a tree-like artificial immune system for character recognition. The obtained results showed the expediency of using the developed model.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ.....	12
1.1 Аналіз завдання ідентифікації	12
1.2 Огляд штучних нейронних мереж.....	14
1.3 Основи штучних імунних систем.....	16
1.4 Постановка завдання проектування	18
2 ОГЛЯД ШТУЧНИХ ІМУННИХ СИСТЕМ.....	19
2.1 Основні поняття в штучних імунних системах.....	19
2.2 Способи подання антитіл і антигенів.....	20
2.3 Аналіз існуючих моделей ШІС.....	21
2.3.1 Модель клонального відбору	21
2.3.2 Модель імунної мережі	21
2.3.3 Моделі негативного і позитивного відбору.....	21
2.4 Аналіз обчислювальної моделі клонального відбору	22
3 РОЗРОБКА АЛГОРИТМУ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ ЗАЛЕЖНОСТЕЙ НА ОСНОВІ ШТУЧНИХ ІМУННИХ СИСТЕМ.....	25
3.1 Постановка завдання ідентифікації.....	25
3.2 Алгоритм ідентифікації нелінійних залежностей.....	26
3.3 Алгоритм навчання нейронної мережі з використанням штучних імунних систем	28
4 РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ	41
4.1 Вибір мови програмування	41
4.2 Розробка інтерфейсу користувача.....	42
4.3 Аналіз компонентів програми.....	44
4.4 Функціональна верифікація розроблених програмних засобів.....	46

ВИСНОВКИ.....	49
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	50
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	52
ДОДАТОК Б Апробація	60

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ
І ТЕРМІНІВ

ОС – операційна система

ПЗ – програмні засоби

РБФ – радіально-базисна функція

ШІС – штучна імунна система

ШНМ – штучна нейронна мережа

ВСТУП

Завдання моделювання нелінійних об'єктів виникали і виникають в різних сферах людської діяльності. З метою вирішення цих завдань паралельно і незалежно один від одного розроблялися і розвивалися методи ідентифікації. Складні системи характеризуються великим числом входів-виходів і елементів, зв'язку між елементами носять різнотипний, нелінійний характер. Частина інформації про систему представлена в якісному вигляді. Функціонування системи відбувається в умовах нечіткості і невизначеності, яку вносить людський фактор.

У цьому випадку, як правило, отримання закону розподілу параметрів, що впливають на систему, стає важкою, часто нерозв'язною за обмежений час завданням. Традиційні засоби (імовірнісний підхід на основі апарату математичної статистики, імітаційне моделювання) не дозволяють будувати моделі таких систем в умовах обмеженості тимчасових, матеріальних і трудових ресурсів.

Спектр таких завдань постійно розширяється: це управління виробничими системами, розпізнавання сигналів і образів, класифікація та багато інших. Тому в останні роки спостерігається підвищення наукового і практичного інтересу до методів інтелектуальної обробки інформації. До них відносяться: штучні нейронні мережі, генетичні алгоритми, моделі на основі нечіткої логіки. У даній роботі розглядається рішення задачі ідентифікації нелінійної залежності з використанням нейронних мереж, яких навчають з використанням штучних імунних систем. В даний час дослідження нейронних мереж вважається одним з найбільш перспективних напрямків в області штучного інтелекту. Нейронні мережі є узагальненням ідеї перцептрона Розенблата.

В основу їх функціонування втілені принципи моделювання роботи людського мозку і вищих хребетних тварин. З формальної точки зору

нейронна мережа представляє собою універсальну модель-аппроксіатор у вигляді графа. Моделюючи реальний об'єкт, такий граф здатний шляхом так званого навчання підвищувати свою адекватність цього об'єкта за рахунок модифікації ваг межелементних зв'язків. Саме наявність формальних методик навчання (за умови існування навчальної вибірки у вигляді експериментальних пар «входи-вихід») є головною перевагою нейронних мереж.

Серед різноманітних методів навчання в даний час найбільш популярним є алгоритм зворотного поширення помилки, в основу якого покладено градієнтний метод пошуку оптимуму. Суть даного алгоритму полягає в зміні ваг міжнейронних зв'язків пропорційно міру їх впливу (першої похідної) на відмінність між модельними і експериментальними значеннями вихідної змінної.

Однак в якості засобу ідентифікації нелінійних об'єктів нейронні мережі характеризуються певними недоліками:

- ваги міжнейронних зв'язків навченої нейронної мережі не підлягають ясною і змістовною інтерпретації;

- на сьогоднішній день моделі нейронних мереж, що використовують якісні входи, розвинені недостатньо, щоб успішно моделювати відповідні об'єкти. Тому пропонується для цих цілей використання штучних імунних систем. З точки зору організації обробки даних імунна система - це високопараллельна структура. У ній реалізовані механізми навчання, пам'яті і асоціативного пошуку для вирішення завдань розпізнавання і класифікації.

Зокрема, імунна система здатна навчатися розпізнавання важливих структур (антигенних пептидів); запам'ятовування вже зустрічалися структур і використанню законів комбінаторики в рамках генних бібліотек для ефективною генерації детекторів структур (варіабельних ділянок молекул антитіл), які взаємодіють із зовнішніми антигенами і власними клітинами організму. При цьому реакція на антиген відбувається не тільки на рівні окремих розпізнають одиниць, а й на системному рівні шляхом взаємного

розпізнавання клонів лімфоцитів в реакціях антиген-антитіло. Таким чином, поведінка імунної системи визначається всією сукупністю локальних мережевих взаємодій. У даній роботі розглядається розробка програмних засобів ідентифікації нелінійних залежностей на основі штучних імунних систем.

Метою кваліфікаційної роботи є аналіз моделей та методів обробки та класифікації даних з використанням штучних імунних систем.

Об'єктом дослідження є функціонування штучних імунних систем.

Завдання:

- дослідження існуючих методів класифікації та розпізнавання символів на зображеннях;
- огляд алгоритмів штучних імунних систем;
- розробка моделі деревовидної штучної імунної системи для рішення подібних завдань.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ

1.1 Аналіз завдання ідентифікації

Ідентифікація, тобто, побудова математичної моделі за результатами спостережень, є важливим завданням, що виникає в техніці, економіці, політиці, медицині, біології і в інших областях. Математичне моделювання - це засіб вивчення реального об'єкта, процесу або системи шляхом їх заміни математичною моделлю, більш зручною для експериментального дослідження за допомогою ЕОМ.

Математична модель є наближеним представленням реальних об'єктів, процесів або систем, вираженим в математичних термінах і зберігає суттєві риси оригіналу. Математичні моделі в кількісній формі, за допомогою логіко-математичних конструкцій, описують основні властивості об'єкта, процесу або системи, його параметри, внутрішні і зовнішні зв'язки.

Математична модель описується (видається) математичними структурами, математичним апаратом (числа, букви, геометричні образи, відносини, алгебраїчні структури та т.д.). У загальному випадку математична модель реального об'єкта, процесу або системи представляється у вигляді системи функціоналів (формула 1.1):

$$\Phi(X, Y, Z, t) = 0, \quad (1.1)$$

де X – вектор вхідних змінних, $X = [x_1, x_2, \dots, x_n]^k$;

Y – вектор вихідних змінних, $Y = [y_1, y_2, \dots, y_n]^k$;

Z – вектор зовнішніх впливів, $Z = [z_1, z_2, \dots, z_n]^k$;

t – координата часу.

Побудова математичної моделі полягає у визначенні зв'язків між тими чи іншими процесами і явищами, створення математичного апарату, що дозволяє висловити кількісно і якісно зв'язок між тими чи іншими процесами і явищами, між важливими факторами фізичними величинами, і факторами, що впливають на кінцевий результат. Зазвичай їх виявляється настільки багато, що ввести в модель всю їх сукупність не вдається. При побудові математичної моделі перед дослідженням виникає завдання виявити і виключити з розгляду фактори, несуттєво впливають на кінцевий результат (математична модель зазвичай включає значно менше число факторів, ніж в реальній дійсності). На основі даних експерименту висуваються гіпотези про зв'язок між величинами, що виражають кінцевий результат, і факторами, введеними в математичну модель. Такий зв'язок часто виражається системами диференціальних рівнянь в приватних похідних (наприклад, в задачах механіки твердого тіла, рідини і газу, теорії фільтрації, теплопровідності, теорії електростатичного та електродинамічного полів). Кінцевою метою цього етапу є формулювання математичної задачі, рішення якої з необхідною точністю виражає результати, що цікавлять фахівця. Однією з основних операцій математичного моделювання є ідентифікація. Завдання ідентифікації можна уявити формально наступним чином.

Нехай $M=M(X,Y,A)$, де $A=\{a_i\}$, $a_i=(a_{i1}, a_{i2}, \dots, a_{ik})$ – вектор стану об'єкту.

Якщо вектор a_i залежить від деяких невідомих параметрів, то задача ідентифікації (моделі, параметрів моделі) полягає у визначенні за деякими додатковими умовами, наприклад, експериментальними даними, що характеризує стан, системи в деяких випадках. Ідентифікація -завдання побудови за результатами спостережень математичних моделей деякого типу, що адекватно описують поведінку системи. Якщо $S=\{S_1, S_2, \dots, S_n\}$ – деяка послідовність повідомлень, одержуваних від джерела інформації про систему, $M=\{m_1, m_2, \dots, m_z\}$ – послідовність моделей, що описують S , серед яких, можливо, міститься оптимальна (в якомусь сенсі) модель, то ідентифікація моделі M означає, що послідовність S дозволяє розрізнити (з

даного критерію адекватності) дві різні моделі в M . Послідовність повідомлень (даних) S назвемо інформативною, якщо вона дозволяє розрізняти різні моделі в M . Мета ідентифікації – побудова надійної, адекватної, ефективно функціонуючої гнучкої моделі на основі мінімального обсягу інформативною послідовності повідомлень. Найбільш часто використовувані методи ідентифікації систем (параметрів систем): метод найменших квадратів, метод максимальної правдоподібності, метод байєсовських оцінок, метод марковських ланцюгових оцінок, метод евристик, експертне оцінювання, ідентифікація на основі нечіткої логіки, штучних нейронних мереж і інші. У даній роботі будемо розглядати ідентифікацію нелінійних залежностей на основі штучних нейронних мереж.

1.2 Огляд штучних нейронних мереж

Штучні нейронні мережі (ШНМ) – математичні моделі, а також їх програмні або апаратні реалізації, побудовані за принципом організації та функціонування біологічних нейронних мереж – мереж нервових клітин живого організму. Для вирішення поставленого завдання використовувати будемо багат шарові нейронні мережі з прямим розповсюдженням сигналу. В ході навчання така система примусово навчається за допомогою прикладів «стимул-реакція».

З точки зору кібернетики, є одним з видів кібернетичного експерименту. Між входами і еталонними виходами (стимул-реакція) може існувати деяка залежність, але вона не відома. Відома тільки кінцева сукупність прецедентів – пар «стимул-реакція», звана навчальної вибіркою. На основі цих даних потрібно відновити залежність (побудувати модель відносин стимул-реакція, придатних для прогнозування), тобто побудувати алгоритм, здатний для будь-якого об'єкта видати досить точну відповідь. Для вимірювання точності відповідей, так само як і в навчанні на прикладах, може вводиться функціонал якості.

Навчання нейронної мережі являє собою задачу глобальної оптимізації. Знаходження глобального екстремуму є складною, а часто нерозв'язною завданням. Для цих цілей можуть використовуватися класичні градієнтні методи оптимізації. Недоліком цих методів є необхідність обчислення градієнта функції, залежність від початкового наближення і, як наслідок, велика ймовірність «застрягання» алгоритму в локальному екстремуму. Широко поширена оптимізація на основі випадкового пошуку, коли градієнт функції замінюється випадковим вектором.

Недоліками даного методу є низька швидкість збіжності, а також невизначеність у виборі умови зупинки. При цьому зі збільшенням розмірності вектора змінних сильно збільшується розмір області пошуку, настає «прокляття розмірності».

Переваги штучних імунних систем при вирішенні завдання навчання нейронної мережі (рисунок 1.1): відсутні значні математичні вимоги до видів цільових функцій і обмежень; відсутня залежність від початкового наближення; дозволяють ефективно відшукувати глобальний оптимум, коли поверхня оптимізується функції не має властивість опуклості; дозволяють виконувати мультимодальне оптимізацію.

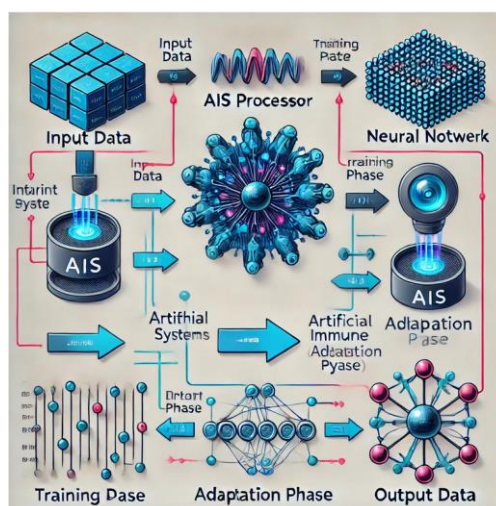


Рисунок 1.1 – Навчання нейронної мережі з використанням штучних імунних систем

1.3 Основи штучних імунних систем

Ідея створення штучних імунних систем з'явилася в результаті процесів біологічного імунітету, який захищає організм від хвороботворних бактерій і вірусів, виявляючи і знищуючи їх.

Біологічна імунна система являє собою складну адаптивну структуру, що складається з різних органів і компонентів, яка для захисту біологічного організму використовує різноманітні імунні механізми, такі як: виробництво імунокомпетентних клітин; їх навчання та відбір; виявлення шкідливих бактерій і вірусів; знищення виявлених вірусів; механізми адаптації, механізми пам'яті і т.д.

В результаті проведеного аналізу біологічної імунної системи був зроблений висновок, що дана система є надійним механізмом виявлення аномалій у вигляді бактерій і вірусів. Така система характеризується здатністю до класифікації об'єктів різного класу, а також наявністю механізмів боротьби з виявленими інфекціями.

Завдяки своїм особливостям і характеристикам, імунна система представляє великий інтерес в області обробки масивів даних, захисту інформації, завдань розпізнавання, ідентифікації і т.д.

Перераховані характеристики і можливості доводять перспективність використання основних концепцій імунітету у вирішенні складних комп'ютерних завдань. Оскільки механізм роботи природного імунної системи до цих пір повністю не вивчений і існує кілька різних теорій, що пояснюють принципи її роботи. Штучні реалізації цих теорій називають моделями ІВС.

Сучасні методи і моделі штучних імунних систем (ШІС) представлені трьома основними моделями: негативного відбору, клональної селекції та імунної мережі.

Успіх застосування тієї чи іншої моделі безпосередньо залежить від виду детекторів, застосовуваних у моделі ШІС, тому багато сучасних

дослідження в області ШІС приділяють увагу розвитку нових і узагальнення вже існуючих видів детекторів. Система імунітету викликає великий інтерес внаслідок її важливої ролі в підтримці цілісності організму. Властивості імунної системи служать чудовим прикладом локальних адаптивних процесів, що реалізують ефективні глобальні реакції. Для пояснення механізмів імунітету існує кілька різних теорій (які іноді суперечать один одному).

Опубліковано низку імітаційних моделей, що описують реакції різних компонентів імунної захисту. Відбувається розширення сфери застосування нових методів вирішення прикладних завдань, заснованих на принципах імунології. Ці методи мають різні назви: штучні імунні системи; системи, засновані на принципах імунітету; імунологічні обчислення, і т.д.

Сфера їх застосування включає наступні області (але не обмежується ними):

- методи обчислень;
- когнітивні моделі;
- штучні імунні системи для розпізнавання образів;
- методи виявлення аномалій і несправностей;
- мультиагентні системи;
- моделі самоорганізації;
- моделі колективного інтелекту;
- системи пошуку і оптимізації;
- моделі автономних розподілених систем;
- моделі штучного життя;
- системи комп'ютерної та інтернет-безпеки;
- моделі навчаються систем;
- методи добування інформації;
- штучні імунні системи для виявлення підробок;
- методи обробки сигналів і зображень.

1.4 Постановка завдання проектування

Сучасні дослідження спрямовані на створення нових більш ефективних і швидкодіючі алгоритмів для вирішення різних завдань, які б мали успіх в застосуванні для складних систем. Передумовою для цього є розвиток штучних імунних систем і нейромережевих технологій, які мають біологічні основи. Здатність таких систем до навчання і узагальнення результатів дозволяє створювати на базі їх інтелектуальні системи для вирішення завдань ідентифікації нелінійних залежностей.

Розроблені алгоритми і програмні засоби дозволяють підвищити ефективність вирішення завдань ідентифікації. Для досягнення поставленої задачі необхідно вирішити такі основні завдання:

- провести аналіз задачі ідентифікації;
- розробити структуру і алгоритми функціонування нейронної мережі для вирішення задачі ідентифікації нелінійних залежностей;
- розробити алгоритми навчання нейронної мережі з використанням ШС;
- розробити програмне забезпечення для розв'язання задачі ідентифікації з метою практичної перевірки розроблених алгоритмів;
- функціональна верифікація розроблених програмних засобів. Як програмне середовище розробки пропонується використання Eclipse. Мова Java має ряд переваг в порівнянні з іншими мовами: кроссплатформенність, велика різноманітність типів даних, гнучкість при роботі з пам'яттю.

2 ОГЛЯД ШТУЧНИХ ІМУННИХ СИСТЕМ

2.1 Основні поняття в штучних імунних системах

Штучні імунні системи – це адаптивні системи, в основі яких лежить теоретична імунологія та спостережувані принципи, функції і моделі імунітету, що застосовуються для вирішення завдань в різних проблемних областях. Основні поняття:

- антиген – будь-яка молекула, яку можуть специфічно розпізнавати лімфоцити (патоген, який потрапив в організм), – завдання, яке необхідно вирішити;

- антитіло – рецептор на поверхні В-лімфоцитів, здатний розпізнати антиген;

- безліч можливих варіантів розв'язання задачі.

Афінність – сила зв'язку між антитілом і антигеном, - характеристика, кількісно описує силу взаємодії антигену і антитіла. Для побудови системи, перш за все, необхідно визначитися з проблемною областю:

- уявлення: необхідно визначитися, яким чином будуть представлені елементи системи – антитіла, антигени;

В ШС спосіб представлення елементів називається простором форм (shape-space);

- обчислення афінності: необхідно визначитися з функцією афінності для вимірювання взаємодії елементів між собою.

Існують різні способи обчислення афінності, які частково залежать від обраного уявлення компонентів системи, тобто від простору форм;

- імунні алгоритми: необхідно визначитися з використовуваними імунними алгоритмами, які керують поведінкою (динамікою) системи. Імунні алгоритми засновані на обчислювальних моделях ШС.

2.2 Способи подання антитіл і антигенів

Набір параметрів, що характеризують антитіло або антиген, називається його узагальненою формою (*generalized shape*). Ці параметри можуть включати різні атрибути, які є важливими для розпізнавання антигену антитілом. Математично узагальнена форма антитіла або антигену може бути представлена у вигляді набору координат: $Ab = \langle Ab_1, Ab_2, \dots, Ab_L \rangle$, $Ag = \langle Ag_1, Ag_2, \dots, Ag_L \rangle$, які можуть розглядатися як точка в L -вимірному матеріальному просторі форм.

Простір форм – це L -мірний простір, в якому вимірювання відповідають набору атрибутів, залучених у взаємодію антитіл і антигенів, що дозволяє представляти антитіла і антигени точками в цьому просторі. Як правило, антитіло і антиген мають одну і ту ж довжину L .

Афінність – характеристика, кількісно описує силу взаємодії антигену і антитіла. У термінології простору форми є відстань між антитілом і антигеном, ігноруючи природу комплементарності між ними. Простір форм і афінність представлені на рисунку 2.1.

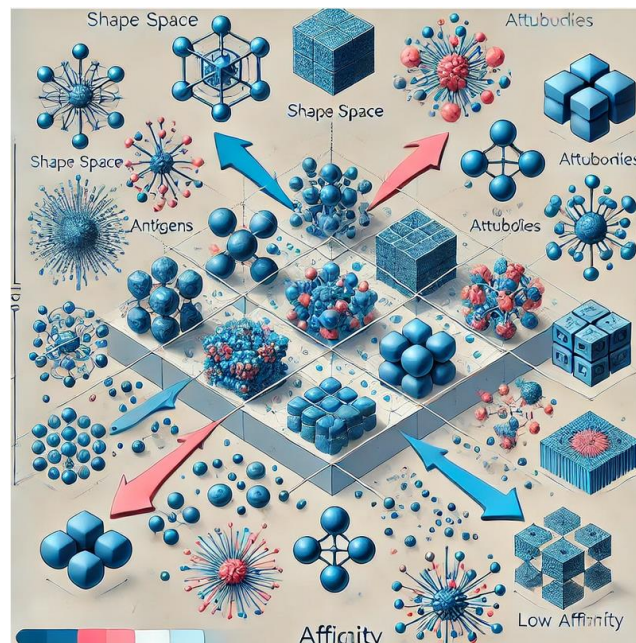


Рисунок 2.1 – Простір форм та афінність

2.3 Аналіз існуючих моделей ШС

2.3.1 Модель клонального відбору

Клональний відбір є найбільш поширеною обчислювальною моделлю. Її особливості полягають в тому, що після генерації випадкового позитивного детектора він піддається клонуванню, а потім здійснюється мутація кожного з клонів. Таким чином, з популяції клонів можна визначити найбільш точно відповідає антигену детектор, який і замінить батьківський детектор в загальній популяції. Цей підхід дозволяє прискорити процес збіжності методу, при цьому розмір популяції істотно не збільшується, що призводить до економії обчислювальних ресурсів.

2.3.2 Модель імунної мережі

Імунна система являє собою регульовану мережу молекул, які розпізнають один одного навіть за відсутності антигену. Різні антитіла один від одного не ізольовані, а підтримують зв'язок шляхом взаємодій між своїми рецепторами. Розпізнавання антигену антитілом призводить до активації мережі і збільшення кількості антитіл. Розпізнаючи один одного, антитіла утворюють області з'єднаних клітин, що веде до стиснення мережі.

2.3.3 Моделі негативного і позитивного відбору

Модель негативного відбору є однією з найбільш широко відомих моделей ІВС. Її принцип роботи полягає в генерації детекторів, відповідних аномальної поведінки досліджуваного об'єкта. Метод негативного відбору складається з двох фаз: навчання та класифікації.

Під час навчання на вхід моделі надходять «свої» антигени (self-

antigen). При цьому в ІВС виробляються випадкові клітини, які стають детекторами лише в тому випадку, якщо вони не реагують ні на один з антигенів, а також і на вже існуючі імунні клітини (антитіла). Таким чином, отримана під час навчання популяція клітин в значній мірі визначає простір передбачуваних "чужих" клітин. У процесі розпізнавання клітина вважається патогенної ("чужий"), якщо її розпізнає хоч один з існуючих детекторів. Метод неодноразово піддавався критиці в силу низької точності класифікації і великих тимчасових витрат на роботу.

Однак деякі його модифікації, зокрема V-детектор, показали хороші результати і визначили нові шляхи розвитку моделі. Позитивний відбір заснований на схемі роботи моделі негативного відбору, за винятком того, що детектором вважають випадкову клітку, що реагує хоча б на один антиген з навчальної вибірки.

2.4 Аналіз обчислювальної моделі клонального відбору

Клональний відбір (рисунок 2.2) – це назва теорії, яка пояснює, як адаптивна імунна система справляється з патогенними мікроорганізмами. Властивості моделі клонального відбору:

- антитіло розпізнає чужорідний антиген з певною афінністю; - дане антитіло відбирається для клонування і виробляє безліч клонів антитіл, які потім піддаються процесу мутації;

- процес мутації призводить до того, що нові створені антитіла мають кращу афінністю до даного антигену, ніж вихідне відібране антитіло; – активовані антитіла з високою афінністю зберігаються в пам'яті.

Характеристики моделі клонального відбору:

- розпізнавання антитілами специфічних антигенів; – відбір антитіл з високою афінністю і їх клонування;

- клонування антитіл пропорційно афінності даного антитіла з антигеном;

- тобто, чим вище аффінність, тим більша кількість клонів проводиться;
- генерація випадкових генетичних змін, шляхом внесення мутацій, для підтримки різноманітності в популяції антитіл;
- мутація, якій піддаються антитіла назад-пропорційна афінності антитіл до антигенів;
- видалення новостворених антитіл, якщо вони мають низьку афінність;
- наявність клітин пам'яті для збереження знайдених рішень.

Відбір антитіл для клонування заснований на афінності даного антитіла до антигену. Для здійснення відбору використовуються різні механізми: елітарний добір, колесо рулетки, турнірний відбір.

Елітарний добір – відбираються антитіла з кращою афінністю в популяції. Колесо рулетки – поля колеса рулетки відповідають нормованим значенням афінності. Показник рулетки після зупинки колеса визначає вбрання антитіло.

Турнірний відбір – з популяції випадковим чином відбирається n антитіл, які беруть участь в турнірі. Переможцем є антитіло з кращою афінністю.

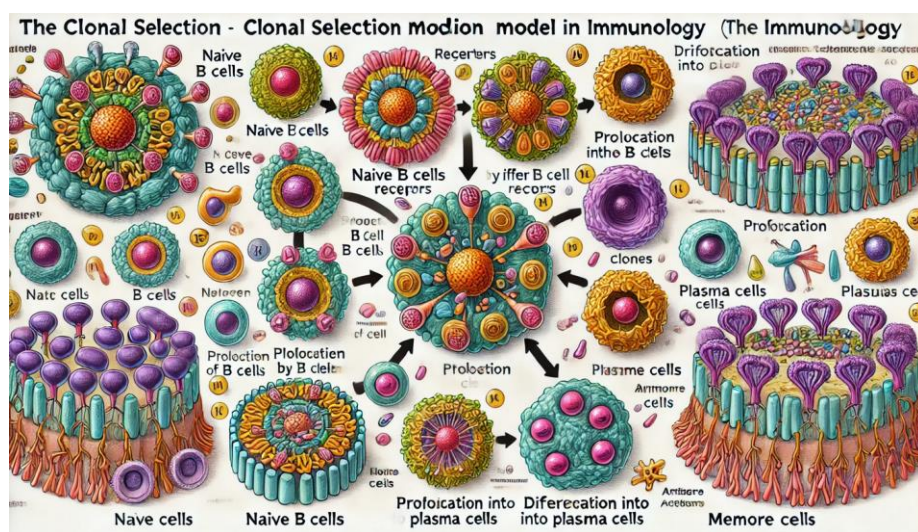


Рисунок 2.2 – Модель клонального відбору

Мутація антитіл – внесення в гени антитіл випадкових змін. Мутація має дві важливі функції:

- внесення різноманітності в популяцію антитіл;
- поліпшення афінності відібраних антитіл. Залежно від обраного простору форм розрізняють 4 види мутації:
 - мутація в хеммінговом просторі форм;
 - мутація в целочисленном просторі форм; – мутація в символному просторі форм;
 - мутація в матеріальному просторі форм.

По виду внесених змін розрізняють:

- одноточковий мутація;
- мультиточечная мутація; – інверсійні мутація;
- мутація r-безперервних біт;
- індуктивна мутація;
- рівномірна мутація.

3 РОЗРОБКА АЛГОРИТМУ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ ЗАЛЕЖНОСТЕЙ НА ОСНОВІ ШТУЧНИХ ІМУННИХ СИСТЕМ

3.1 Постановка завдання ідентифікації

У даній роботі розглядається наступне завдання. Виконати нормалізацію кольору: спроектувати нейронну мережу для виконання нелінійного відображення зображення, освітленого невідомим джерелом світла, щодо зображення, освітленого іншим джерелом світла (еталонним). набір векторів кольорів зображення представимо в такий спосіб:

$$\Omega = \{(r, g, b) \in \mathbb{R}^3: 0 \leq r, g, b \leq 255\}$$

Розглянемо $C(p_{i,j}^k) \in \Omega$, $1 \leq i \leq M$, $1 \leq j \leq N$ – колірне значення пікселя p в позиції (i, j) зображення, освітлене джерелом освітлення I_k , $M \times N$ – розмір зображення.

Є два джерела освітлення: I_1 і I_2 . Зображення, освітлене джерелом світла I_2 є еталонним. Відображення $f(I_1, I_2): C(p_{i,j}^1) \rightarrow C(p_{i,j}^2)$ пікселей зображення можна представити наступним чином:

$$C(p_{i,j}^2) = A * C(p_{i,j}^1), 1 \leq i \leq M, 1 \leq j \leq N$$

таке відображення в навколишньому світі є нелінійним і елементи матриці A залежать від I_1 і I_2 . Завдання ідентифікації нелінійної залежності $f(I_1, I_2)$ вирішується за допомогою ІНС, навченою з використанням алгоритму клонального відбору. Подання вихідних даних виконується в два етапи.

Етап 1: визначаємо навчальну вибірку – це RGB вектори пікселів зображення $0 \leq r, g, b \leq 255$ для 8-бітного представлення. Її отримуємо шляхом отримання з картинки з еталонним освітленням матриці пікселів.

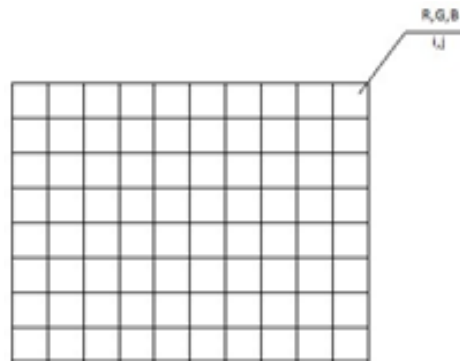


Рисунок 3.1 – Зображення, як матриця пікселів

Етап 2: виробляємо нормалізацію зображення. Для цього значення кожного кольору пікселів ділимо на їх максимальне значення. У підсумку маємо $r, g, b \in [0; 1]$.

3.2 Алгоритм ідентифікації нелінійних залежностей

Для ідентифікації нелінійної залежності, представлені у вигляді (рисунок 3.2). Для її вирішення використовується штучна нейронна мережа.

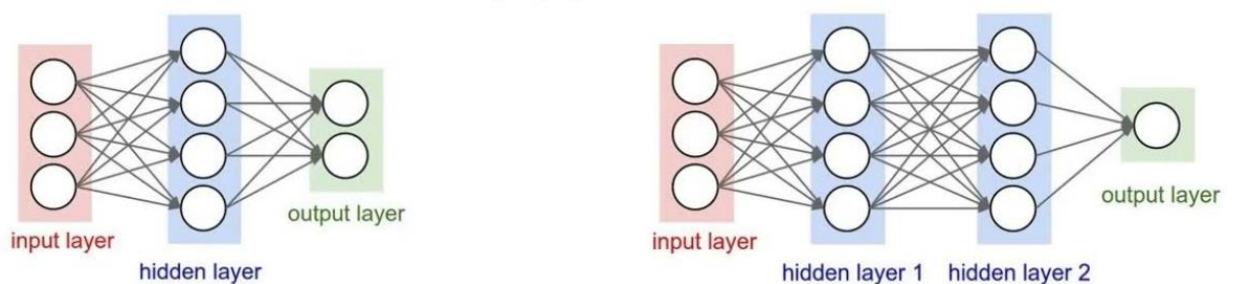


Рисунок 3.2 – Нейронна мережа прямого поширення

Архітектуру нейронної мережі (рисунок 3.3) можна представити в наступному вигляді:

- вхідний шар (3 нейрони), лінійна функція активації (формула 3.1);
- прихований шар (100-200 нейронів), сигмоїдальна функція активації (формула 3.2);

$$O_z = X_1 * W_{11} + X_2 * W_{12} + X_3 * W_{13}; \quad (3.1)$$

$$Out_z = 1 / (1 - \exp(-O_z)); \quad (3.2)$$

- вихідний шар (3 нейрони $C2(p_i, j)$), лінійна функція активації (формулы 3.3, 3.4, 3.5).

$$R = Out_1 * k_{11} + Out_2 * k_{21} + \dots + Out_z * k_{n1}, \quad (3.3)$$

$$G = Out_1 * k_{12} + Out_2 * k_{22} + \dots + Out_z * k_{n2}, \quad (3.4)$$

$$B = Out_1 * k_{13} + Out_2 * k_{23} + \dots + Out_z * k_{n3}. \quad (3.5)$$

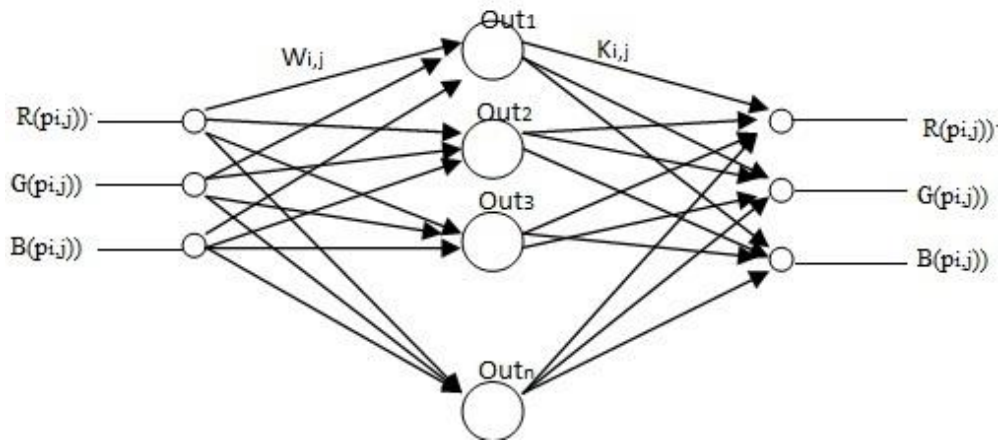


Рисунок 3.3 – Структура нейромережі

Для навчання нейронної мережі необхідно мінімізувати відхилення реального виходу мережі від еталонного. Знаходимо мінімізацію за формулою 3.6 На кожній ітерації оновлюються ваги і зміщення нейронів 3 шарів.

$$E_{(w,k)} = \frac{1}{2} (\sum (Y_{m,p} - d_{m,p})^2) \quad (3.6)$$

3.3 Алгоритм навчання нейронної мережі з використанням штучних імунних систем

Для вирішення завдання навчання штучної нейронної мережі прийнято рішення використовувати алгоритм клонального відбору, що дозволяє знаходити глобальний екстремум функції. Знаходження глобального екстремуму в багатьох випадках є досить складним завданням.

Для цих цілей можуть використовуватися класичні градієнтні методи оптимізації. Недоліком цих методів є необхідність обчислення градієнта функції, залежність від початкового наближення і, як наслідок, велика ймовірність «застрягання» алгоритму в локальному екстремуму. Широко поширена оптимізація на основі випадкового пошуку, коли градієнт функції замінюється випадковим вектором.

Недоліками даного методу є низька швидкість збіжності, а також невизначеність у виборі умови зупинки. При цьому зі збільшенням розмірності вектора змінних сильно збільшується розмір області пошуку, настає «прокляття розмірності». Генетичні алгоритми і штучні імунні системи як частина парадигми обчислювального інтелекту дозволяють вирішувати багато проблем класичних методів оптимізації.

Переваги штучних імунних систем при вирішенні задач оптимізації:

- відсутні значні математичні вимоги до видів цільових функцій і обмежень;
- відсутня залежність від початкового наближення;
- дозволяють ефективно відшукувати глобальний оптимум, коли поверхня оптимізується функції не має властивість опуклості;
- дозволяють виконувати мультимодальне оптимізацію. Для вирішення завдань оптимізації нелінійних функцій використовуються імунні алгоритми, засновані на обчислювальній моделі клонального відбору.

Алгоритм клонального відбору є таку послідовність кроків.

1 крок: генерація популяції антитіл P .

2 крок: визначити n кращих антитіл популяції шляхом обчислення їх афінності (в якості опції афінності використовується цільова функція).

3 крок: клонувати кращі антитіла популяції задану кількість разів, сформувавши таким чином тимчасову популяцію клонів (C). Кратність клонування антитіла пропорційна його афінності.

4 крок: піддати популяцію клонів антитіл процесу мутації, пропорційно афінності антитіла до антигену. При цьому створюється популяція (C^*).

5 крок: вибрати з популяції C^* антитіла з поліпшеною в результаті мутації афінністю, замінити ними відповідні антитіла, від яких були зроблені клони.

6 крок: замінити d антитіл новими випадково згенерували для підтримки різноманітності популяції. Чим нижче афінність антитіл, тим більше шансів, що вони будуть замінені новими.

7 крок: перевірити критерій зупинки (задану кількість поколінь). Якщо критерій зупинки досягнуто, то результатом оптимізації буде антитіло з кращого по популяції афінністю. В іншому випадку повернутися до кроку 2.

В даному алгоритмі, клонування – отримання заданої кількості ідентичних копій вихідного антитіла. Клонування антитіл виконується пропорційно афінності даного антитіла з антигеном – тобто, чим краще афінність, тим більша кількість клонів проводиться.

Розрахунок кратності клонування (формула 3.7) виконується пропорційно афінності антитіла:

$$C(Ab_i) = \begin{cases} \text{Min}_C & \text{if } \text{Aff}(Ab_i) \leq (\text{Max_Aff} * 0.3) \\ \text{Max}_C & \text{if } \text{Aff}(Ab_i) \geq (\text{Max_Aff} * 0.7) \end{cases}, \quad (3.7)$$

де $\text{Aff}(Ab_i)$ – афінність антитела i ;

- Max_Aff , Min_Aff – макс. і мін. значення афінності на даному поколінні.

Клони антитіл піддаються процесу мутації. Мутація – генерація випадкових генетичних змін для підтримки різноманітності в популяції антитіл. Мутація, якій піддаються антитіла назад-пропорційна аффинности антитіл до антигенів.

Розрахунок ймовірності мутації клону (3.8) виконується обернено пропорційно аффинности вихідного антитіла:

$$\text{Mut_rate}(Ab_i) = \text{Max_mut_rate} \cdot \frac{(\text{Aff}(Ab_i) - \text{Max_Aff})}{(\text{Min_Aff} - \text{Max_Aff})}, \quad (3.8)$$

де $\text{Mut_rate}(Ab_i)$ – ймовірність мутації клону i ;

Max_mut_rate – максимальна ймовірність мутації.

Результатом роботи алгоритму клонального відбору є антитіло з кращого по популяції аффинностью, яке і являє собою рішення задачі.

3.4 Функції навчання РБМ

Радіально-базисні мережі (Radial Basis Function Nets, RBFN) були запропоновані для апроксимації функцій багатьох змінних. За допомогою радіально-базисних функцій можна як завгодно точно апроксимувати задану функцію. Як і багатошаровий перцептрон, радіально-базисна мережа (РБМ) є універсальним аппроксіматором.

Математичну основу РБ-мережі становить метод потенційних функцій, розроблений М.А.Айзерманом, Е.М.Бравержаном. і Л. І. Розоноером, що дозволяє представити деяку функцію у вигляді суперпозиції потенційних або базисних функцій f_a :

$$y(\mathbf{x}) = \sum_{i=1}^N a_i f_i(\mathbf{x}) = a^T f(\mathbf{x}), \quad (3.9)$$

де $a_i(t) = (a_{i1}, a_{i2}, \dots, a_{in})^T$ – вектор підлягають визначенню параметрів;

$f(x) = (f_1(x), f_2(x), \dots, f_N(x))^T$ – вектор базисних функцій.

У РБС в якості базисних вибираються деякі функції відстані між векторами

$$f_i(x) = f(\|x - c_i\|), \quad (3.10)$$

Вектори c_i є центрами базисних функцій. Функції $f_i(x)$ обираються негативними $\|x - c_i\|$. В якості міри сближення x и c_i обирається евклідова

метрика $\|x - c_i\| = \left(\sum_{j=1}^N (x_j - c_{ij})^2 \right)^{\frac{1}{2}}$, або манхеттенська $\|x - c_i\| = \sum_{j=1}^N |x_j - c_{ij}|$,

где $|x_j - c_{ij}| = (x_j - c_{ij}) \operatorname{sgn}(x_j - c_{ij})$,

$$\operatorname{sgn}(x_j - c_{ij}) = \begin{cases} 1, \text{если } (x_j - c_{ij}) > 0 \\ 0, \text{если } (x_j - c_{ij}) = 0. \\ -1, \text{если } (x_j - c_{ij}) < 0 \end{cases} \quad (3.11)$$

Радіально-базисні мережі мають багато спільного зі стохастичними мережами. Як і стохастичні мережі, РБМ володіють великою швидкістю навчання. Слід також зазначити, що при їх навчанні не виникає проблем з «застреванием» в локальних мінімумах. Однак у зв'язку з тим, що при виконанні безпосередньо класифікації проводяться досить складні обчислення, зростає час отримання результату.

3.4 Архітектура РБМ

Особливістю цих мереж є наявність радіально-симетричного шаблонного шару. Структура РБС відповідає мережі прямого поширення першого порядку (рисунок 2.1). Інформація про образи передається з вхідного шару на прихований, є шаблонною і містить нейронів. Кожен

нейрон шаблонного шару, отримуючи повну інформацію про вхідних сигналах обчислює функцію

$$f_i(x) = f((x - c_i)^T R^{-1}(x - c_i)), i = \overline{1, p}, \quad (3.12)$$

де x – вектор вхідних сигналів ($N \times 1$); c_i – вектор центрів ($N \times 1$); R – вагова матриця.

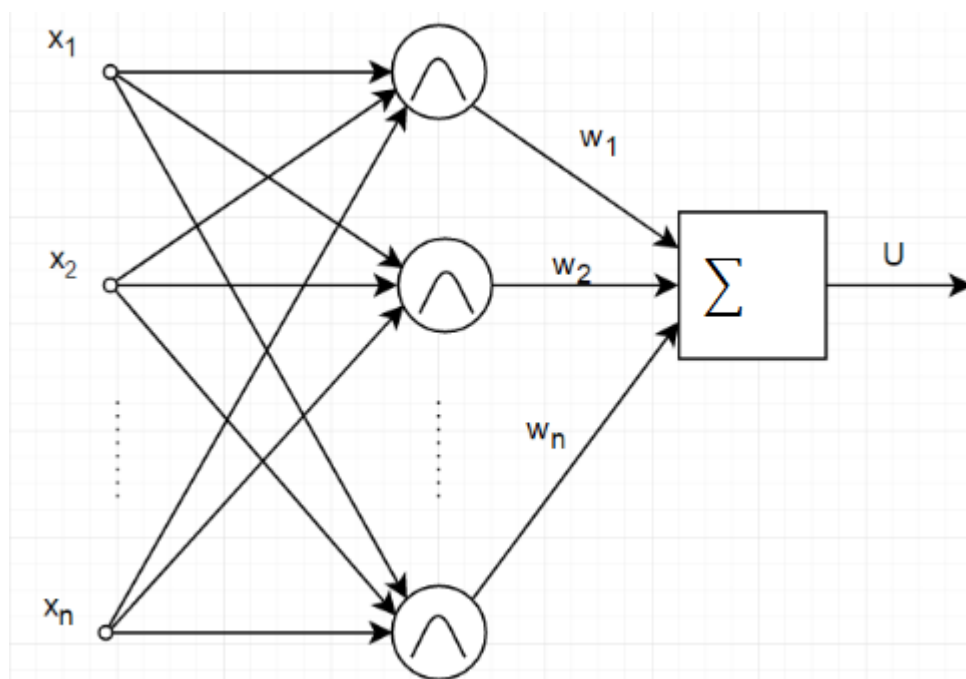


Рисунок 3.4 – Структура РБФ мережі

Як уже зазначалося, особливістю даних мереж є наявність радіально-симетричного шаблонного шару, в якому аналізується відстань $(x - c_i)^T R^{-1}(x - c_i)$ між вхідним вектором і центром, представленим у вигляді вектора у вхідному просторі. Вектор центрів визначається за навчальною вибіркою і зберігається в просторі ваг від вхідного шару до шару шаблонів.

На рисунку 3.5 представлений $i - \text{й}$ нейрон шаблонного шару РБФ. Обробку надходить на нього інформації умовно можна розділити на два етапи: на першому обчислюється відстань між пред'явленими чином i вектором центрів з урахуванням обраної метрики і норми матриці R , на другому яку пробразується нелінійної активаційної функцією. Подвійні стрілки на малюнку позначають векторні сигнали, а потрійні - матричний сигнал.

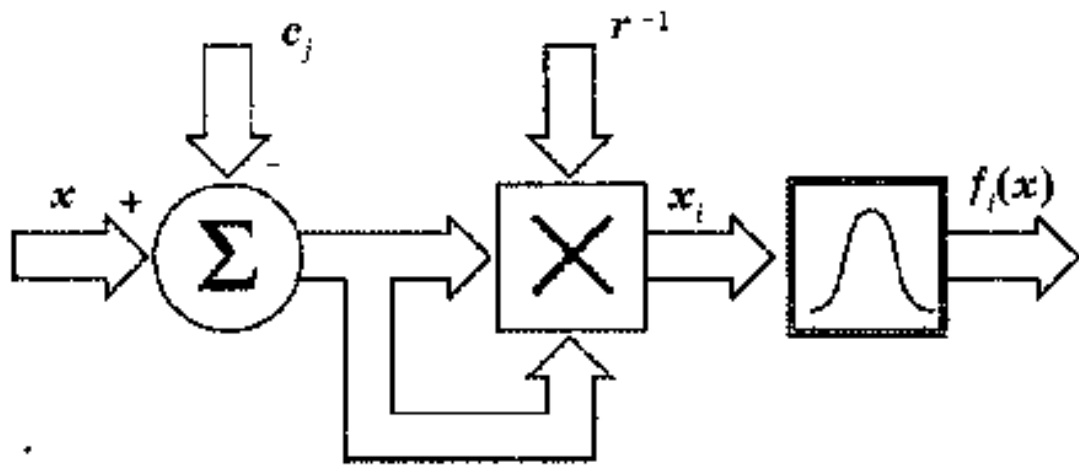


Рисунок 3.5 – Нейрон шаблонного шару РБФ

В якості функції перетворення $f(\cdot)$ найчастіше обираються наступні:

- гаусова функція (рисунок 3.6)

$$f(x) = \exp \left\{ -\frac{(x-c)^2}{2\sigma^2} \right\}, \quad (3.13)$$

- мультіквадратична функція (рисунок 3.7)

$$f(x) = \left[-\frac{(x-c)^2}{\sigma^2} + a^2 \right]^{\frac{1}{2}}; \quad (3.14)$$

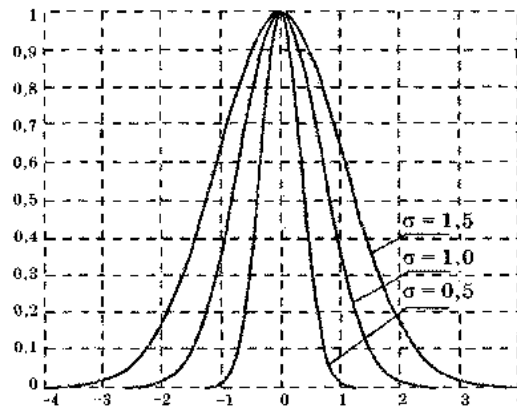


Рисунок 3.6 – Гаусова функція

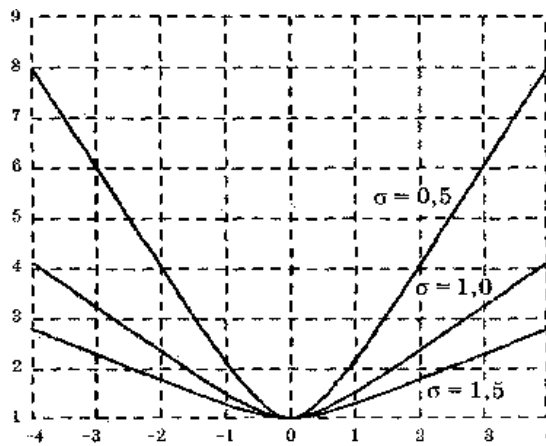


Рисунок 3.7 – Мультиквадратична функція

- зворотня мультиквадратична функція (рисунок 3.8)

$$f(x) = \left[-\frac{(x-c)^2}{\sigma^2} + a^2 \right]^{\frac{1}{2}} \quad (3.15)$$

- сплайн-функція (рисунок 3.9)

$$f(x) = x^2 \log(x); \quad (3.16)$$

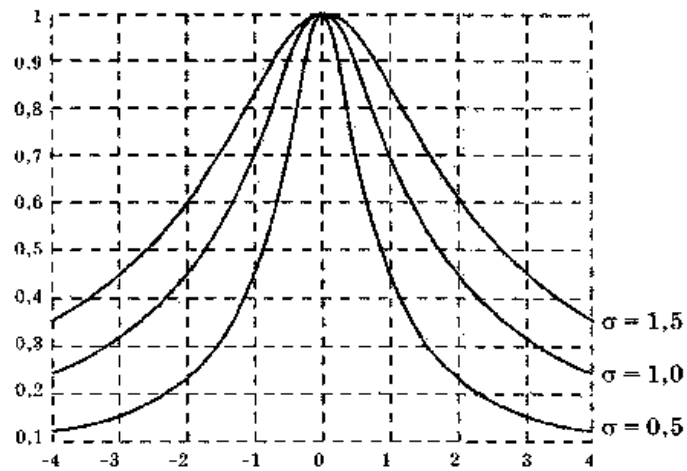


Рисунок 3.8 – Зворотня мультікватрична функція ($a = 1, c = 0$)

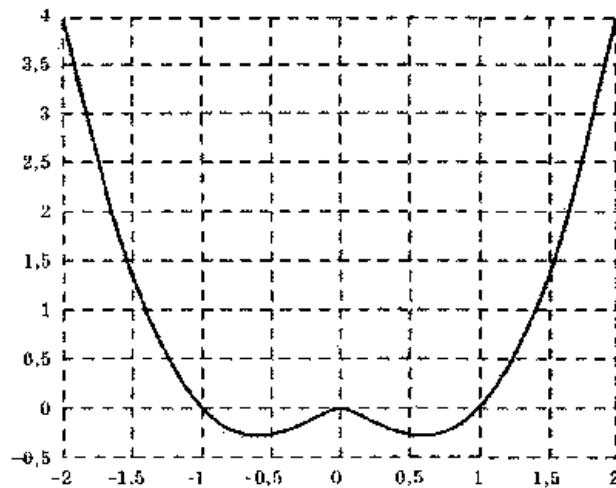


Рисунок 3.9 – Сплайн-функція Коши

$$f(x) = (1 + |x|)^{-1} \quad (3.17)$$

Слід зазначити, що зазначені функції допускають різні модифікації. Приклади таких модифікацій для гауссових функції $f(x) = \exp\{-P(x)\}$ наведені в таблиці 2.1

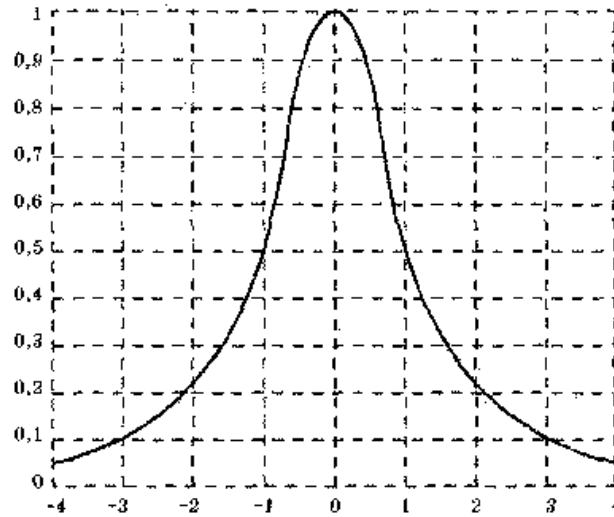


Рисунок 3.10 – Функція Коши

Таблиця 3.1 – Значення змінних

	$P_k(x)$
	$\lambda_k^2 \ x - c_k\ ^2 = \lambda_k^2 [(x_1 - c_{k1})^2 + (x_2 - c_{k2})^2]$
	$\lambda_k^2 \ D(x - c_k)\ ^2 = \lambda_k^2 [d_1^2 (x_1 - c_{k1})^2 + d_1^2 (x_2 - c_{k2})^2]$
	$\lambda_k^2 \ W^T(x - c_k)\ ^2 = \lambda_k^2 [(w_{11}(x_1 - c_{k1})^2 + w_{21}(x_2 - c_{k2}))^2 \times$ $\times (w_{12}(x_1 - c_{k1})^2 + w_{22}(x_2 - c_{k2}))^2]$
	$\ D_k(x - c_k)\ ^2 = d_{k1}^2 (x_1 - c_{k1})^2 + d_{k2}^2 (x_2 - c_{k2})^2$
	$\ D_k W^T(x - c_k)\ ^2 = [d_{k1}(w_{11}(x_1 - c_{k1})^2 + w_{21}(x_2 - c_{k2}))^2] +$ $+ [d_{k2}(w_{12}(x_1 - c_{k1})^2 + w_{22}(x_2 - c_{k2}))^2]$
	$\ D_k(x - c_k)\ ^2 = d_{k1}^2 (x_1 - c_{k1})^2 + d_{k2}^2 (x_2 - c_{k2})^2,$ где $d_{ki} = \begin{cases} d_{ki}^+, & x_i \geq c_{ki}; \\ d_{ki}^-, & x_i < c_{ki}. \end{cases}$

Таким чином, дана мережа є нейросетовою реалізацією конкретної функції, що апроксимується, при якій кожному висунутій образу відповідає свій нейрон шаблонного шару. Визначення вагових параметрів нейронів

вихідного шару здійснюється шляхом вирішення системи лінійних алгебраїчних рівнянь.

Показано, що online-навчання мережі шляхом рекуррентної настройки її параметрів є нестійким. Більш стійким є навчання в режимі offline, коли для знаходження параметрів мережі використовується критерій. Безсумнівно, важливим є питання вибору кількості нейронів прихованого шаблонного шару. Принципово може бути реалізована РБ-мережу з варійованим кількістю цих нейронів, що збільшується, наприклад, в разі, якщо досягається після навчання помилка апроксимації буде перевищувати допустиму. Такий підхід може бути застосований і в разі, якщо не проведено первинне розбиття образів на кластери.

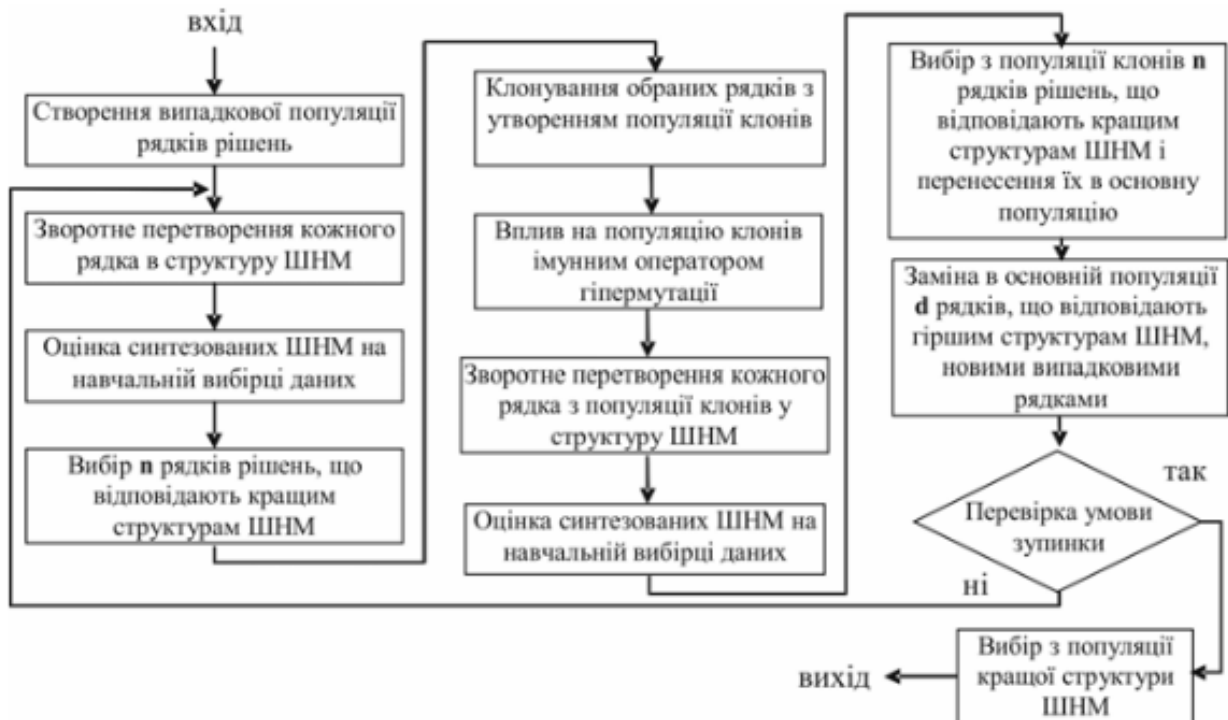


Рисунок 3.11 – Алгоритм синтезу РБФ мережі алгоритмом клонального відбору

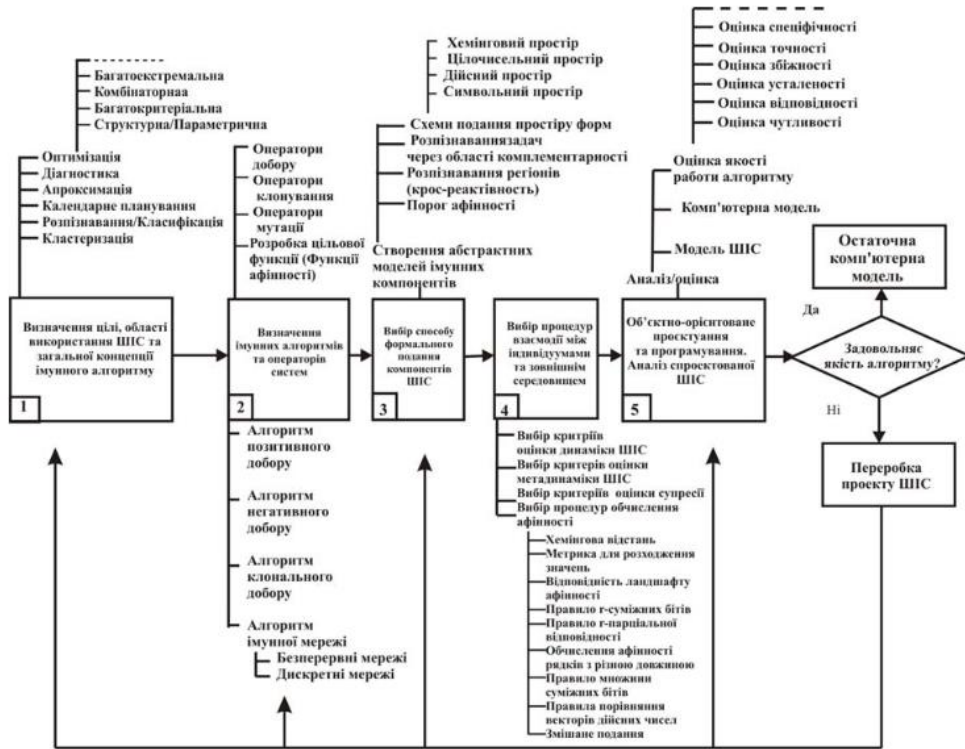


Рисунок 3.12 – Процес проектування й оцінки ефективності ШІС

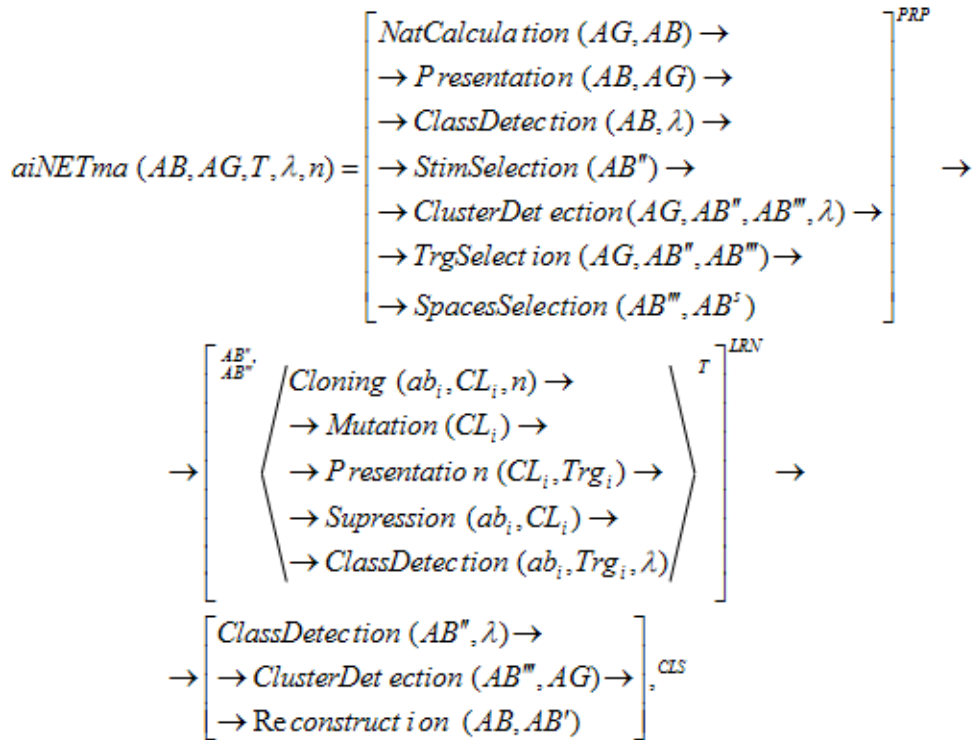


Рисунок 3.13 – Модель автоматичної класифікації aiNET

Основною особливістю запропонованої моделі автоматичної класифікації aiNETma є зміна поведінки стимулюючих антитіл, які взаємодіють з різними множинами об'єктів, що класифікуються в процесі імунного навчання. Відповідно до цього, під час роботи з об'єктами, які не можуть бути віднесені до жодного з початкових класів у процесі навчання, використовуються стимулюючі антитіла як початкові центри нових кластерів. У такому разі з цих стимулюючих антитіл формується множина цільових об'єктів, що підлягають кластеризації в процесі імунного навчання.

Представлено спосіб модифікації моделі імунної мережі для розв'язання задач кластеризації

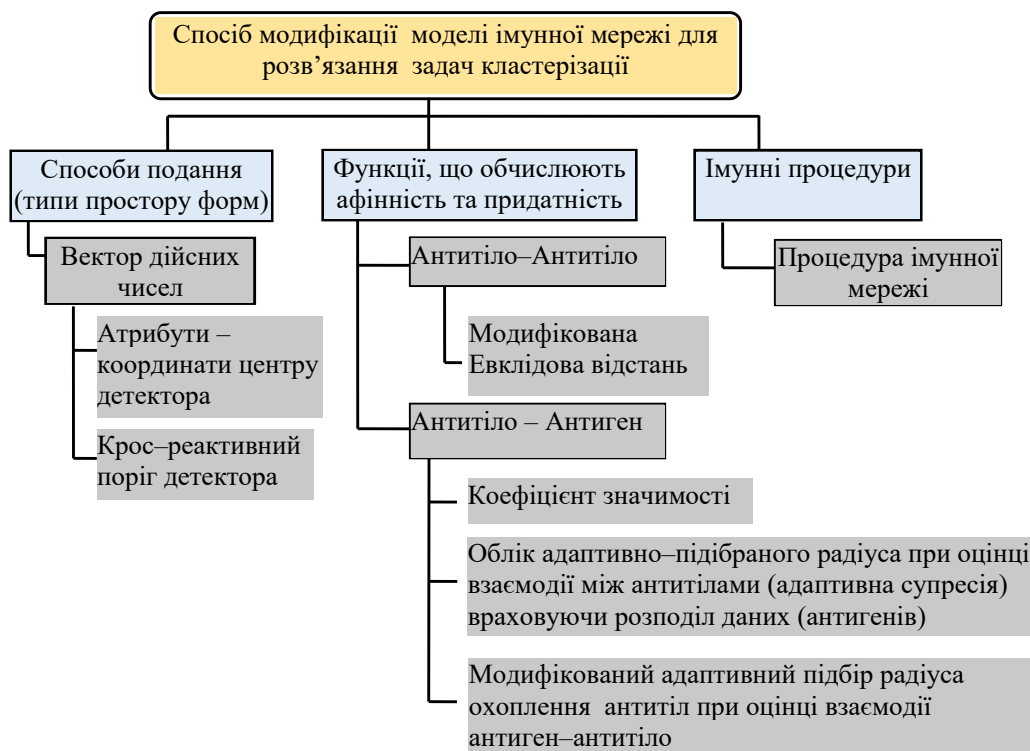


Рисунок 3.14 – Спосіб модифікації моделі імунної мережі для розв'язання задач кластеризації

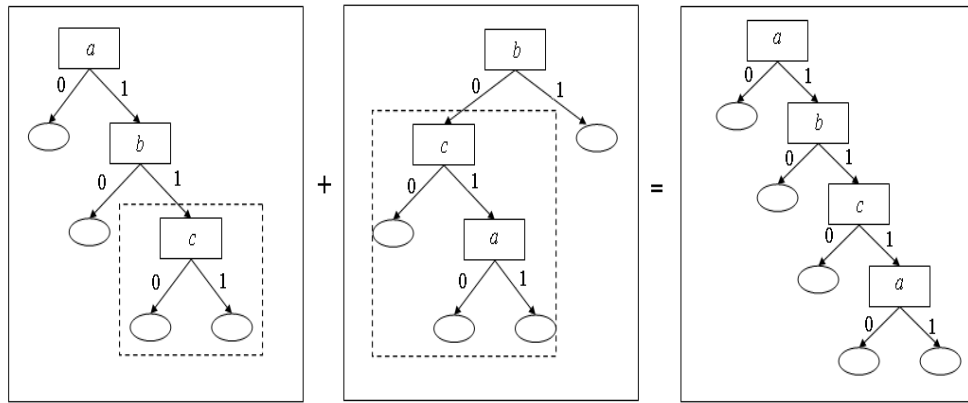


Рисунок 3.15 – Спосіб рішення

Дерево рішень є зручним способом завдання дискретної функції, що залежить від кінцевого числа булевих змінних. Воно являє собою позначене дерево, мітки в якому розставлені за наступним правилом:

- внутрішні вузли позначені символами змінних;
- ребра – значеннями змінних;
- листя – значеннями шуканої функції.

Для визначення значення функції за значеннями змінних необхідно спуститися від кореня до листа, і сформулювати значення, яким позначений отриманий лист.

Наведемо пропонуваний спосіб подачі автомата за допомогою дерев рішень. Для використання подання автоматів у вигляді набору дерев рішень в генетичних алгоритмах визначимо наступні операції:

- випадкове породження автомата – в кожному стані створюється випадкове дерево рішень;
- схрещування автоматів – схрещуються дерева рішень у відповідних станах;
- мутація автомата – у випадковому дереві рішень проводиться мутація.

Тут вважається, що число станів в автоматі фіксоване. Тому протиріч при виконанні певних таким чином операцій не виникне.

4 РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ

4.1 Вибір мови програмування

Для розробки програмних засобів була вибрана мова програмування JAVA. Java – об'єктно-орієнтована мова програмування, розроблений компанією Sun Microsystems (в подальшому придбаної компанією Oracle). Програми на Java транслюються в байт-код, що виконується віртуальною машиною Java (JVM) – програмою, обробній байтовий код і передавальній інструкції обладнанню як інтерпретатор.

Гідність подібного способу виконання програм – в повній незалежності байт-коду від операційної системи і устаткування, що дозволяє виконувати Java-додатки на будь-якому пристрої, для якого існує відповідна віртуальна машина. Іншою важливою особливістю технології Java є гнучка система безпеки завдяки тому, що виконання програми повністю контролюється віртуальною машиною.

Будь-які операції, які перевищують встановлені повноваження програми (наприклад, спроба несанкціонованого доступу до даних або з'єднання з іншим комп'ютером) викликають негайне переривання.

Основні можливості:

- автоматичне керування пам'яттю;
- розширені можливості обробки виняткових ситуацій;
- багатий набір засобів фільтрації введення/виведення;
- набір стандартних колекцій, таких як масив, список, стек і т. П .;
- наявність простих засобів створення мережеских додатків (у тому числі з використанням протоколу RMI);
- наявність класів, що дозволяють виконувати HTTP-запити і обробляти відповіді; – вбудовані в мову засоби створення багатопоточних додатків;
- уніфікований доступ до баз даних: а) на рівні окремих SQL-запитів –

на основі JDBC, SQLJ; б) на рівні концепції об'єктів, що володіють здатністю до зберігання в базі даних – на основі Java Data Objects (англ.) і Java Persistence API;

- підтримка шаблонів (починаючи з версії 1.5);
- паралельне виконання програм. Таким чином, мова Java відповідає всім необхідним вимогам для розробки ПЗ. Як програмне середовище розробки використовується середу Eclipse.

4.2 Розробка інтерфейсу користувача

Інтерфейс програми представлений на рисунку 4.1.

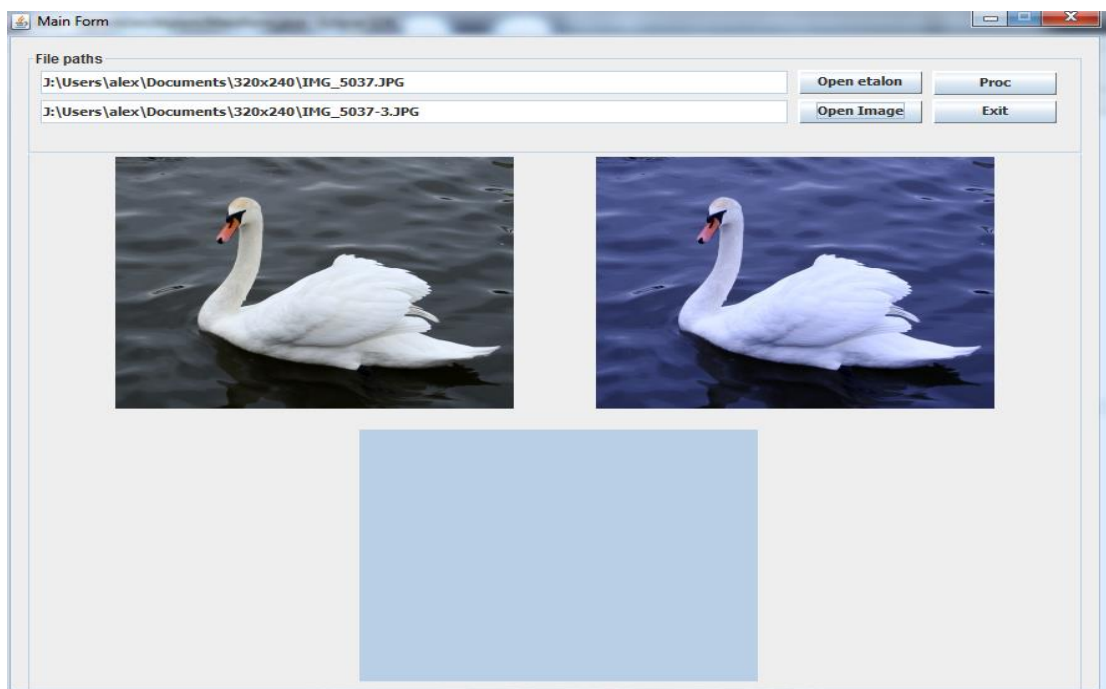


Рисунок 4.1 – Інтерфейс програми

Інтерфейс програми містить 3 поля для зображень, які реалізовані за допомогою класу `ImagePanel`, що включає в себе `JPanel` бібліотеки `java`.

При натисканні кнопки «Open etalon» і «OpenImage» відкривається віконце для вибору файлу, при виборі якого виконується промальовування

обраної картинки. Адреса до картинки відображається в поле JTextField на рівні кнопки.

Лістинг 4.1 – Метод класу ImagePanel

```
public ImagePanel(Image image) {
    this.setDoubleBuffered(true);
    this.image = image;
    this.scale = new Scale(image.getWidth(null),
image.getHeight(null));
    this.repaint();
}
```

При цьому відразу ж виконується буферизація картинки і отримання матриць квітів пікселів, їх нормалізація.

Лістинг 4.2 – Завантаження та обробка зображення

```
public void openFile2() throws IOException {
    Toolkit toolkit = Toolkit.getDefaultToolkit();
    JFileChooser chooser = new JFileChooser();
    Filter filter = new Filter(FilterType.Image);
    chooser.setFileFilter(filter);
    int returnVal = chooser.showOpenDialog(mainForm);
    if (returnVal == JFileChooser.APPROVE_OPTION) {
        String path2 = chooser.getSelectedFile().getAbsolutePath();
        if (filter.accept(path2)) {
            pathImg.setText(path2);
            image = toolkit.getImage(path2);
            imgPanel.setImage(image, new
            Scale(imgPanel.getWidth(), imgPanel.getHeight()));
            BufferedImage img2 = ImageIO.read(chooser.getSelectedFile());
            int w = img2.getWidth();
            int h = img2.getHeight();
            red = new double[w][h];
            green = new double[w][h];
            blue = new double[w][h];
            for(int i=0; i<w; i++){for(int j=0; j<h; j++){
            Color c = new Color(img2.getRGB(i, j));
            red[i][j] = c.getRed();
            green[i][j] = c.getGreen();
            blue[i][j] = c.getBlue(); }};
            for(int i=0; i<w; i++){for(int j=0; j<h; j++){
            red[i][j]=(red[i][j])/255;
            green[i][j]=(green[i][j])/255;
            blue[i][j]=(blue[i][j])/255;}};
            } else {JOptionPane.showMessageDialog(mainForm, "Select the image
            file.", "Warning",
            JOptionPane.WARNING_MESSAGE);}}}
```

4.3 Аналіз компонентів програми

Процес ідентифікації нелінійних залежностей виконується при натисканні кнопки «Proc». Відбувається виклик методу Proc, в якому виконується алгоритм ідентифікації.

У ньому реалізовані функції нейронної мережі і алгоритм клонального відбору для реалізації завдання ідентифікації. В результаті роботи алгоритму ідентифікації заповнюються матриці значеннями 3-х складових кольору пікселя. За допомогою методу setRGB класу BufferedImage числову матрицю перетворимо в картинку. Виводимо її на екран.

Лістинг 4.3 – Алгоритм ідентифікації. Фрагмент

```
public void Proc() throws Exception{
    BufferedImage img3= new BufferedImage(319, 239,
    BufferedImage.TYPE_3BYTE_BGR);
    o=new double[s1];
    out=new double[s1];
    red3 = new int[320][240];
    green3 = new int[320][240];
    blue3 = new int[320][240];
    for (int i=0;i<320;i++){
    for(int j=0;j<240;j++){
    double y1;
    double y2;
    double y3;
    double[] []antib = new double[3] [s1];
    double []besty1 = new double [iter];
    double []besty2 = new double [iter];
    double []besty3 = new double [iter];
    for (int u=0;u<3;u++) {
    for(int x=0;x<s1;x++){
    antib[u][x]=rand.nextDouble();
    }}
    for (int k=0;k<iter;k++){
    y1=0;y2=0;y3=0; double [] F= new double[s1];
    for(int z=0;z<s1;z++){
    for(int x=0;z<s1;z++){
    o[z]=(red2[i][j])*(rand.nextDouble())+(green2[i][j])*(rand.nextD
    ouble())+(blue2[i][j])*(rand.nextDouble());
    out[z]=(1/(1-(Math.exp(-o[z]))));
    y1=out[z]*antib[0][x];
    y2= out[z]*antib[1][x];
    y3=out[z]*antib[2][x];
```

Лістинг 4.4 – Алгоритм ідентифікації. Фрагмент

```

F[z]=(Math.pow((y1-red1[i][j]),2)+Math.pow((y2-
green1[i][j]),2)+Math.pow((y3-blue1[i][j]),2))/2;
if(z==0){
bestanti1 = antib[0][x];
bestanti2 = antib[1][x];
bestanti3 = antib[2][x];
}else if (F[z]<F[z-1]){
bestanti1 = antib[0][x];
bestanti2 = antib[1][x];
bestanti3 = antib[2][x];
}else {F[z]=F[z-1];};
}}
for(int z=0;z<sl;z++){
y1=out[z]*bestanti1;
y2= out[z]*bestanti2;
y3=out[z]*bestanti3;
F[z]=(Math.pow((y1-red1[i][j]),2)+Math.pow((y2-
green1[i][j]),2)+Math.pow((y3-blue1[i][j]),2))/2;
if (z==0){
besty1[k]= y1;
besty2[k]= y2;
besty3[k]= y3;
}else if (F[z]<F[z-1]){
besty1[k]= y1;
besty2[k]= y2;
besty3[k]= y3;
}else{F[z]=F[z-1];};
bestanti1=bestanti1+0.05;
bestanti2=bestanti2+0.05;
bestanti3=bestanti3+0.05;
if (k==0){red3[i][j]=(int) (besty1[k]*255);
green3[i][j]=(int) (besty2[k]*255);
blue3[i][j]=(int) (besty3[k]*255);}else if (((besty1[k]-
red1[i][j])<=(besty1[k-1]-red1[i][j])) && ((besty2[k]-
green1[i][j])<=(besty2[k-1]-green1[i][j]))
&& ((besty3[k]-blue1[i][j])<=(besty3[k-1]-blue1[i][j]))) {
red3[i][j]=(int) (besty1[k]*255);
green3[i][j]=(int) (besty2[k]*255);
blue3[i][j]=(int) (besty3[k]*255);
}else{besty1[k]=besty1[k-1];
besty2[k]=besty2[k-1];
besty3[k]=besty3[k-1];};
}}}
for (int i=0;i<319;i++){
for(int j=0;j<239;j++){
int rgb=(red3[i][j]<<16| green3[i][j]<<8|blue3[i][j]);
img3.setRGB(i, j, rgb);
}};imgPanel3.setImage(img3,new Scale(imgPanel3.getWidth(),
imgPanel3.getHeight()));};

```

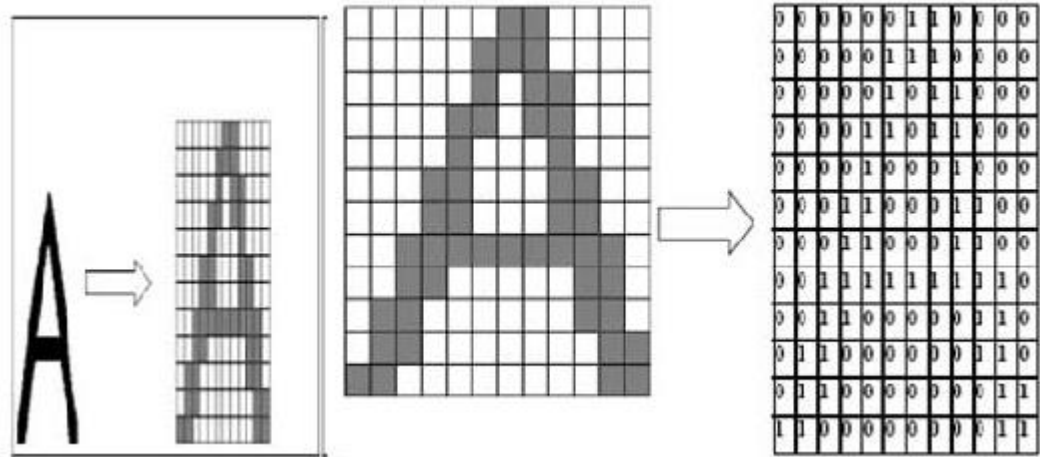


Рисунок 4.2 – Ідентифікація об'єктів на зображенні

4.4 Функціональна верифікація розроблених програмних засобів

Для функціональної верифікації розробленого ПО були взяті тестові зображення: еталонне, зображення освітлене іншим джерелом світла.

На рисунку 4.2 представлені зображення: зліва - еталонне, праворуч – освітлене іншим джерелом світла.



Рисунок 4.3 – Тестові зображення

У процесі відкриття картинок ми отримуємо матриці значень кольорів пікселів, при запуску алгоритму ідентифікації проводиться обчислення

функцій. Кожен елемент пікселя за допомогою лінійної функції активації і випадкових ваг перетворимо для прихованого шару нейромережі, де отримуємо значення сигмоїдальної функціонально, далі для значень виходів використовуємо лінійну функцію з набором ваги, які є антитілами. На основі алгоритму клонального відбору виробляємо навчання заданого алгоритму нейромережі. У підсумку маємо відображення результату (рисунок 4.3).



Рисунок 4.3 – Скріншот

Для порівняння розроблених методів і моделей класифікації у середовищі моделювання було сформовано набори класифікованих даних, які відрізняються між собою кількістю об'єктів, класів та характеристик кожного з них. Для аналізу методів класифікації з контрольованим навчанням набори даних, наведені на слайді у верхній таблиці, використовувалися без змін, а для аналізу методів кластеризації групуванню підлягали тільки об'єкти навчальної вибірки.

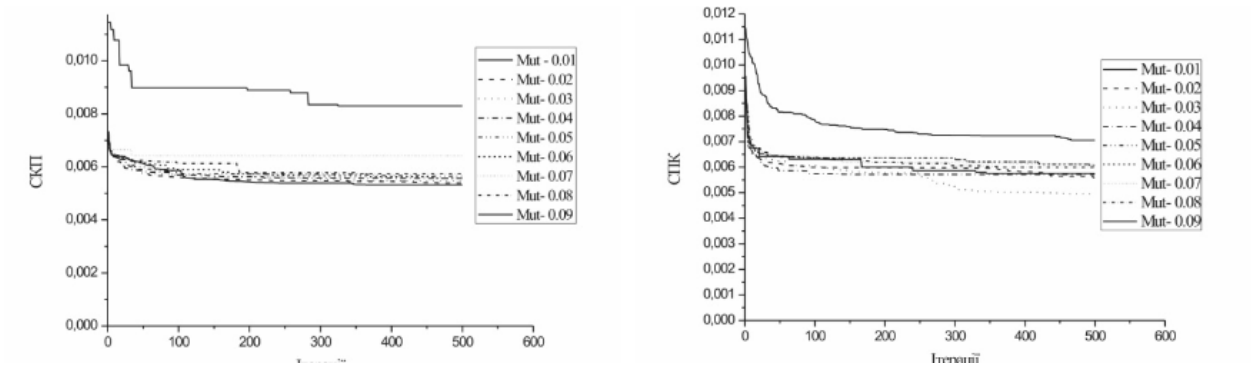


Рисунок 4.4 – Збіжність алгоритму при різних рівнях мутації

У роботі досліджувався вплив рівня мутації на збіжність алгоритму. Загальний рівень мутації послідовно відбирався: 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09. Для навчання мережі використовувалося 500 ітерацій. Як міра якості оцінки класифікатора використовувалося середньоквадратичне відхилення.

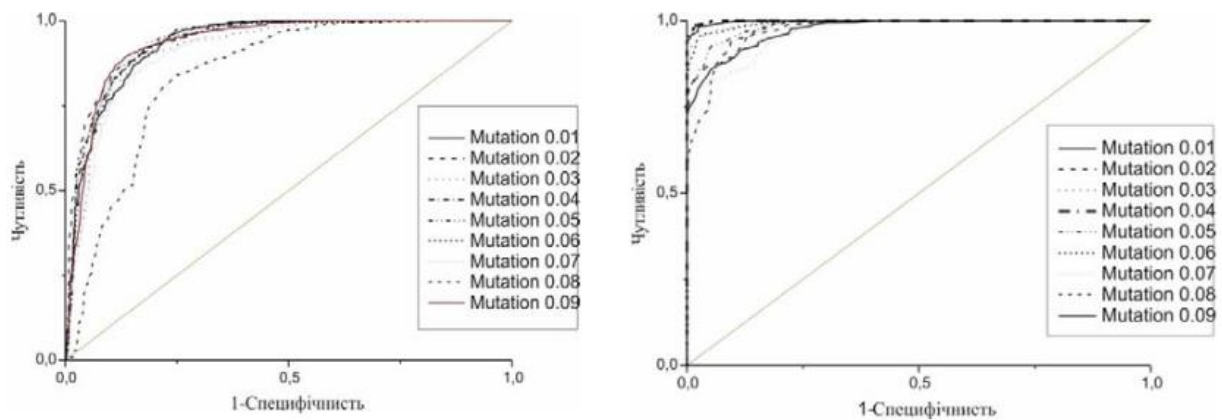


Рисунок 4.5 – Крива ROC

Для оцінки отриманих колективів бінарних нейромережних класифікаторів використовувалися ROC-криві, AUC – чисельний показник площі під кривою і % правильно розпізнаних об'єктів на тестовій множині.

ВИСНОВКИ

ШС здатні адаптуватися до змін в даних та середовищі, що дозволяє їм ефективно працювати в умовах нестабільних і динамічних даних. В ШС використовуються багато агентів, що працюють незалежно, що забезпечує високу стійкість до збоїв та можливість паралельної обробки даних. Також вони мають здатність до самонавчання і можуть покращувати свою ефективність з часом, використовуючи накопичені знання та досвід. ШС демонструють високу стійкість до шуму та невизначеності в даних, що дозволяє їм зберігати точність навіть при роботі з неякісними або неповними даними. В них можуть використовувати різні підходи до класифікації, включаючи детектування аномалій та кластеризацію, що робить їх універсальними для різних завдань. ШС легко інтегруються з іншими обчислювальними моделями, такими як нейронні мережі, що дозволяє створювати гібридні системи з підвищеною ефективністю. Використання принципів імунної системи людини надає ШС природну здатність до розпізнавання та класифікації, що відображає природні механізми захисту та адаптації. Ці переваги роблять штучні імунні системи привабливим інструментом для задач класифікації даних, особливо в умовах складних і змінних середовищ.

Проведено аналіз існуючих моделей та методів розпізнавання символів на зображеннях з використанням штучних імунних мереж. Досліджені існуючі методи розпізнавання символів на зображеннях; проведено огляд алгоритмів штучних імунних мереж; розроблена модель деревовидної штучної мережі для розпізнавання символів. Отримані результати показали доцільність використання розробленої моделі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Бурда А.С., Прудіус М.А., Стефанюк Я.Г., Фомічов О.О. Методи обробки та інтелектуального аналізу даних з використанням штучних імунних систем // Системи управління, навігації та зв'язку. 2024. № 3. С.100-103.
2. Ротштейн А. П. Интеллектуальные технологии идентификации: нечеткая логика, генетические алгоритмы, нейронные сети.– Винница: УНИВЕРСУМ – Винница, 1999.– 320 с.
3. Искусственные иммунные системы и их применение / Под ред. Д. Дасгупты. Пер. с англ. под ред. А.А.Романюхи. – М.: ФИЗМАТЛИТ, 2006. – 344 с.
4. Timmis J. An Overview of Artificial Immune Systems / J. Timmis, T. Knight, L.N. de Castro, E. Hart // Computation in Cells and Tissues: Perspectives and Tools for Thought, Natural Computation Series, Springer, 2004 – pp. 51-86.
5. Castro L. N. Artificial Immune Systems: A Novel Paradigm to Pattern Recognition / Castro L. N., Timmis J. I. // In Artificial Neural Networks in Pattern Recognition, SOCO-2002, University of Paisley, UK, pp. 67-84.
6. Castro L. N. The Clonal Selection Algorithm with Engineering Applications / Castro L. N., Von Zuben, F. J. // In Proceedings of GECCO'00, Workshop on Artificial Immune Systems and Their Applications, pp. 36-37.
7. Борисов В.В., Круглов В.В., Федулов Ф.С. Нечеткие модели и сети. – М: Горячая линия. – Телеком, 2007. 7. Е.В. Бодянский, О.Г. Руденко Искусственные нейронные сети: архитектуры, обучение, применения // Харьков: ТЕЛІТЕХ, 2004. – 372с.
8. Растрингин Л.А. Адаптация сложных систем. Методы и приложения // Рига: Зинатне, 1981.– 375 с.
9. Цыпкин Я.З. Адаптация и обучение в автоматизированных системах // М., Наука, 1968.

10. Lauren M. Sompayrac *How the Immune System Works*. 4th Edition // Wiley-blackwell. 2012. – 153 p.
11. Kephart, J. O. "A biologically inspired immune system for computers". *Proceedings of Artificial Life IV: The Fourth International Workshop on the Synthesis and Simulation of Living Systems*. MIT Press. 1994. – pp. 130–139.
12. Andrews and Timmis "A Computational Model of Degeneracy in a Lymph Node". *Artificial Immune Systems. Lecture Notes in Computer Science*. Vol. 4163. 2006. – pp. 164 –177. doi:10.1007/11823940_13.
13. Mendao; et al. "The Immune System in Pieces: Computational Lessons from Degeneracy in the Immune System". *2007 IEEE Symposium on Foundations of Computational Intelligence*. 2007. – pp. 394–400. doi:10.1109/FOCI.2007.371502.
14. Greensmith, J.; Aickelin, U. "Artificial Dendritic Cells: Multi-faceted Perspectives". *Human-Centric Information Processing Through Granular Modelling (PDF)*. *Studies in Computational Intelligence*. Vol. 182.2009. – pp. 375–395. doi:10.1007/978-3-540-92916-1_16.