

УДК 658.511.3

КОНЦЕПТУАЛЬНОЕ МОДЕЛИРОВАНИЕ ПРОЦЕССА СОПРОВОЖДЕНИЯ И РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРИСВОЕНИЯ ПРИОРИТЕТА ЗАЯВОК НЕЙРОННОЙ СЕТЬЮ

Мишин Артём Александрович, магистрант, направление подготовки 27.04.03 Системный анализ и управление, Оренбургский государственный университет, Оренбург
e-mail: artmisch@mail.ru

Пищухина Татьяна Александровна, кандидат технических наук, доцент, доцент кафедры управления и информатики в технических системах, Оренбургский государственный университет, Оренбург
e-mail: tpuits@mail.ru

***Аннотация.** В данной статье поднимается тема построения концептуальных моделей процесса сопровождения и создания программного обеспечения на основе нейронных сетей Кохонена для присвоения приоритета заявкам, поступившим в работу. Сопровождение систем автоматизации программного обеспечения определяется как вся совокупность деятельности, необходимой для обеспечения эффективной поддержки программных систем [8].*

Одной из основных проблем при сопровождении программного обеспечения является проблема присвоения приоритета заявкам, что приводит к уменьшению надежности программного обеспечения.

Построена диаграмма «модель-сущность», на которой показан процесс работы специалиста сопровождения от поступления заявки до ее решения. На диаграмме моделей данных продемонстрирован процесс присвоения приоритета. Рассмотрены процессы обучения нейронной сети и присвоения приоритета заявке.

***Ключевые слова:** концептуальное моделирование, сопровождение, нейронная сеть, присвоение приоритета.*

***Для цитирования:** Мишин А. А., Пищухина Т. А. Концептуальное моделирование процесса сопровождения и разработка программного обеспечения присвоения приоритета заявок нейронной сетью // Шаг в науку. – 2022. – № 4. – С. 55–59.*

CONCEPTUAL MODELING OF THE MAINTENANCE PROCESS AND DEVELOPMENT OF SOFTWARE FOR ASSIGNING THE PRIORITY OF APPLICATIONS BY A NEURAL NETWORK

Mishin Artem Aleksandrovich, postgraduate student, training program 27.04.03 System analysis and Management, Orenburg State University, Orenburg
email: artmisch@mail.ru

Pishukhina Tatiana Aleksandrovna, Candidate of Technical Sciences, Associate Professor, Associate Professor of the Department of Management and Computer Science in Technical Systems, Orenburg State University, Orenburg
email: tpuits@mail.ru

***Abstract.** This report raises the topic of building conceptual models of the maintenance process and creating software based on Kohonen neural networks to assign priority to applications received for work. Maintenance of software automation systems is defined as the whole set of activities necessary to ensure effective support of software systems [7].*

One of the main problems in software maintenance is the problem of assigning priority to applications, which leads to a decrease in the reliability of the software.

The diagram “model-entity” is constructed, which shows the process of work of the support specialist from the receipt of the application to its solution. The data model diagram shows the process of assigning priority. The processes of training a neural network and assigning priority to an application are considered.

***Key words:** conceptual modeling, maintenance, neural network, priority assignment.*

***Cite as:** Mishin, A. A., Pishukhina, T. A. (2022) [Conceptual modeling of the maintenance process and development of software for assigning the priority of applications by a neural network]. *Shag v nauku* [Step into science]. Vol. 4, pp. 55–59.*

Процесс сопровождения состоит из работ и задач, необходимых для модификации программного продукта при сохранении его целостности¹. От эффективности работ на этапе поддержки и сопровождения программного обеспечения зависит непрерывность бизнес-процессов фирмы и сохранность информации, необходимой для выполнения ее стратегических, тактических и оперативных задач [5]. В организациях, где около 100 компаний используют их ПО, в отдел технической поддержки поступает большое количество заявок. Среднее число заявок составляет несколько тысяч. В ходе сопровождения ПО возникает необходимость вносить изменения, исправлять недоработки и дефекты, представляющие собой ошибки в коде или недоработки в интерфейсе, добавлять новый функционал и многое другое с целью повысить качество ПО [2]. Одновременная обработка такого количества запросов невозможна, становится трудно оптимизировать поступающие задания и определять, какое из них является наиболее важным и как обслужить наибольшую часть из них в короткие сроки.

Поставленному вопросу отвечает присвоение приоритета и сортировка заявок, которые можно провести при помощи нейронной сети.

Нейронные сети используются для распознавания, анализа данных и прогнозирования. Они обрабатывают входные данные и заполняют список заявок на основе доступной информации, в зависимости от того, какие заявки должны быть обработаны в первую очередь. Чтобы прописать эти действия в нейронной сети, необходимо рассмотреть технологический процесс, осуществляемый специалистом сопровождения программного обеспечения.

Рассмотрим концептуальное моделирование процессов сопровождения программного обеспечения: отдел технической помощи занимается решением инцидентов, консультированием клиентов и составлением отчетов.

Специалист технической помощи должен иметь представление, как функционирует ПО, взаимодействие с ним клиентов и основные ошибки, возникающие при его использовании [4].

На рисунке 1 показана ER-модель отдела технической помощи.

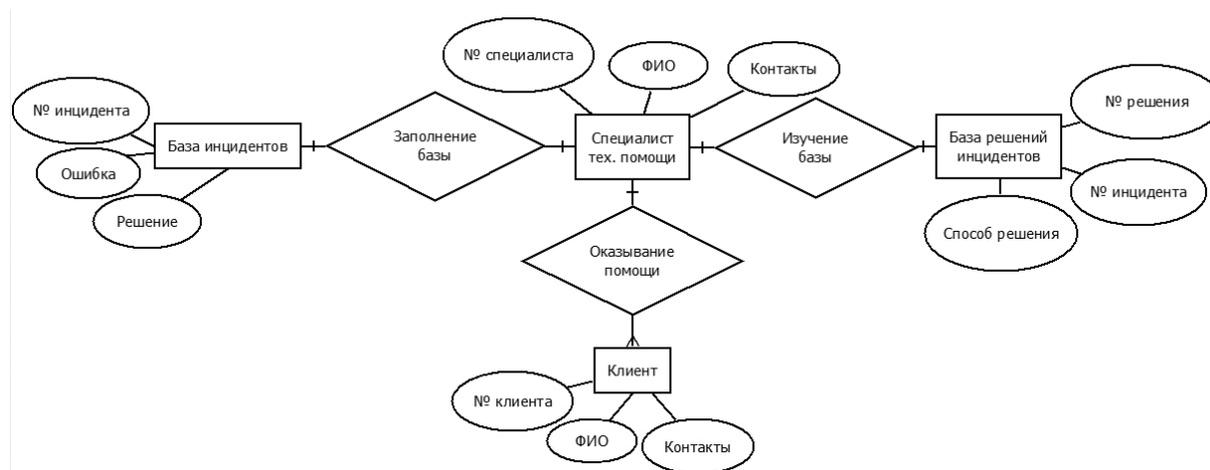


Рисунок 1. Диаграмма «модель-сущность» отдела технической помощи
 Источник: разработано автором Мишиным А. А.

На основе диаграммы «модель-сущность» строится диаграмма моделей данных процесса обработки заявки, что показано на рисунке 2.

Чтобы нейронная сеть могла распознавать и сортировать заявки, необходимо создать библиотеку, с помощью которой будет происходить классификация заявок, определение времени выполнения, сложности заявки [3].

При проектировании библиотеки необходимы: выборка заявок, унификация слов при составлении списка, то есть использование механизма N-грамм,

удаление всех стоп-слов, цифр, знаков пунктуации, а также сокращение библиотеки с помощью стемминга, то есть приведения слов к их основам. Последний этап – ликвидация повторений.

Далее строится нейронная сеть Кохонена для кластеризации каждой заявки по классу проблемы. Сети Кохонена относятся к самоорганизующимся нейронным сетям. Самоорганизующаяся сеть позволяет выявлять кластеры входных векторов, обладающих некоторыми общими свойствами [1].

¹ ГОСТ Р ИСО/МЭК 14764–2002 Информационная технология (ИТ). Сопровождение программных средств. – Введ. 2003–07–01. – Москва: Стандартинформ, 2002. – 32 с.

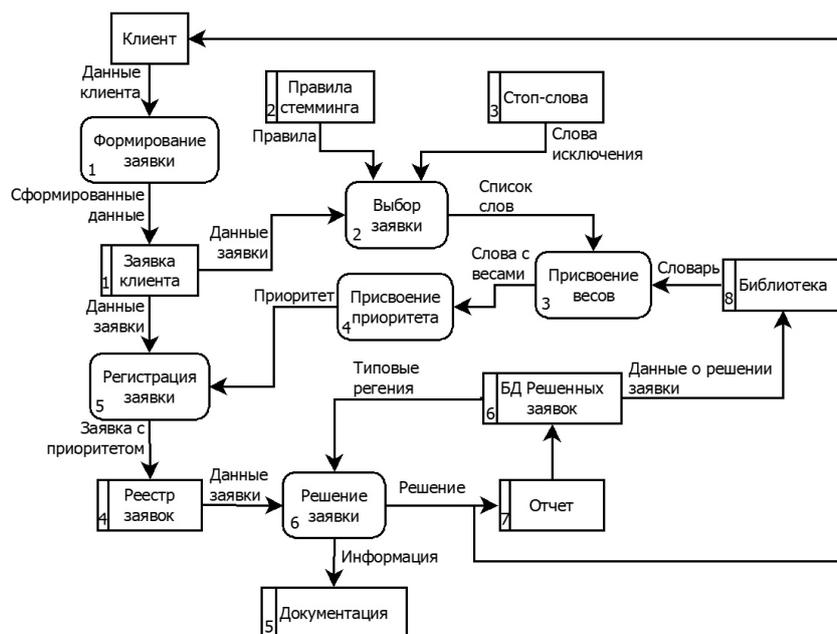


Рисунок 2. Диаграмма моделей данных процесса обработки заявки
 Источник: разработано авторами

Для адекватной работы нейронной сети проводится процесс ее обучения. Входные значения получают путем разбивки заявки на унифицированные слова.

Каждому слову из библиотеки присваиваются весовые коэффициенты. Общее количество весовых коэффициентов рассчитывается как произведение:

$$N_w = MK, \tag{1}$$

где

- M – количество входных переменных сети;
- K – количество кластеров (классов ошибок).

Библиотека представляет из себя матрицу, где строки – это количество кластеров (K), а входные переменные (M) являются столбцами.

Весовой коэффициент устанавливается для каждого слова отдельно во время обучения нейронной сети.

При присвоении веса заявке используют сумму весовых коэффициентов слов, используемые в заявке. Кластер, набравший максимальный вес (W_K),

становится победителем.

На основе важности каждой заявки устанавливается коэффициент для времени ожидания каждого критерия заявки.

Важность заявки в зависимости от описанных в ней ошибок, влияющих на технологический процесс, и коэффициенты времени ожидания (k) показаны в таблице 1.

Таблица 1. Входные значения с точки зрения технического процесса

Критичность технической реализации			
Низкий	Средний	Высокий	Очень высокий
1. Консультационный вопрос по эксплуатации программного обеспечения или запрос по расширению функциональности программного обеспечения ($k = 1$)	2. Приводит к незначительному нарушению исполнения технологических процессов и допускает равнозначное «обходное» решение ($k = 1,3$)	3. Приводит к нарушению исполнения технологических процессов, с возможностью временного или постоянного альтернативного решения ($k = 1,6$)	4. Приводит к остановке технологических процессов или существенно меняет их ($k = 1,9$)

Источник: разработано автором Мишиным А. А. на основе [6]

Критичность технической реализации устанавливается для каждой заявки в зависимости от присвоенного кластера.

Время ожидания находится по формулам, применяемым в системах массового обслуживания (СМО).

Следующие параметры используются для того, чтобы характеризовать эффективность работы

системы:

n – число каналов;

λ – интенсивность поступления заявок;

μ – интенсивность обслуживания заявок;

ρ – коэффициент загрузки.

Коэффициент загрузки находится по следующей формуле:

$$\rho = \frac{\lambda}{\mu}. \quad (2)$$

Вероятность того, что система уже занята, находится по следующей формуле:

$$P_0 = \left(1 + \frac{\rho}{1!} + \frac{\rho^2}{2!} + \dots + \frac{\rho^{n-1}}{(n-1)!} + \frac{\rho^n}{(n-1)!} \cdot \frac{1}{n-\rho} \right)^{-1}. \quad (3)$$

Используя формулу занятости системы, можно определить среднее число заявок в очереди:

$$L_{оч} = \frac{\rho^{n+1}}{n!} \cdot \frac{n}{(n-\rho)^2} \cdot P_0. \quad (4)$$

Время ожидания заявки в очереди находится по формуле:

$$T_0 = \frac{L_{оч}}{\lambda} \cdot k \quad (5)$$

Время выполнения является среднестатистическим значением времени, затраченного на решение кластера.

Присвоенный приоритет представляет собой

список заявок, сортированных по весам. Каждая заявка получает вес, который высчитывается, как произведение среднего времени выполнения, критичности и времени ожидания:

$$w_3 = \frac{1}{T_0} T_W w_K, \quad (6)$$

где

T_W – время выполнения;

w_K – весовой коэффициент кластера.

Новая заявка дополняет список, который выстраивается в порядке убывания, где наивысшее значение – это максимальный приоритет.

Разработанная концептуальная модель и модель присвоения приоритета позволят спроектировать нейронную сеть для классификации и назначения приоритета заявкам.

Нейронная сеть сама разбирает описание проблемы, сравнивает со списком заявок, которые стоят в очереди и на основе этих данных присваивает приоритет. Это должно позволить полностью освободить от присвоения приоритета заявкам специалистов и освободить их от монотонного и зачастую очень трудоемкого процесса.

Литература

1. Кохонен Т. Самоорганизующиеся карты. – М.: Бинум. Лаборатория знаний, 2008. – 655 с.
2. Мишин А. А. Приоритизация в процессе сопровождения программного обеспечения // Наука – практике: материалы I Междунар. науч.-практ. конф., Барановичи, 15 мая 2020 г.: в 2 ч. – Барановичи: БарГУ, 2020. – Ч. 1. – С. 39–42.
3. Мишин А. А., Пищухина Т. А. Назначение приоритетов заявкам в процессе сопровождения ПО на основе технологии нейронных сетей // Школа-семинар молодых ученых и специалистов в области компьютерной интеграции производства: сборник материалов. – Оренбург: Оренбургский государственный университет, 2020. – С. 122–125.
4. Мишин А. А., Пищухина Т. А. Системный анализ факторов повышения надежности программного обеспечения // Компьютерная интеграция производства и ИПИ-технологии: сборник материалов X Всероссийской конференции, Оренбург, 18-19 ноября 2021 г. – Оренбург: Оренбургский государственный университет, 2021. – С. 361–364.
5. Мусина К. Р. Автоматизация процессов сопровождения программных продуктов на базе передовых стандартов // Научный форум: Технические и физико-математические науки: Сборник статей по материалам XVI международной научно-практической конференции, Москва, 05–15 июля 2018 года. – Москва:

Общество с ограниченной ответственностью «Международный центр науки и образования», 2018. – С. 20–24.

6. Регламент работы службы технической поддержки (СТП)– 2019. – URL: https://uralsib.itfinance.io/media/filer_public/a1/b9/a1b99214-3cdb-4a9f-92bd-18b6f60b5f68/reglament_tsp_ooo_serviskhab_.docx (дата обращения: 26.04.2022).

7. Сычугов А. А., Токарев В. Л., Анчишкин А. П. Обнаружение сетевых атак на основе искусственных иммунных систем // Известия Тульского государственного университета. Технические науки. – 2018. – № 10. – С. 117–124.

8. Хан В. Л., Кизим А. В. Сопровождение систем автоматизации программного обеспечения // Молодой ученый. – 2011. – № 5–1. – С. 110–112.

Статья поступила в редакцию: 24.05.2022; принята в печать: 25.10.2022.

Авторы прочитали и одобрили окончательный вариант рукописи.