

ДОДАТОК А

Перелік наукових праць співробітників кафедри ПІ за темою дослідження

14. A. Yerokhin, O. Zolotukhin. Fuzzy Probabilistic Neural Network in Document Classification Tasks. Interbranch collection of scientific papers «Information Extraction and Processing». National Academy of Sciences of Ukraine. -2019. <http://vidbir.ipm.lviv.ua/>
35. Maksym Bekuzarov, Oleksandr Samantsov, Oksana Mazurova, Mariia Shirokopetleva. Neural Network Architecture Editor With Code Generation. Problem of Infocommunications. Science and Technology (PIC S&T'2020), Kharkiv, Ukraine.- 6-9 October 2020.
36. Turevska, O. , Shubin, I. Improving the automated testing of Web-based services by reflecting the social habits of target audiences. 2015 Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, с. 93-96.
37. О.Ф.Лановий, О.В.Золотухін. Застосування нейромережевого підходу для класифікації втручань в роботу комп'ютерних систем // Застосування інформаційних технологій у діяльності НПУ: матеріали наук.-практ. семінару (м.Харків, 21 грудня 2018 р.) / МВС України, Харк.нац.ун-т внутр.справ. Харків. ХНУВС, 2018.— С.78-79.
38. Лановий О.Ф. Про один підхід до функціонального тестування web-додатків // Полиграфические, мультимедийные и web-технологии. Т1. Тез. докл. 2-й Международ. науч.-техн. конф.(16-22 мая 2017) / редкол.: ВФ Ткаченко, ИБ Чеботарева и др.—Харьков: ХНУРЭ, 2017.—246 с.
39. Лановий О.Ф. Візуалізація в методах тестування програмного забезпечення // Харків, 6-а Міжнародна науково-технічна конференція «ІСТ-2017», ХНУРЕ, 11-16 вересня 2017 р., С.110-111.

ДОДАТОК Б
Слайди презентації

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

АТЕСТАЦІЙНА РОБОТА МАГІСТРА

**Дослідження методів нейромережевих технологій при
проведенні функціонального тестування програмного
забезпечення**

Виконав:
ст. гр. ПЗСрм-19-1
Лановий О.Ф.

Науковий керівник:
проф. Дудар З.В.

Рисунок Б.1 – Титульний слайд

Мета роботи

Вдосконалення методів тестування програмного забезпечення з використанням машинного навчання та алгоритмів штучного інтелекту.

В роботі досліджуються різні підходи до тестування програмного забезпечення з використанням ШІ, який починає грати все більш важливу роль, знаходячи своє застосування в інструментах тестування, розроблених протягом останніх кількох років. Основна мета дослідження полягає у формуванні рекомендацій з використання нейромережевих технологій для підвищення ефективності функціонального тестування програмного забезпечення.

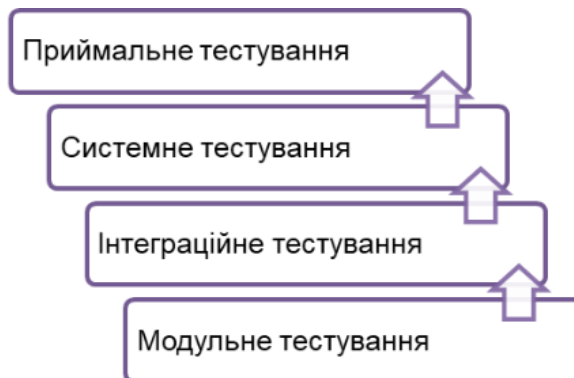
Рисунок Б.2 – Мета роботи

Аналіз предметної галузі

Тестування є основним способом контролю якості програмних рішень і може бути визначено як процес виконання програми з метою перевірки відповідності між її реальною та очікуваною поведінкою.

Протягом життєвого циклу розробки і супроводу програмного забезпечення тестування здійснюється на різних його рівнях.

Проведення тестування методом «чорного ящика» спрямовано на виявлення дефектів: неправильні або відсутні функції системи, недоліки інтерфейсу, дефекти в структурах даних або під час доступу до зовнішніх баз даних, дефекти продуктивності, дефекти ініціалізації та завершення тощо.



3

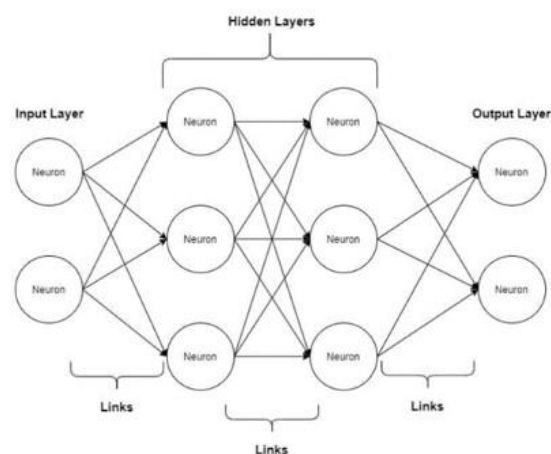
Рисунок Б.3 – Аналіз предметної галузі

Аналіз предметної галузі

Автоматизоване тестування є аналогом ручного, проте для виконання тестів і оцінки їх результатів використовуються програмні засоби.

Багатошарова нейронна мережа навчається на оригінальному програмному додатку за допомогою випадково сформованих тестових даних, що відповідають специфікації.

Штучні нейронні мережі – це апроксиматори функцій, які зазвичай використовуються для визначення безперервних станів та дій об'єкту моделювання.



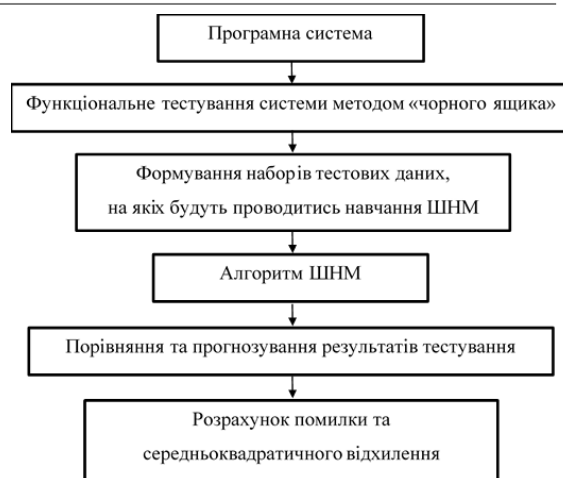
4

Рисунок Б.4 – Аналіз предметної галузі (продовження)

Постановка задачі

Програмне забезпечення за своєю суттю представляє функцію перетворення вхідної інформації у вихідну в багатовимірному просторі. В такий математичній моделі досить легко застосувати нейронні мережі.

Для досягнення мети дослідження в роботі планується створити модель ШНМ для прогнозування рівня точності отриманих результатів при тестуванні ПЗ методом «чорного ящика».



5

Рисунок Б.5 – Постановка задачі

Постановка задачі

Для досягнення мети дослідження виконаємо деталізацію локальних задач, які необхідно розв'язати в роботі.

Локальна задача 1. Відповідно до специфікації визначити набори тестових даних.

Локальна задача 2. Поділити отримані набори у відповідності до класів еквівалентностей.

Локальна задача 3. Визначити набори даних, що відносяться до навчальної виборки.

Локальна задача 4. Нормалізувати вагові коефіцієнти для кожного з наборів даних.

Локальна задача 5. Розробити алгоритм динамічної зміни вагових коефіцієнтів з використанням ШНМ для досягнення мети дослідження.

6

Рисунок Б.6 – Постановка задачі (продовження)

Сучасний стан проблеми



Scopus

Найбільш популярними алгоритмами ШІ в тестуванні є:

- кластеризація (загальний метод неконтрольованого навчання)
- ШНМ (використовується в навчанні під контролем та підкріплення)
- ГА (зазвичай використовується в навчанні з підкріпленням)

В роботі «AI for Testing Today and Tomorrow: Industry Perspectives» вказано: «...ключові переваги тестування, керованого штучним інтелектом, полягають в тому, що воно є універсальним, багаторазовим, надійним, адаптивним та гарно масштабується».

Асоціація штучного інтелекту для тестування програмного забезпечення (AISTA): «Це нова область, що спрямована на розробку систем штучного інтелекту для тестування програмного забезпечення, методів тестування систем зі штучним інтелектом та, у підсумку на розробку програмного забезпечення, здатного до самоперевірки і самоцілювання»

7

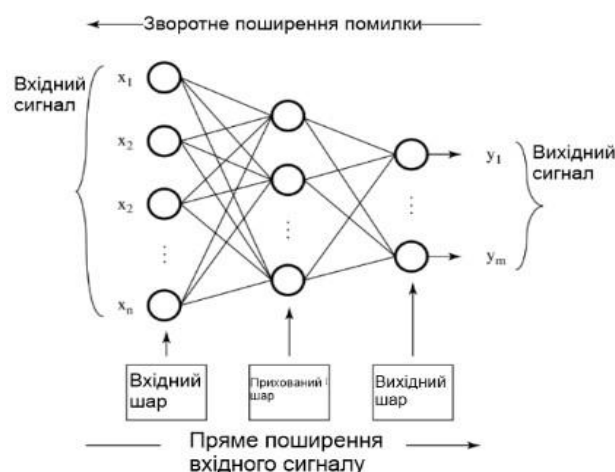
Рисунок Б.7 – Сучасний стан проблеми

Проектування мережі

Алгоритм полягає в передачі вхідного сигналу в прямому напрямку і сигналу помилки – в зворотному. При прямому проході вектор вхідних сигналів подається на вхідний шар ШНМ, після цього він поступово поширюється по мережі від одного шару до наступного. В результаті утворюється набір вихідних сигналів, які представляють собою фактичну реакцію мережі на вхідний набір.

$$net = \sum_{j=1}^n x_j w_{ij}$$

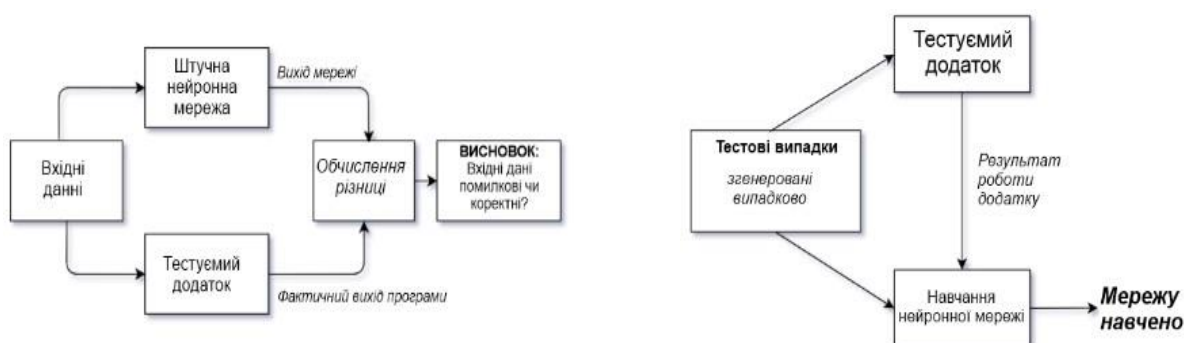
$$y_i = \frac{1}{1 + e^{-net}}$$



8

Рисунок Б.8 – Проектування мережі

Методика дослідження

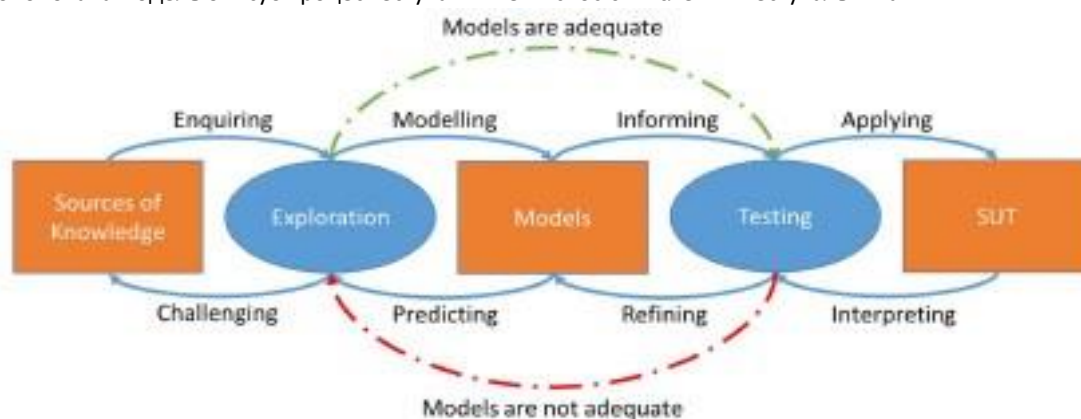


9

Рисунок Б.9 – Методика дослідження

Модель тестування ПЗ

Модель тестування програмного забезпечення було запропоновано Полом Джерардом (Paul Gerrard) у 2017 р. Запропонована модель описує процес тестування ПЗ як спосіб мислення тестувальника.



10

Рисунок Б.10 – Модель тестування ПЗ

Використання моделі тестування

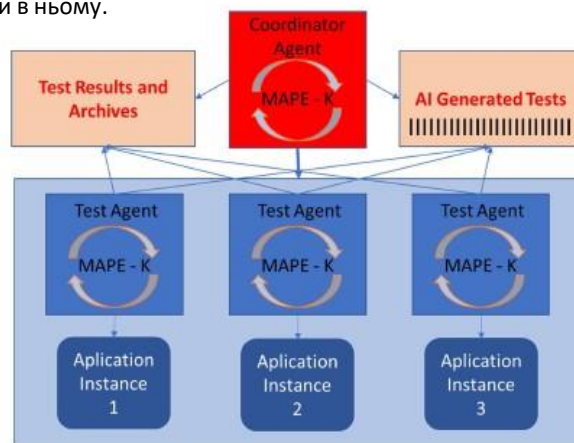
Тестування з використанням ШНМ передбачає використання інтелектуальних самонавчаємих агентів, які можуть автономно сприймати навколишнє середовище і діяти в ньому.

Агенти-координатори базуються на використанні циклу управління MAPE-K, представленого фірмою IBM

Тестові агенти автономні та повинні діяти в середовищі цільового додатку незалежно один від одного, не взаємодіючи між собою

На етапі аналізу і проектування тестів агенти тестування досліджують додаток

На етапі виконання агенти виконують тести і зберігають результат в архівах результатів тестування



11

Рисунок Б.11 – Використання моделі тестування

Інструментальні засоби підтримки

Фазінг (Fuzzing) – це одна з технологій тестування програмного забезпечення, в якій замість очікуваних вхідних даних програмі передаються випадкові або спеціально сформовані дані. Метою такого дослідження поведінки програмної системи є спроба знайти непередбачені розробниками ПЗ вхідні дані, які призводять до аварійного завершення програми або до її некоректної поведінки.



12

Рисунок Б.12 – Інструментальні засоби підтримки

Проведення досліджень

Для перевірки правильності гіпотези щодо ефективності використання неймережевого підходу при виконанні функціонального тестування ПЗ було проведено тестування відокремленої підсистеми ІАС Університет для формування академічної довідки.

Оцінка з дисципліни	Оцінка за національною шкалою		Оцінка за шкалою ЄКТС
	екзамен	залік	
96-100	5 (відмінно)	Зараховано	A
90-95	5 (відмінно)		B
75-89	4 (добре)		C
66-74	3 (задовільно)		D
60-65	3 (задовільно)		E
35-59	2 (незадовільно)	Не зараховано	FX
1-34			F

13

Рисунок Б.13 – Проведення досліджень

Проведення досліджень: класи еквівалентностей

Для перевірки правильності гіпотези щодо ефективності використання неймережевого підходу при виконанні функціонального тестування ПЗ було проведено тестування відокремленої підсистеми ІАС Університет для формування академічної довідки.

Позначимо оцінку з дисципліни $O_{ц}$, значення якої може бути тільки цілим позитивним числом.

ПКЕ (Національна шкала, залік):
($60 \leq O_{ц} \leq 100$);

НКЕ (Національна шкала, залік):
($O_{ц} < 60$; $O_{ц} > 100$);

ПКЕ (Національна шкала, екзамен):
($0 < O_{ц} \leq 59$; $60 \leq O_{ц} \leq 74$; $75 \leq O_{ц} \leq 89$; $90 \leq O_{ц} \leq 100$);

НКЕ (Національна шкала, екзамен):
($O_{ц} < 0$; $O_{ц} > 100$);

ПКЕ (Шкала ЄКТС):
($0 < O_{ц} \leq 59$; $60 \leq O_{ц} \leq 74$; $75 \leq O_{ц} \leq 89$; $90 \leq O_{ц} \leq 95$; $96 \leq O_{ц} \leq 100$);

НКЕ (Шкала ЄКТС):
($O_{ц} < 0$; $O_{ц} > 100$).

14

Рисунок Б.14 – Проведення досліджень: класи еквівалентностей

Проведення досліджень: уточнені класи еквівалентностей

Національна шкала, залік:

ПКЕ_{зал}: ($60 \leq O_{\text{ц}} \leq 100$);
 НКЕ_{зал}: ($O_{\text{ц}} \leq 59$; $O_{\text{ц}} > 100$);
 ПКЕ_{нез}: ($1 \leq O_{\text{ц}} \leq 59$);
 НКЕ_{нез}: ($O_{\text{ц}} \leq 0$; $O_{\text{ц}} > 59$)

Національна шкала, екзамен:

ПКЕ₅: ($90 \leq O_{\text{ц}} \leq 100$);
 НКЕ₅: ($O_{\text{ц}} \leq 89$; $O_{\text{ц}} > 100$);
 ПКЕ₄: ($75 \leq O_{\text{ц}} \leq 89$);
 НКЕ₄: ($O_{\text{ц}} \leq 74$; $O_{\text{ц}} > 89$);
 ПКЕ₃: ($60 \leq O_{\text{ц}} \leq 74$);
 НКЕ₃: ($O_{\text{ц}} \leq 59$; $O_{\text{ц}} > 74$);
 ПКЕ₂: ($1 \leq O_{\text{ц}} \leq 59$);
 НКЕ₂: ($O_{\text{ц}} \leq 0$; $O_{\text{ц}} > 59$).

Шкала ЄКТС:

ПКЕ_A: ($96 \leq O_{\text{ц}} \leq 100$);
 НКЕ_A: ($O_{\text{ц}} \leq 95$; $O_{\text{ц}} > 100$);
 ПКЕ_B: ($90 \leq O_{\text{ц}} \leq 95$);
 НКЕ_B: ($O_{\text{ц}} \leq 89$; $O_{\text{ц}} > 95$);
 ПКЕ_C: ($75 \leq O_{\text{ц}} \leq 89$);
 НКЕ_C: ($O_{\text{ц}} \leq 74$; $O_{\text{ц}} > 89$);
 ПКЕ_D: ($66 \leq O_{\text{ц}} \leq 74$);
 НКЕ_D: ($O_{\text{ц}} \leq 65$; $O_{\text{ц}} > 74$);
 ПКЕ_E: ($60 \leq O_{\text{ц}} \leq 65$);
 НКЕ_E: ($O_{\text{ц}} \leq 59$; $O_{\text{ц}} > 65$);
 ПКЕ_F: ($35 \leq O_{\text{ц}} \leq 59$);
 НКЕ_F: ($O_{\text{ц}} \leq 34$; $O_{\text{ц}} > 59$);
 ПКЕ_{FX}: ($1 \leq O_{\text{ц}} \leq 34$);
 НКЕ_{FX}: ($O_{\text{ц}} \leq 0$; $O_{\text{ц}} > 34$).

15

Рисунок Б.15 – Проведення досліджень: уточнені класи еквівалентностей

Зведена таблиця результатів

Помилка	Оцінка помилки
Функціональна	0
Структури даних	0
Інтерфейсу	0
Ініціалізації	0
Продуктивності	10
Сумарна помилка	10
Діапазон значень екзаменаційної оцінки	$75 \leq O_{\text{ц}} \leq 89$
Очікуваний результат	C

16

Рисунок Б.16 – Зведена таблиця результатів

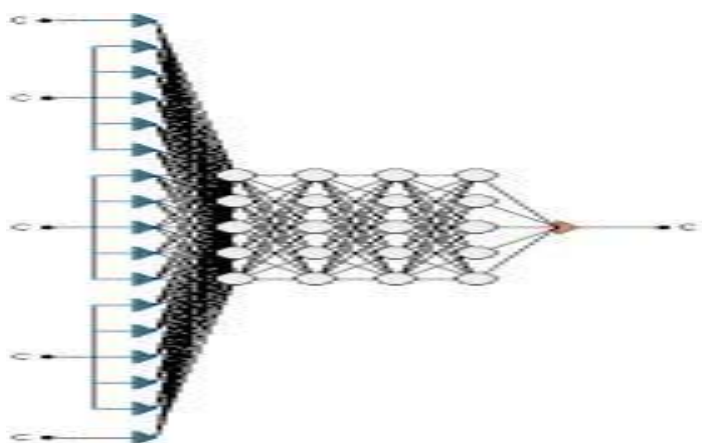
Нормалізація результатів навчання

Форма тесту	Номер тесту	Тестер	Категорія помилки	Сумарна помилка	Дефект	Результат
1	1	0,12	0,12	0,3	0,12	Correct
1	2	0,12	0,14	0,5	0,12	Correct
1	3	0,12	0,16	0,7	0,12	Correct
1	4	0,12	0,18	0,9	0,12	Correct
1	5	0,12	0,1	0,1	0,1	InCorrect
...
40	5	0,1	0,1	0,1	0,1	Correct
40	2	0,1	0,1	0,1	0,1	Correct
40	3	0,1	0,1	0,1	0,1	Correct
40	4	0,1	0,1	0,1	0,1	Correct
40	5	0,1	0,1	0,1	0,1	Correct

17

Рисунок Б.17 – Нормалізація результатів навчання

Архітектура штучної нейронної мережі



18

Рисунок Б.18 – Архітектура штучної нейронної мережі

Результати досліджень: 900 епох

Рядок	Формула	Реальні дані		Прогноз		Співпадіння	Імовірність
		Дефект	Ціль	Дефект	Вихід		
0	1	0	InCorrect	0,000812	InCorrect	Так	0,999188
8	2	0	InCorrect	0,000812	InCorrect	Так	0,999188
13	3	1	Correct	0,999903	Correct	ОК	0,999903
18	4	0	InCorrect	0,000813	InCorrect	ОК	0,999187
22	5	0	InCorrect	0,000813	InCorrect	ОК	0,999187
31	7	0	InCorrect	0,000813	InCorrect	ОК	0,999187
32	7	1	Correct	0,999908	Correct	ОК	0,999908
37	8	0	InCorrect	0,000813	InCorrect	ОК	0,999187
48	10	0	InCorrect	0,000812	InCorrect	ОК	0,999188
51	11	1	Correct	0,999908	Correct	ОК	0,999908
56	12	1	Correct	0,999908	Correct	ОК	0,999908
62	13	1	Correct	0,999908	Correct	ОК	0,999908
66	14	0	InCorrect	0,000813	InCorrect	ОК	0,999187
72	15	0	Correct	0,999908	Correct	ОК	0,999908
75	16	0	InCorrect	0,000855	InCorrect	ОК	0,999145
80	17	1	Correct	0,999901	Correct	ОК	0,999901
85	18	1	Correct	0,999904	Correct	ОК	0,999904
89	18	1	Correct	0,999908	Correct	ОК	0,999908
96	20	1	Correct	0,999908	Correct	ОК	0,999908
100	21	1	Correct	0,999884	Correct	ОК	0,999884

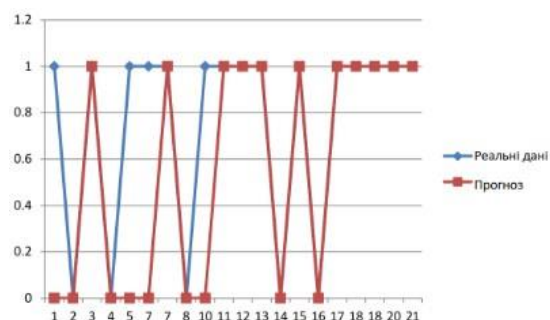


Рисунок Б.19 – Результати досліджень: 900 епох

Результати досліджень: 1000 епох

Рядок	Формула	Реальні дані		Прогноз		Співпадіння	Імовірність
		Дефект	Ціль	Дефект	Вихід		
0	1	1	Correct	0,999891	Correct	ОК	0,999891
8	2	InCorrect	0,000856	InCorrect	ОК	0,999144	
13	3	1	Correct	0,999885	Correct	ОК	0,999885
18	4	InCorrect	0,000855	InCorrect	ОК	0,999145	
22	5	1	Correct	0,999885	Correct	ОК	0,999885
31	7	1	Correct	0,999890	Correct	ОК	0,99989
32	7	1	Correct	0,999891	Correct	ОК	0,999891
37	8	InCorrect	0,000855	InCorrect	ОК	0,999145	
48	10	1	Correct	0,999891	Correct	ОК	0,999891
51	11	1	Correct	0,999882	Correct	ОК	0,999882
56	12	1	Correct	0,999890	Correct	ОК	0,99989
62	13	1	Correct	0,999891	Correct	ОК	0,999891
66	14	InCorrect	0,000853	InCorrect	ОК	0,999147	
72	15	1	Correct	0,999891	Correct	ОК	0,999891
75	16	InCorrect	0,000855	InCorrect	ОК	0,999145	
80	17	1	Correct	0,999884	Correct	ОК	0,999884
85	18	1	Correct	0,999890	Correct	ОК	0,99989
89	18	1	Correct	0,999881	Correct	ОК	0,999881
96	20	1	Correct	0,999889	Correct	ОК	0,999889
100	21	1	Correct	0,999890	Correct	ОК	0,99989

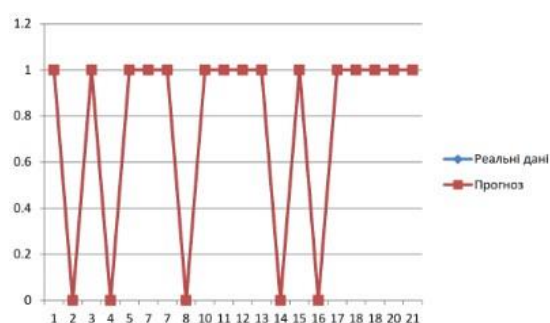


Рисунок Б.20 – Результати досліджень: 1000 епох

Результати досліджень: висновки

За отриманими результатами можна зробити висновки, що рівень точності результатів, отриманих з використанням ШНМ при проведенні тестування програмного забезпечення методом чорного ящика має точність близько 85%. Для оцінювання використовувалась характеристика AUC (Area Under Curve), яка є агрегированной характеристикою якості класифікації

<i>Кількість прихованих шарів ШНМ</i>	<i>Кількість epoch</i>	<i>Швидкість навчання</i>	<i>Точність</i>	<i>AUC</i>
4	900	0,1	85,00%	0,850 +/- 0,071
4	1000	0,1	99,00%	0,976 +/- 0,071
5	1000	0,1	80,00%	0,800 +/- 0,071

21

Рисунок Б.21 – Результати досліджень: висновки

Висновки по роботі

В роботі були детально досліджені різні підходи щодо використання нейромережевих технологій для тестування програмного забезпечення, проаналізовано підходи науковців та розробників програмного забезпечення до оцінювання ролі, яку відіграють алгоритми ШІ в тестуванні ПЗ.

В результаті написання атестаційної роботи за темою дослідження було проведено: аналіз першоджерел; виконано дослідження методів та алгоритмів тестування програмного забезпечення; визначено можливі шляхи покращення якості тестування за рахунок використання нейромережевих підходів як для генерації тестів, так і для формування тестових наборів даних; виконано постановку задачі дослідження; запропоновано методику проведення функціонального тестування ПЗ з використанням нейромережевих технологій – використання штучних нейронних мереж для підтримки прийняття рішень множинних інтелектуальних агентів, що відстежують проведення тестування; розробку структури ШНМ, розроблено алгоритм навчання на підготовлених наборах даних, виконано експериментальні дослідження, які наочно довели ефективність використання цієї методики.

22

Рисунок Б.19 – Висновки по роботі