

## **Искусственные нейронные сети в задачах поддержки принятия решений при повторном системном тестировании web-приложений**

*Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ»*

Представлено обоснование необходимости проведения повторного системного тестирования web-приложений. Обоснован выбор метода реализации поддержки принятия решений при повторном системном тестировании web-приложений. Описана методика обнаружения ошибок с помощью повторного системного тестирования web-приложений. Приведена модель для поддержки принятия решений при повторном системном тестировании web-приложений.

**Ключевые слова:** web-приложение, основное системное тестирование, повторное системное тестирование, поддержка принятия решений, искусственные нейронные сети.

### **1. Введение**

В наше время размах коммерческой деятельности в сети Интернет приобретает ошеломляющие масштабы. Различные фирмы и отдельные пользователи все больше и больше в своей работе зависят от онлайн-приложений. Функциональные возможности web-приложений непрерывно растут. Чтобы успешно существовать на этом рынке, необходимо серьезно подойти к проблеме тестирования web-приложений.

Клиент-серверная архитектура web-приложений предполагает структуру системы, в состав которой входят три основные составляющие: сервер, клиент и база данных. Соответственно вероятность возникновения ошибок в такой системе возрастает.

Способность web-приложений обрабатывать и распространять информацию через Интернет делает их очень уязвимыми, поскольку они полностью открыты для обеспечения публичного доступа.

Именно поэтому от тестовой инфраструктуры требуется выявлять значительное количество дефектов в приложениях на как можно более ранних стадиях.

Многие компании, занимающиеся созданием web-приложений, до 50 % средств, выделенных на разработку, тратят на тестирование, что составляет миллиарды долларов по всему миру в целом. Но, несмотря на это, большинство web-сайтов остаются неприемлемо ненадежными даже после «основательного тестирования».

После системного тестирования web-приложения вследствие ряда причин в приложении остаются ошибки. Такие ошибки относятся к скрытым ошибкам, т.е. таким, которые остались в программном продукте после его тестирования.

Скрытые ошибки отличаются от обнаруженных тем, что они на определенный момент времени после тестирования существуют, еще не обнаружены и могут неожиданным образом повлиять на работу web-приложения в любой момент. Таким образом, скрытые ошибки являются чрезвычайно опасными, т.к. никогда неизвестно, где и как они себя могут проявить.

Выявление скрытых ошибок должно проводиться после разработки и отладки web-приложений, где системное тестирование осуществлялось как частичная технологическая операция. Процесс выявления скрытых ошибок можно считать

отдельным технологическим процессом. В таком случае условно поделим системное тестирование web-приложений на два вида: основное и повторное.

Основное – системное тестирование, которое осуществляется на этапах разработки и отладки web-приложений и является частичной составляющей этих процессов.

Повторное – системное тестирование с целью выявления скрытых ошибок, которое осуществляется после разработки и отладки web-приложений и является отдельным технологическим процессом.

Таким образом, повысить эффективность тестирования и, соответственно, качество разрабатываемого продукта можно не только путем обнаружения дефектов на этапах разработки и отладки, но и путем повторного тестирования с целью выявления скрытых ошибок в web-приложениях после основного тестирования.

## 2. Методика обнаружения ошибок с помощью повторного системного тестирования web-приложений

Вследствие ряда причин, таких, как несовершенство методов тестирования, неполнота тестов, недостаточность средств и квалификации разработчиков и тестировщиков, влияние человеческого фактора, после основного системного тестирования в web-приложениях остаются необнаруженные ошибки. Эти ошибки могут быть выявлены в процессе повторного системного тестирования.

На рис. 1 показано, какое место отводится повторному тестированию в жизненном цикле программного обеспечения, используя каскадную модель жизненного цикла ПО [1, 2].

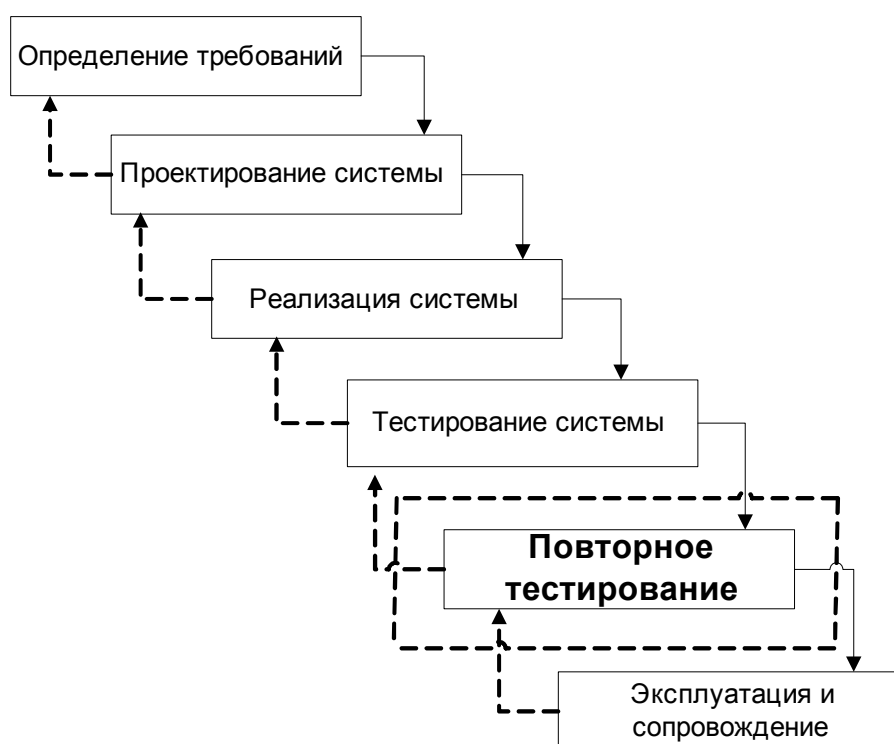


Рис. 1. Повторное тестирование в рамках каскадной модели жизненного цикла программного обеспечения

В результате анализа методов, реализующих технологию поддержки принятия, для решения задачи обнаружения скрытых ошибок путём повторного системного тестирования web-приложений было решено использовать искусственные нейронные сети (ИНС), т.к. в процессе обучения ИНС способны выявлять сложные зависимости между входными и выходными данными. Они являются универсальными структурами, позволяющими реализовать любой вычислительный алгоритм.

Суть методики выявления оставшихся после основного тестирования ошибок заключается в последовательном обнаружении ошибок определённого типа с последующим устранением причин их возникновения.

Касательно распределения ошибок по видам и влиянию на работу web-приложений в литературе известно распределение их по приоритетам и категориям [3].

Для описания ненайденных ошибок в процессе повторного системного тестирования будем использовать следующую классификацию: *критические* - системные сбои, когда какая-то часть ПО рухнет, потеря данных - чаще всего это происходит, когда данные не достигают базы данных либо незапланированно удаляются из нее, проблема с безопасностью - например, когда после логина пароль виден как часть URL, так что кто-то может подсмотреть пароль и использовать его в своих корыстных целях; *значительные* - блокируют кодирование, тестирование или использование web-приложения, ситуация, когда работа не может быть продолжена, так как на одном из этапов появляется проблема, препятствующая дальнейшему продвижению; *умеренные* - функциональные проблемы, как правило, это простое расхождение между фактическим и ожидаемым результатами; *косметические* – ошибки, связанные с содержанием web-приложения, правописанием и интерфейсом пользователя.

Присвоим каждому виду ошибок определённый уровень категориальности: косметические – 1-й уровень категориальности С1, умеренные – 2-й уровень категориальности С2, значительные – 3-й уровень категориальности С3, критические – 4-й уровень категориальности С4.

На основании распределения невыявленных ошибок по категориям будет введен порог допустимого количества ошибок разных типов одной категории, при превышении которого необходимо осуществлять повторное системное тестирование для выявления оставшихся ошибок этой же категории.

Исходными данными для реализации повторного системного тестирования являются операция тестирования и тип ошибок, обнаруженных во время основного системного тестирования.

### **3. Модель для поддержки принятия решений при повторном системном тестировании web-приложений на основании искусственных нейронных сетей**

Для решения задачи определения уровней категориальности скрытых ошибок с целью определения необходимости повторного системного тестирования web-приложений необходимо решить следующую систему уравнений:

$$\begin{cases} C_1 = f_1(x_1, x'_1), \\ C_2 = f_2(x_2, x'_2), \\ C_3 = f_3(x_3, x'_3), \\ C_4 = f_4(x_4, x'_4), \end{cases} \quad (1)$$

где  $x_i$  - операции основного тестирования,  $x'_i$  - типы обнаруженных во время основного тестирования ошибок.

Теорема Хехт-Нильсена [4] доказывает представимость функции многих переменных достаточно общего вида с помощью двухслойной нейронной сети с прямыми полными связями с  $n$  нейронами входного слоя и  $m$  нейронами выходного слоя с неизвестными функциями активации. Теорема в неконструктивной форме доказывает возможность решения задачи представления функции произвольного вида на нейронной сети и указывает для каждой задачи минимальные числа нейронов сети, необходимых для ее решения.

Из теоремы Хехт-Нильсена следует возможность представления любой многомерной функции нескольких переменных с помощью нейронной сети фиксированного размера.

Таким образом, нейронные сети являются универсальными структурами, позволяющими реализовать любой вычислительный алгоритм, следовательно, для реализации нелинейной функции нескольких переменных будут использоваться именно нейронные сети.

На рис. 2 изображена модель для поддержки принятия решений при повторном системном тестировании web-приложений, в основе которой лежит искусственная нейронная сеть типа прямонаправленного персептрона (данная модель построена на основе нейросетевой категориальной модели процесса повторного тестирования прикладного программного обеспечения [5]). Структура многослойного персептрона соединяется со структурой простого однослойного персептрона Розенблатта. В прямонаправленном многослойном персептроне нейроны располагаются в слоях, расположенных друг за другом, и взаимосвязи между нейронами слоев предусмотрены только в направлении от входа к выходу сети.

Выходы нейронов из каждого слоя многослойного персептрона связаны как с последующим слоем, так и с нейроном второго слоя персептрона Розенблатта.

Множество  $X = \{x_i\}$ ,  $i = (\overline{1, 2})$  - число нейронов рецепторного слоя, которые связаны с суммирующим нейроном  $S_1$  персептрона Розенблатта. Количество рецепторных входов этого слоя определено множеством  $X^x = \{x^{x_i}\}$ ,  $i = (\overline{1, 2})$ . На вход  $x^{x_1}$  будут подаваться номера таких операций системного тестирования, при выполнении которых зачастую выявляются ошибки первого уровня категориальности – косметические (например, орфографическая ошибка в тексте, неверное имя кнопки, неверные размеры графических элементов, некорректное цветовое оформление). На вход  $x^{x_2}$  будут подаваться номера типов ошибок 1-го уровня категориальности, которые были выявлены в результате основного системного тестирования.

Присвоение номеров операциям и ошибкам системного тестирования web-приложений будет приведено в таблицах базы знаний.

Множество  $Z = \{z_i\}$ ,  $i = (\overline{1,2})$  - число нейронов первого ассоциативного слоя персептрона. Нейроны этого слоя связаны с суммирующим нейроном  $S_2$ . Количество рецепторных входов этого слоя:  $X^z = \{x^{z^i}\}$ ,  $i = (\overline{1,2})$ . Вход  $x^{z^1}$  - номера операций системного тестирования, при выполнении которых выявляются ошибки второго уровня категориальности – умеренные (например, ошибка валидации, неверные значения выпадающего списка, неверные значения чекбоксов, время обработки запросов превышает указанное время в спецификации). Вход  $x^{z^2}$  - номера типов ошибок 2-го уровня категориальности.

Число нейронов второго ассоциативного слоя сети определено множеством  $M = \{m_i\}$ ,  $i = (\overline{1,2})$ . Нейроны этого слоя связаны с суммирующим нейроном  $S_3$ . Число рецепторных входов второго ассоциативного слоя определено множеством  $X^m = \{x^{m^i}\}$ ,  $i = (\overline{1,2})$ , где вход  $x^{m^1}$  - номера операций системного тестирования, которые приводят к выявлению ошибок третьего уровня категориальности – значительных (отсутствие линеек прокрутки, ошибки вычислений, ошибки при авторизации, данные теряются после сбоя без возможности восстановления); вход  $x^{m^2}$  - номера типов ошибок 3-го уровня категориальности.

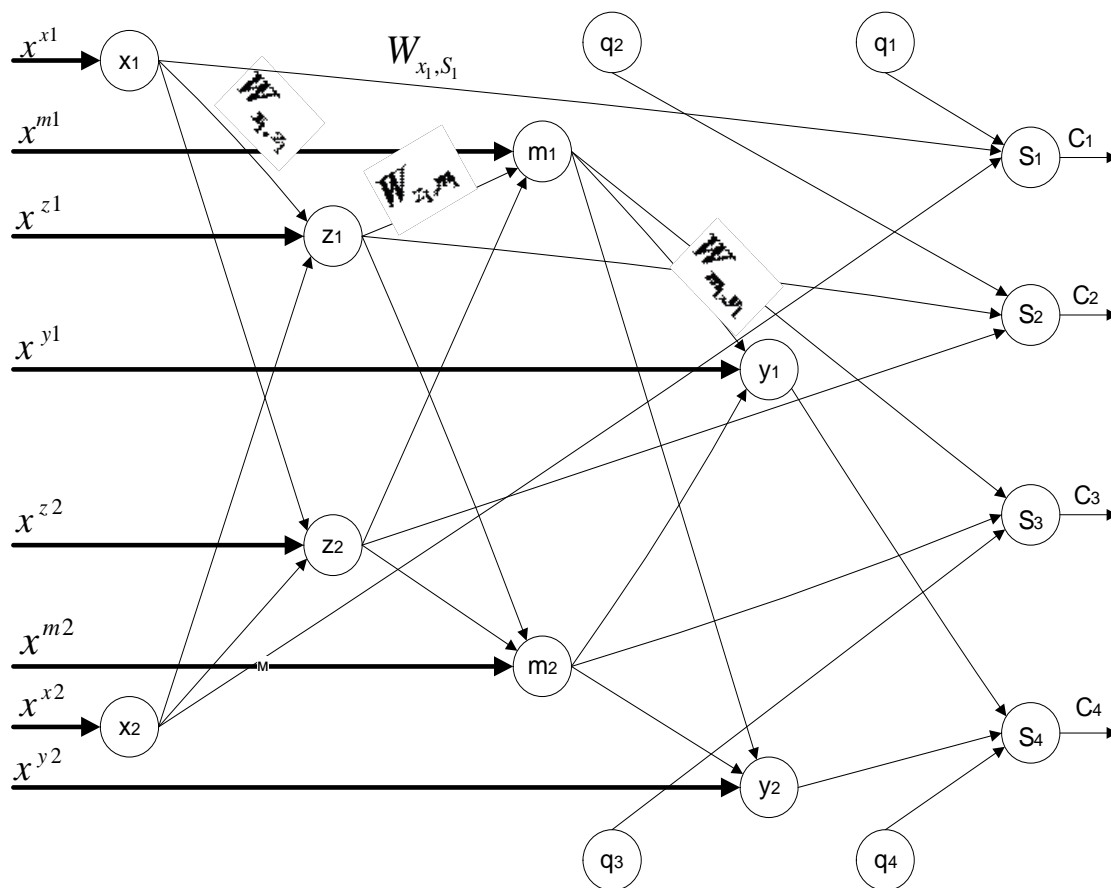


Рис. 2. Модель для поддержки принятия решений при повторном системном тестировании web-приложений

Множество  $Y = \{y_i\}$ ,  $i = (\overline{1,2})$  - число нейронов третьего ассоциативного слоя персептрона, нейроны которого связаны с суммирующим нейроном  $S_4$ . Множество  $X^y = \{x^{yi}\}$ ,  $i = (\overline{1,2})$  определяет количество рецепторных входов этого слоя. На вход  $x^{y1}$  будут подаваться номера операций системного тестирования, при выполнении которых выявляются ошибки четвертого уровня категориальности – критические (приложение не запускается, сервер базы данных недоступен, данные не сохраняются в базе данных). На вход  $x^{y2}$  будут подаваться номера типов ошибок 4-го уровня категориальности, выявленные во время основного системного тестирования.

Вектор пороговых величин смещения множества нейронных элементов определен как  $\bar{\theta} = \{\theta_h\}$ ,  $h = (\overline{1,4})$ ,  $\theta_h = q_h w_{q_h, S_h}$ , где  $q_h$  - смещение  $S_h$ -го нейронного элемента,  $w_{q_h, S_h}$  - весовой коэффициент смещения.

$C = \{C_i\}$ ,  $i = (\overline{1,4})$  - множество выходов суммирующего слоя персептрона Розенблатта. Выходы нейронов с каждого слоя многослойного персептрона связаны как с последующим слоем, так и с нейроном суммирующего слоя персептрона Розенблатта.

Учитывая это, функционал искусственной нейронной сети определится по формуле

$$C_1 = F_{S_1} \left( \sum_{i=1}^2 x_i w_{x_i} - q_1 w_{q_1} \right). \quad (2)$$

Для определения  $C_2$  необходимо учесть множество входных значений с их весовыми коэффициентами и множество исходных значений активности нейронов, а также смещения:

$$C_2 = F_{S_2} \left( \sum_{i=1}^2 \left( f(z_i) \cdot \left( \sum_{i=1}^2 x_i w_{x_i, z_i} + x^{z_i} w_{x^{z_i}} \right) \right) \cdot w_{z_i, S_2} - q_2 w_{q_2, S_2} \right), \quad (3)$$

где  $f(z_i)$  - функция активации нейронов скрытого слоя Z (функция гиперболического тангенса);  $x_i$  - выходные значения рецепторного слоя сети, нейроны которого соединены со скрытым слоем Z;  $w_{x_i, z_i}$  - весовые коэффициенты связей нейронов слоя X, которые связаны с нейронами слоя Z;  $x^{z_i}$  - входы в слой Z;  $w_{x^{z_i}}$  - весовые коэффициенты входов, которые связаны со скрытым слоем Z.

Аналогично определению  $C_2$  определим функционалы  $C_3$  и  $C_4$ .

Каждый из выходов  $C_i$  отвечает за один из четырех уровней категориальности и принимает значение "1", если искусственной нейронной сетью спрогнозировано наличие в программе ошибки i-го уровня категориальности, в противном случае значение выхода составляет "0".

Сравнивая полученные результаты с порогом допустимого количества ошибок определенного уровня категориальности, система делает вывод о том, необходимо ли проводить повторное системное тестирование web-приложения и какие операции системного тестирования при этом применять.

#### 4. Заключение

Задача поддержки принятия решений при повторном системном тестировании web-приложений будет реализована на основе метода искусственных нейронных сетей, т.к. они являются универсальными структурами, позволяющими реализовать любой вычислительный алгоритм.

Выбор аппарата искусственных нейронных сетей мотивирован тем, что за счет возможности аппроксимаций нелинейных функций искусственные нейронные сети дают возможность учитывать пороги предельного количества допустимых ошибок каждой категории, влияние одной категории ошибок на другую. Определение исходного функционала каждого из слоев нейронной сети, которые соответствуют определенной категории ошибок, дает возможность сделать выводы о необходимости повторного системного тестирования web-приложения.

#### Список литературы

1. Соммервил, И. Инженерия программного обеспечения [Текст] : пер.с англ. - 6-е изд. / И. Соммервил. – М.: Изд. дом “Вильямс”, 2002. – 624 с.
2. Орлов, С. Технологии разработки программного обеспечения [Текст] / С. Орлов. – СПб.: Питер, 2002. – 464 с.
3. Савин, Р. Тестирование Dot Com, или Пособие по жестокому обращению с багами в интернет-стартапах [Текст] / Р. Савин — М.: Дело, 2007. — 312 с.
4. Круглов, В.В. Искусственные нейронные сети. Теория и практика [Текст] / В.В Круглов, В.В. Борисов. – М.: Горячая линия – Телеком, 2001. – 382 с.
5. Локазюк, В.М. Категорiальна модель процесу повторного тестування дефектiв програмного забезпечення [Текст] / В.М. Локазюк, Т.О. Пантелеева // Вісник Технологiчного унiверситету Подiлля. – Хмельницький ТУП. – Х., 2004. – Ч. 1. – С. 53-58.

**Рецензент:** д.т.н., доц., проф. И. В. Шостак, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», г. Харьков.

Поступила в редакцию 10.11.11

### **Штучні нейронні мережі в задачах підтримки прийняття рішень при повторному системному тестуванні web-додатків**

Подано обґрунтування необхідності проведення повторного системного тестування web-додатків. Обґрунтовано вибір методу реалізації технології підтримки прийняття рішень при повторному системному тестуванні web-додатків. Описано методику виявлення помилок за допомогою повторного системного тестування web-додатків. Наведено модель для підтримки прийняття рішень при повторному системному тестуванні web-додатків.

**Ключові слова:** веб-додаток, основне системне тестування, повторне системне тестування, підтримка прийняття рішень, штучні нейронні мережі.

## **Artificial neural networks in problems of decision's support during system re-testing of web-based applications**

Justification for a system re-testing of the web-based applications. The choice of technology implementation method has been justified for decision's support during system re-testing of web-based applications. The method of error detection by system re-testing of the web-based applications has been described. A model for decision's support during system re-testing of web-based applications has been presented.

**Keywords:** web-application, main system testing, system retesting, decision's support, artificial neural networks.