

## МОДЕЛЮВАННЯ ПРОЦЕСУ ОЦІНЮВАННЯ ДОСТАТНОСТІ ІНФОРМАЦІЇ ЩОДО ЯКОСТІ У СПЕЦИФІКАЦІЯХ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

*Метою даного дослідження є розроблення моделей процесу оцінювання достатності інформації для визначення якості програмного забезпечення на основі стандарту ISO 25010:2011, моделей процесу оцінювання достатності інформації для визначення якості та складності програмного забезпечення з використанням результатів метричного аналізу, моделі процесу оцінювання результатів проектування та прогнозування характеристик програмного забезпечення на основі штучної нейронної мережі. Розроблені моделі дають можливість відобразити особливості процесу оцінювання достатності інформації щодо якості у специфікації вимог до ПЗ, формалізувати процес оцінювання достатності інформації щодо якості у специфікації вимог до ПЗ, а також забезпечують теоретичне підґрунтя для розроблення методів та засобів оцінювання достатності інформації щодо якості у специфікаціях вимог до ПЗ.*

*Ключові слова: програмне забезпечення (ПЗ), специфікація вимог до ПЗ, якість програмного забезпечення, достатність інформації, онтологія, штучна нейронна мережа (ШНМ).*

T. O. HOVORUSHCHENKO, O. V. POMOROVA

Khmelnyskyi National University

## MODELLING OF THE PROCESS OF EVALUATING THE SUFFICIENCY OF THE INFORMATION ON QUALITY IN THE SOFTWARE REQUIREMENTS SPECIFICATIONS

*The aim of this study is the development of: the models of the process of evaluating the sufficiency of information for the software quality assessment on the basis of ISO 25010: 2011, the models of the process of evaluating the sufficiency of information for the software quality and complexity assessment using the metric analysis results, the model of the process of evaluating the design results and predicting the software characteristics based on artificial neural network. The developed models models of the software quality by ISO 25010 and the models of the software quality and complexity using metric analysis results based on ontologies and weighted ontologies provide the modelling the process of evaluating the sufficiency of the information on quality in the software requirements specifications. The developed information models of the process of evaluating the sufficiency of the information for determining the software quality by ISO 25010:2011 and for determining the software quality and complexity using the results of metric analysis reflect the peculiarities of the process of evaluating the sufficiency of the information on quality in the software requirements specifications of, and the relationship between software quality models based on ISO 25010 on the basis of ontologies, between software quality and complexity models using the metric analysis results based on ontologies. The developed models of the process of evaluating the sufficiency of the information for determining the software quality by ISO 25010:2011 and for determining the software quality and complexity using metric analysis results formalize the process of evaluating the sufficiency of the information on quality in the software requirements specifications and provide the theoretical basis for the development of the methodology of evaluating the sufficiency of the information on quality in the software requirements specifications. The proposed concept of evaluating the design results and predicting the software characteristics provided the possibility to develop the model of the process of evaluating the design results and predicting the software characteristics based on artificial neural network. The developed model of the process of evaluating the design results and predicting the software characteristics based on ANN differs from the known models because it provides to take into account the weights and interactions of the quality and complexity software metrics within each characteristic of the software, and to estimate the total impact of metrics on the software characteristics (software project quality and complexity, predicted quality and complexity of software).*

*Keywords: software, software requirements specification (SRS), software quality, sufficiency of information, ontology, artificial neural network (ANN).*

### Вступ

Практично усі сфери людської діяльності на сьогодні пов'язані з інформаційними системами, основою яких є програмне забезпечення (ПЗ). Ключовим фактором забезпечення ефективного застосування програмних продуктів та однією із основних потреб замовників є досягнення високих значень показників якості ПЗ.

Як показує статистика [1–5], наразі існують проблеми у галузі забезпечення якості ПЗ. Причому велика кількість помилок ПЗ виникає на етапі формування та формулювання вимог – такі помилки становлять 10–23% всіх помилок. Переважна більшість аварій, пов'язаних із ПЗ, виникли через помилкові вимоги, а не через помилки кодування [5]. Чим раніше буде виявлено відповідний дефект (помилка, порушення, недолік, несправність), тим дешевше обійдеться його виправлення [6].

В процесі формування та формулювання вимог відбуваються інформаційні втрати через неповне та різне розуміння потреб та контексту інформації – особливо такі втрати суттєві для програмних проектів, які розробляються на стику предметних галузей (наприклад, ПЗ для медицини), коли враховувати потрібно як стандарти щодо розроблення ПЗ, так і стандарти предметної галузі, для якої розробляється ПЗ. Таку кількість стандартів важко імплементувати, а ще важче верифікувати ступінь врахування рекомендацій цих стандартів.

Програмні проекти з неповними вимогами та специфікаціями не можуть мати успішної реалізації [7]. За таких умов актуальним і дуже важливим є аналіз специфікації вимог до ПЗ, можливість «відсікти» програмні проекти з неповною (з недостатньою інформацією) специфікацією. Достатність інформації є

одним з найважливіших аспектів оцінювання якості програмного забезпечення. Якість та успішність реалізації програмного проекту суттєво залежать від специфікації вимог, а також від достатності наявної у ній інформації як наявності всіх інформаційних елементів (даних), необхідних для визначення якості ПЗ. Неповнота, неточність та спотворення інформації специфікації вимог до ПЗ призводять, відповідно, до зменшення достовірності оцінок якості ПЗ, а також до збільшення прогалини в знаннях про програмне забезпечення, що спричиняє виникнення непрогнозованих емерджентних властивостей програмних систем.

Сьогодні оцінювання якості ПЗ за стандартом ISO 25010:2011 [8] відбувається так: на основі атрибутів якості, зазначених у ISO 25023:2016 [9], оцінюються підхарактеристики та характеристики якості, які, в свою чергу, надають комплексну оцінку якості ПЗ. Оцінювання ж якості та складності ПЗ на основі використання результатів метричного аналізу відбувається так: на основі показників розраховуються значення метрик якості та складності, які, в свою чергу, надають комплексну оцінку якості та складності ПЗ [10]. Тоді атрибути якості, а також показники якості та складності ПЗ, визначені у специфікації вимог, становлять *інформацію щодо якості* специфікації вимог до ПЗ. Під *достатністю інформації щодо якості* у специфікації вимог до ПЗ будемо розуміти наявність у специфікації всіх інформаційних елементів (атрибутів та показників), необхідних для визначення якості ПЗ. Якщо деякі атрибути та (або) показники відсутні, то у специфікації вимог недостатньо інформації для визначення якості ПЗ, і розробники повинні внести необхідні доповнення у специфікацію.

Тоді *актуальною задачею* є задача оцінювання достатності інформації щодо якості у специфікації вимог до ПЗ (як наявності атрибутів та (або) показників якості ПЗ у специфікації вимог).

Проведений аналіз стандартів ISO 25010 [8], ISO 25023 [9] показав, що вони представлені природною мовою у текстовій формі, тому відсутній механізм верифікації результатів імплементації цих стандартів у процес розроблення ПЗ. Було встановлено, що інформацію щодо якості зручно подавати у вигляді онтологій, що дозволяють відобразити причинно-наслідкові зв'язки між поняттями.

Аналіз відомих моделей в галузі інженерії інформаційних систем та програмного забезпечення показав, що наразі розроблено моделі профілю при сертифікації ПЗ [11], онтологічні моделі інтелектуальних систем підтримки прийняття рішень [12], моделі для єдиного когерентного підґрунтя стандартів ISO/IEC JTC1/SC7 [13, 14], моделі доменної онтології для ISO 24744 [15] і для аналізу ПЗ та засобів реінжинірингу [16], модель для опису та визначення предметних і операційних знань щодо забезпечення якості ПЗ [17]. Але наразі відсутні моделі процесу оцінювання достатності інформації для визначення якості програмного забезпечення, а також моделі процесу оцінювання результатів проектування та прогнозування характеристик ПЗ.

Тому *метою даного дослідження* є розроблення: моделей процесу оцінювання достатності інформації для визначення якості програмного забезпечення на основі стандарту ISO 25010:2011, моделей процесу оцінювання достатності інформації для визначення якості та складності програмного забезпечення з використанням результатів метричного аналізу, моделі процесу оцінювання результатів проектування та прогнозування характеристик програмного забезпечення на основі штучної нейронної мережі.

#### **Моделі процесу оцінювання достатності інформації для визначення якості програмного забезпечення на основі стандарту ISO 25010:2011**

Якість ПЗ ( $Q$ ), згідно зі стандартом [8], є функцією від восьми основних характеристик якості. Тоді множина характеристик якості ПЗ:  $QCH = \{qch_1, \dots, qch_8\}$ . Кожна з основних характеристик якості є функцією від декількох підхарактеристик якості. Множина підхарактеристик якості ПЗ має вигляд:  $QSCH = \{qsch_1, \dots, qsch_{31}\}$ . Водночас, кожна підхарактеристика якості ПЗ є функцією певних атрибутів якості ПЗ, описаних у стандарті [9]. Множина атрибутів якості ПЗ, згідно зі стандартом [9], має вигляд  $QMS = \{qms_1, \dots, qms_{138}\}$ , оскільки аналіз атрибутів дав можливість визначити залежність підхарактеристик якості від 138 різних атрибутів (всього від 203 атрибутів). Тоді якість ПЗ може бути виражена не тільки функцією від підхарактеристик якості, але й середнім зваженим арифметичним показником з урахуванням значень атрибутів та коефіцієнтів їхньої вагомості.

Отже, необхідним є обчислення вагових коефіцієнтів атрибутів якості ПЗ. Оскільки характеристики та підхарактеристики якості ПЗ корелюють за атрибутами, то при обчисленні вагових коефіцієнтів атрибутів варто враховувати існування таких взаємозв'язків. Цінними при виявленні спільних атрибутів є знання досвідчених фахівців щодо взаємовпливу та кореляції характеристик, підхарактеристик та атрибутів якості ПЗ, тому їх варто зберігати та використовувати. У якості засобу відображення цих знань обрано онтології.

Модель якості ПЗ за стандартом ISO 25010:2011 на основі онтології має вигляд:  $O_Q = \langle X_Q, RX_Q, F_Q \rangle$ , де  $X_Q$  – скінченна множина характеристик, підхарактеристик та атрибутів якості ПЗ,  $RX_Q$  – скінченна множина відношень між поняттями,  $F_Q$  – скінченна множина функцій інтерпретації, заданих для характеристик, підхарактеристик та атрибутів якості ПЗ.

Множина характеристик, підхарактеристик та атрибутів якості ПЗ:  $X_Q = \{QCH, QSCH, QMS\} = \{x_Q^1, \dots, x_Q^{177}\}$ , де:  $\{x_Q^1, \dots, x_Q^8\} \in QCH$ ,  $\{x_Q^9, \dots, x_Q^{39}\} \in QSCH$ ,  $\{x_Q^{40}, \dots, x_Q^{177}\} \in QMS$ . Множина відношень між поняттями  $RX_Q$  складається з відношення «залежить від»,

тобто  $RX_Q = \{ \text{"depends on"} \}$ . Множина  $F_Q$  функцій інтерпретації, заданих для характеристик, підхарактеристик та атрибутів якості ПЗ, складається з функцій залежності якості від характеристик, характеристик від підхарактеристик, підхарактеристик від атрибутів якості, тобто  $F_Q = \{ f_Q^1, \dots, f_Q^{40} \} = \{ f(), f_1(), \dots, f_8(), \phi_1(), \dots, \phi_{31}() \}$ .

Тоді базова (універсальна) модель предметної галузі «Інженерія програмного забезпечення» (частина «Якість ПЗ») на основі онтології має вигляд [18, 19]:

$$O_Q = \{ qch_1, \dots, qch_8, qsch_1, \dots, qsch_{31}, qms_1, \dots, qms_{138}, \text{"depends on"}, f(), f_1(), \dots, f_8(), \phi_1(), \dots, \phi_{31}() \}. \quad (1)$$

Оскільки необхідним є врахування вагових коефіцієнтів атрибутів якості ПЗ, то є потреба внесення таких вагових коефіцієнтів у базову (універсальну) модель предметної галузі «Інженерія програмного забезпечення» (частина «Якість ПЗ») та побудови зваженої онтології.

Тоді множина характеристик, підхарактеристик та атрибутів якості ПЗ для моделі якості ПЗ за стандартом ISO 25010:2011 на основі зваженої онтології матиме наступний вигляд:  $X_{Q^w} = \{ x_{Q^w}^1, \dots, x_{Q^w}^{177} \}$ , де:

$\{ x_{Q^w}^1, \dots, x_{Q^w}^8 \} \in QCH$ ,  $\{ x_{Q^w}^9, \dots, x_{Q^w}^{39} \} \in QSCH$ ,  $\{ x_{Q^w}^{40}, \dots, x_{Q^w}^{177} \} = \{ (qms_1, w_1^m), \dots, (qms_{138}, \dots, w_{138}^m) \}$ , тобто підмножина  $\{ x_{Q^w}^{40}, \dots, x_{Q^w}^{177} \}$  складається з пар  $(qms_i, w_i^m)$ , де  $qms_i$  –  $i$ -й атрибут якості ПЗ ( $qms_i \in QMS$ ),  $w_i^m$  – ваговий коефіцієнт  $i$ -го атрибуту якості ПЗ,  $i = 1, 138$ .

Базова (універсальна) модель предметної галузі «Інженерія програмного забезпечення» (частина «Якість ПЗ») на основі зваженої онтології матиме наступний вигляд:

$$O_Q^w = \left\{ \begin{array}{l} qch_1, \dots, qch_8, qsch_1, \dots, qsch_{31}, (qms_1, w_1^m), \dots, (qms_{138}, \dots, w_{138}^m), \\ \text{"depends on"}, f(), f_1(), \dots, f_8(), \phi_1(), \dots, \phi_{31}() \end{array} \right\}. \quad (2)$$

Модель якості конкретного програмного продукту на основі онтології має вигляд [18, 19] (ця онтологія наповнюватиметься на основі інформації, взятої зі специфікації вимог до конкретного ПЗ):

$$O_Q^{real} = \{ qch_1, \dots, qch_{nch}, qsch_1, \dots, qsch_{nsch}, qms_1, \dots, qms_{nm}, \text{"depends on"}, f(), f_1(), \dots, f_8(), \phi_1(), \dots, \phi_{31}() \}, \quad (3)$$

де  $nch$  ( $nch \leq 8$ ) – кількість характеристик якості ПЗ, які можна обчислити на основі наявних у специфікації вимог до конкретного ПЗ атрибутів якості,  $nsch$  ( $nsch \leq 31$ ) – кількість підхарактеристик якості ПЗ, які можна обчислити на основі наявних у специфікації вимог до конкретного ПЗ атрибутів якості,  $nm$  ( $nm \leq 138$ ) – кількість атрибутів якості, наявних у специфікації вимог до конкретного ПЗ. Онтологія (онтологічна база знань), яка розробляється за моделлю (3), наповнюється на основі інформації, взятої зі специфікації вимог до конкретного ПЗ.

Враховуючи розроблені базову (універсальну) модель предметної галузі «Інженерія програмного забезпечення» (частина «Якість ПЗ») на основі онтології та на основі зваженої онтології, базову (універсальну) модель предметної галузі «Інженерія програмного забезпечення» (частина «Якість ПЗ») на основі зваженої онтології, а також модель якості конкретного програмного забезпечення на основі онтології, представлені формулами (1)-(3), розробимо моделі процесу оцінювання достатності інформації для визначення якості ПЗ на основі стандарту ISO 25010:2011.

Процес оцінювання достатності інформації для визначення якості ПЗ полягає у:

1) порівнянні онтології для визначення якості конкретного ПЗ з базовою (універсальною) онтологією з метою виявлення атрибутів, відсутніх в онтології для визначення якості конкретного ПЗ, тобто відсутніх у специфікації вимог до ПЗ, за якою ця онтологія була побудована, а також виявлення характеристик та підхарактеристик якості ПЗ, які неможливо обчислити на основі наявних у специфікації вимог до конкретного ПЗ атрибутів;

2) формуванні висновку про недостатність інформації щодо якості у специфікації вимог до конкретного ПЗ, якщо встановлено атрибути, відсутні у специфікації вимог до ПЗ, а також характеристики та підхарактеристики якості ПЗ, які неможливо обчислити на основі наявних атрибутів у специфікації вимог до конкретного ПЗ;

3) порівнянні онтології для визначення якості конкретного ПЗ зі зваженою базовою (універсальною) онтологією, якщо було сформовано висновок про недостатність інформації, – з метою встановлення вагових коефіцієнтів атрибутів, відсутніх у специфікації вимог до конкретного ПЗ, для подальшого визначення пріоритетності доповнення атрибутів у специфікації вимог до ПЗ.

Тоді модель процесу оцінювання достатності інформації для визначення якості ПЗ за стандартом ISO 25010:2011 на основі онтології подамо у вигляді:

$$\begin{cases}
 SMIQ = O_Q \setminus O_Q^{real}; \\
 \text{if } \{qch_1, \dots, qch_{(8-nch)}, qsch_1, \dots, qsch_{(31-nsch)}, qms_1, \dots, qms_{(138-nm)}\} = \emptyset \\
 \text{then "SRS information is sufficient"} \\
 \text{else: "SRS information is insufficient"}, \\
 SMMW = O_Q^w \setminus O_Q^{real}
 \end{cases}
 \quad (4)$$

де  $SMIQ = \{qch_1, \dots, qch_{(8-nch)}, qsch_1, \dots, qsch_{(31-nsch)}, qms_1, \dots, qms_{(138-nm)}\}$ ,  $\{qch_1, \dots, qch_{(8-nch)}\}$  – підмножина характеристик якості ПЗ, які неможливо обчислити на основі наявних у специфікації вимог до конкретного ПЗ атрибутів якості;  $\{qsch_1, \dots, qsch_{(31-nsch)}\}$  – підмножина підхарактеристик якості ПЗ, які неможливо обчислити на основі наявних у специфікації вимог до конкретного ПЗ атрибутів якості;  $\{qms_1, \dots, qms_{(138-nm)}\}$  – підмножина атрибутів, відсутніх у специфікації вимог до конкретного ПЗ;  $SMMW = \{(qms_1, w_1^m), \dots, (qms_{(138-nm)}, w_{(138-nm)}^m)\}$  – підмножина атрибутів, відсутніх у специфікації вимог до конкретного ПЗ, та їхніх вагових коефіцієнтів.

Інформаційна модель процесу оцінювання достатності інформації для визначення якості ПЗ на основі стандарту ISO 25010:2011 представлена на рис. 1.

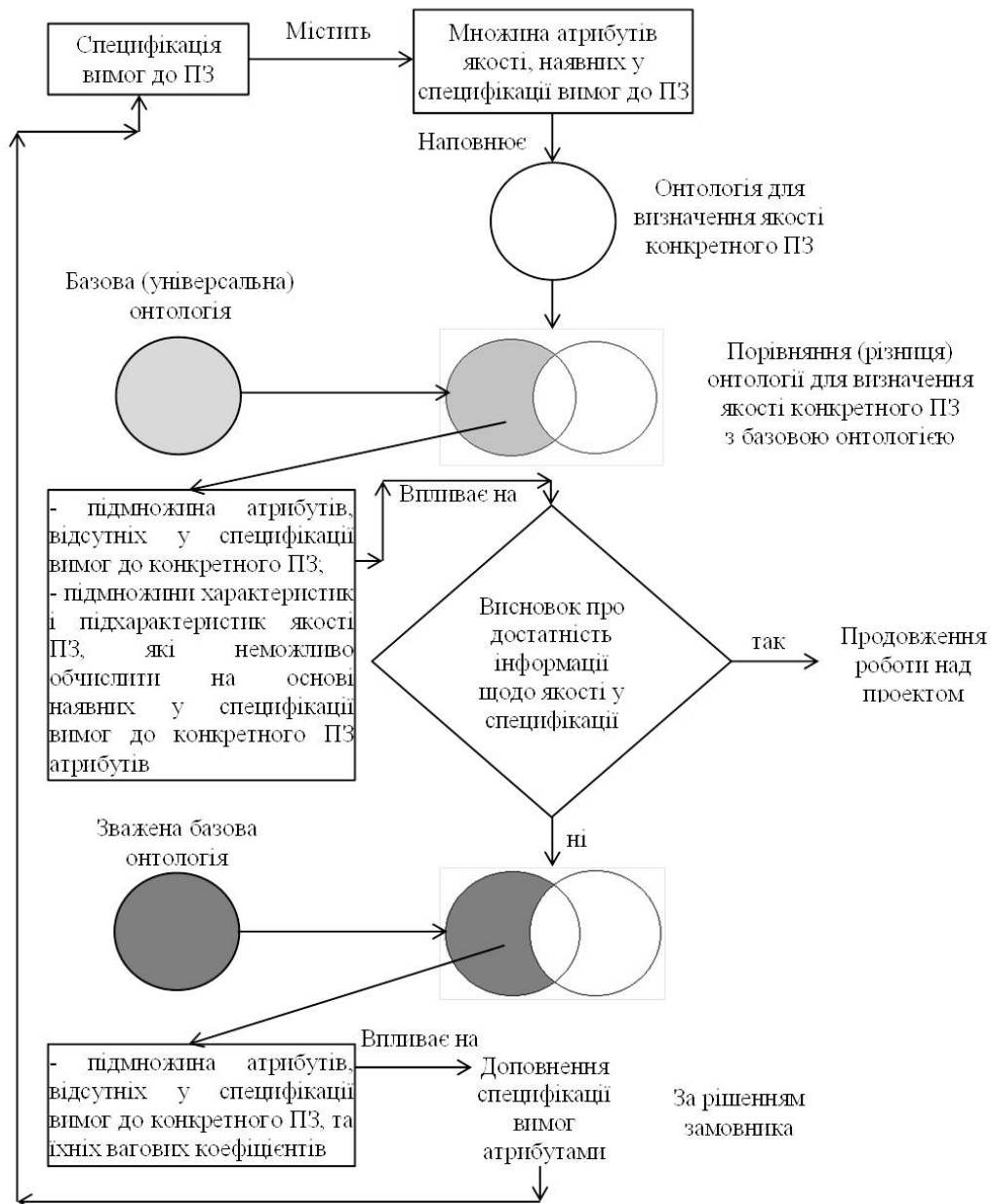


Рис. 1. Інформаційна модель процесу оцінювання достатності інформації для визначення якості ПЗ на основі стандарту ISO 25010:2011

Розроблені моделі процесу оцінювання достатності інформації для визначення якості ПЗ за стандартом ISO 25010:2011 відображають особливості оцінювання достатності інформації для визначення якості ПЗ за стандартом ISO 25010:2011, забезпечують адаптацію до особливостей предметної галузі та надають можливість доступу, аналізу та розуміння інформації не лише людиною, але й віртуальними агентами (ботами), а також є теоретичним підґрунтям для розроблення методів та засобів оцінювання достатності інформації щодо якості у специфікаціях вимог до ПЗ.

**Моделі процесу оцінювання достатності інформації для визначення якості та складності програмного забезпечення з використанням результатів метричного аналізу**

У [18] було доведено, що якість ПЗ на етапі проектування архітектури ( $QDS$ ) залежить від 14 метрик ( $\{sqm_1, \dots, sqm_{14}\}$ ), а складність ПЗ на етапі проектування архітектури ( $CXDS$ ) залежить від 10 метрик з точними або прогнозованими значеннями ( $\{scxm_1, \dots, scxm_{10}\}$ ). Множину метрик якості ПЗ на етапі проектування архітектури можна представити також у вигляді  $SQM = \{SQM_{exv}, SQM_{prv}\}$ , де  $SQM_{exv}$  – підмножина метрик якості з точними значеннями на етапі проектування архітектури,  $SQM_{prv}$  – підмножина метрик якості з прогнозованими значеннями на етапі проектування архітектури. Множину метрик складності ПЗ на етапі проектування архітектури можна представити також у вигляді  $SCXM = \{SCXM_{exv}, SCXM_{prv}\}$ , де  $SCXM_{exv}$  – підмножина метрик складності з точними значеннями на етапі проектування архітектури,  $SCXM_{prv}$  – підмножина метрик складності з прогнозованими значеннями на етапі проектування архітектури. Кожна з зазначених метрик є функцією від декількох показників [18], причому метрики якості та складності залежать всього від 72 показників, але від 42 різних показників, тоді множина показників якості та складності ПЗ для подальшого метричного аналізу має вигляд  $SQCXI = \{sqcxi_1, \dots, sqcxi_{42}\}$ .

Модель якості та складності програмного забезпечення з використанням результатів метричного аналізу на основі онтології має вигляд:  $O_{metr} = \langle X_{metr}, RX_{metr}, F_{metr} \rangle$ , де  $X_{metr}$  – скінченна множина метрик і показників якості та складності ПЗ,  $RX_{metr}$  – скінченна множина відношень між поняттями,  $F_{metr}$  – скінченна множина функцій інтерпретації, заданих для метрик і показників якості та складності ПЗ.

Множина метрик і показників якості та складності ПЗ:  $X_{metr} = \{SQM, SCXM, SQCXI\} = \{x^1_{metr}, \dots, x^{66}_{metr}\}$ , де:  $\{x^1_{metr}, \dots, x^{14}_{metr}\} \in SQM$  (причому  $\{x^1_{metr}, \dots, x^5_{metr}\} \in SQM_{exv}$ ,  $\{x^6_{metr}, \dots, x^{14}_{metr}\} \in SQM_{prv}$ ),  $\{x^{15}_{metr}, \dots, x^{24}_{metr}\} \in SCXM$  (причому  $\{x^{15}_{metr}, \dots, x^{18}_{metr}\} \in SCXM_{exv}$ ,  $\{x^{19}_{metr}, \dots, x^{24}_{metr}\} \in SCXM_{prv}$ ),  $\{x^{25}_{metr}, \dots, x^{66}_{metr}\} \in SQCXI$ . Множина відношень між поняттями  $RX_{metr}$  складається з відношення «залежить від», тобто  $RX_{metr} = \{ "depends on" \}$ . Множина  $F_{metr}$  функцій інтерпретації, заданих для якості ПЗ, складається з функцій залежності якості від метрик якості, метрик від показників якості, тобто  $F_{metr} = \{f^1_{metr}, \dots, f^{26}_{metr}\} = \{\psi(), \psi_1(), \dots, \psi_{14}(), \varphi(), \varphi_1(), \dots, \varphi_{10}()\}$ .

Тоді базова (універсальна) модель для визначення якості та складності програмного забезпечення з використанням результатів метричного аналізу на основі онтології має вигляд [18, 20]:

$$O_{metr} = \left\{ \begin{array}{l} sqm_1, \dots, sqm_{14}, scxm_1, \dots, scxm_{10}, sqcxi_1, \dots, sqcxi_{42}, \\ "depends on", \psi(), \psi_1(), \dots, \psi_{14}(), \varphi(), \varphi_1(), \dots, \varphi_{10}() \end{array} \right\}. \quad (5)$$

Оскільки необхідним є врахування вагових коефіцієнтів показників якості та складності ПЗ, то є потреба внесення таких вагових коефіцієнтів у базову (універсальну) модель для визначення якості та складності ПЗ з використанням результатів метричного аналізу та побудови зваженої онтології. Тоді множина метрик і показників якості та складності ПЗ моделі для визначення якості та складності ПЗ з використанням результатів метричного аналізу на основі зваженої онтології матиме наступний вигляд:

$X_{metrw} = \{x^1_{metrw}, \dots, x^{38}_{metrw}\}$ , де:  $\{x^1_{metrw}, \dots, x^{14}_{metrw}\} \in SQM$ ,  $\{x^{15}_{metrw}, \dots, x^{24}_{metrw}\} \in SCXM$ ,  $\{x^{25}_{metr}, \dots, x^{66}_{metr}\} = \{(sqcxi_i, w_i^{ind}), \dots, (sqcxi_{42}, w_{42}^{ind})\}$ , тобто підмножина  $\{x^1_{metr}, \dots, x^{66}_{metr}\}$  складається з пар  $(sqcxi_i, w_i^{ind})$ , де  $sqcxi_i$  –  $i$ -й показник якості та складності ПЗ ( $sqcxi_i \in SQCXI$ ),  $w_i^{ind}$  – ваговий коефіцієнт  $i$ -го показника,  $i = 1, 42$ .

Базова (універсальна) модель для визначення якості та складності програмного забезпечення з використанням результатів метричного аналізу на основі зваженої онтології матиме наступний вигляд:

$$O_{metrw} = \left\{ \begin{array}{l} sqm_1, \dots, sqm_{14}, scxm_1, \dots, scxm_{10}, (sqcxi_1, w_1^{ind}), \dots, (sqcxi_{42}, w_{42}^{ind}), \\ "depends on", \psi(), \psi_1(), \dots, \psi_{14}(), \varphi(), \varphi_1(), \dots, \varphi_{10}() \end{array} \right\}. \quad (6)$$

Модель для визначення якості та складності конкретного програмного продукту з використанням

результатів метричного аналізу на основі онтології має вигляд [18, 20]:

$$O_{\text{metr}}^{\text{real}} = \left\{ \begin{array}{l} \{sqm_1, \dots, sqm_{nqm}, scxm_1, \dots, scxm_{ncxm}, sqcxi_1, \dots, sqcxi_{nqcx_i}\} \\ \text{"depends on", } \psi(), \psi_1(), \dots, \psi_{14}(), \varphi(), \varphi_1(), \dots, \varphi_{10}() \end{array} \right\}, \quad (7)$$

де  $nqm$  ( $nqm \leq 14$ ) – кількість метрик якості ПЗ, які можна обчислити на основі наявних у специфікації вимог до конкретного ПЗ показників,  $ncxm$  ( $ncxm \leq 10$ ) – кількість метрик складності ПЗ, які можна обчислити на основі наявних у специфікації вимог до конкретного ПЗ показників,  $nqcx_i$  ( $nqcx_i \leq 42$ ) – кількість показників якості та складності, наявних у специфікації вимог до конкретного ПЗ. Онтологія (онтологічна база знань), яка розроблятиметься за моделлю (7), наповнюватиметься на основі інформації, взятої зі специфікації вимог до конкретного ПЗ.

Враховуючи розроблені базові (універсальні) моделі для визначення якості та складності ПЗ з використанням результатів метричного аналізу на основі онтології та на основі зваженої онтології, а також модель для визначення якості та складності конкретного ПЗ з використанням результатів метричного аналізу на основі онтології, представлені формулами (5)–(7), розробимо моделі процесу оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу.

Процес оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу полягає у:

1) порівнянні онтології для визначення якості та складності конкретного ПЗ з використанням результатів метричного аналізу з базовою (універсальною) онтологією з метою виявлення показників, відсутніх в онтології для визначення якості та складності конкретного ПЗ, тобто відсутніх у специфікації вимог до ПЗ, за якою ця онтологія була побудована, а також виявлення метрик якості та складності ПЗ, які неможливо обчислити на основі наявних у специфікації вимог до конкретного ПЗ показників;

2) формуванні висновку про недостатність інформації щодо якості у специфікації вимог до конкретного ПЗ, якщо встановлено показники, відсутні у специфікації вимог до ПЗ, а також метрики якості та складності ПЗ, які неможливо обчислити на основі наявних показників у специфікації вимог до конкретного ПЗ;

3) порівнянні онтології для визначення якості та складності конкретного ПЗ зі зваженою базовою (універсальною) онтологією, якщо було сформовано висновок про недостатність інформації, – з метою встановлення вагових коефіцієнтів показників, відсутніх у специфікації вимог до конкретного ПЗ, для подальшого визначення пріоритетності доповнення таких показників у специфікацію вимог до ПЗ.

Тоді модель процесу оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу на основі онтології може бути представлена у наступному формалізованому вигляді:

$$\left\{ \begin{array}{l} SMIQM = O_{\text{metr}} \setminus O_{\text{metr}}^{\text{real}} \\ \text{if } \{sqm_1, \dots, sqm_{(14-nqm)}, scxm_1, \dots, scxm_{(10-ncxm)}, sqcxi_1, \dots, sqcxi_{(42-nqcx_i)}\} = \emptyset \\ \text{then "SRS information is sufficient"} \\ \text{else: "SRS information is insufficient"} \\ SMIW = O_{\text{metr}}^w \setminus O_{\text{metr}}^{\text{real}} \end{array} \right\}, \quad (8)$$

де  $SMIQM = \{sqm_1, \dots, sqm_{(14-nqm)}, scxm_1, \dots, scxm_{(10-ncxm)}, sqcxi_1, \dots, sqcxi_{(42-nqcx_i)}\}$ ,

$\{sqm_1, \dots, sqm_{(14-nqm)}\}$  – підмножина метрик якості ПЗ, які неможливо обчислити на основі наявних показників у специфікації вимог до конкретного ПЗ;  $\{scxm_1, \dots, scxm_{(10-ncxm)}\}$  – підмножина метрик складності ПЗ, які неможливо обчислити на основі наявних показників у специфікації вимог до конкретного ПЗ;  $\{sqcxi_1, \dots, sqcxi_{(42-nqcx_i)}\}$  – підмножина показників, відсутніх у специфікації вимог до конкретного ПЗ;

$SMIW = \{(sqcxi_1, w_1^{\text{ind}}), \dots, (sqcxi_{(42-nqcx_i)}, w_{(42-nqcx_i)}^{\text{ind}})\}$  – підмножина показників, відсутніх у специфікації вимог до конкретного ПЗ, та їхніх вагових коефіцієнтів.

Інформаційна модель процесу оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу представлена на рис. 2.

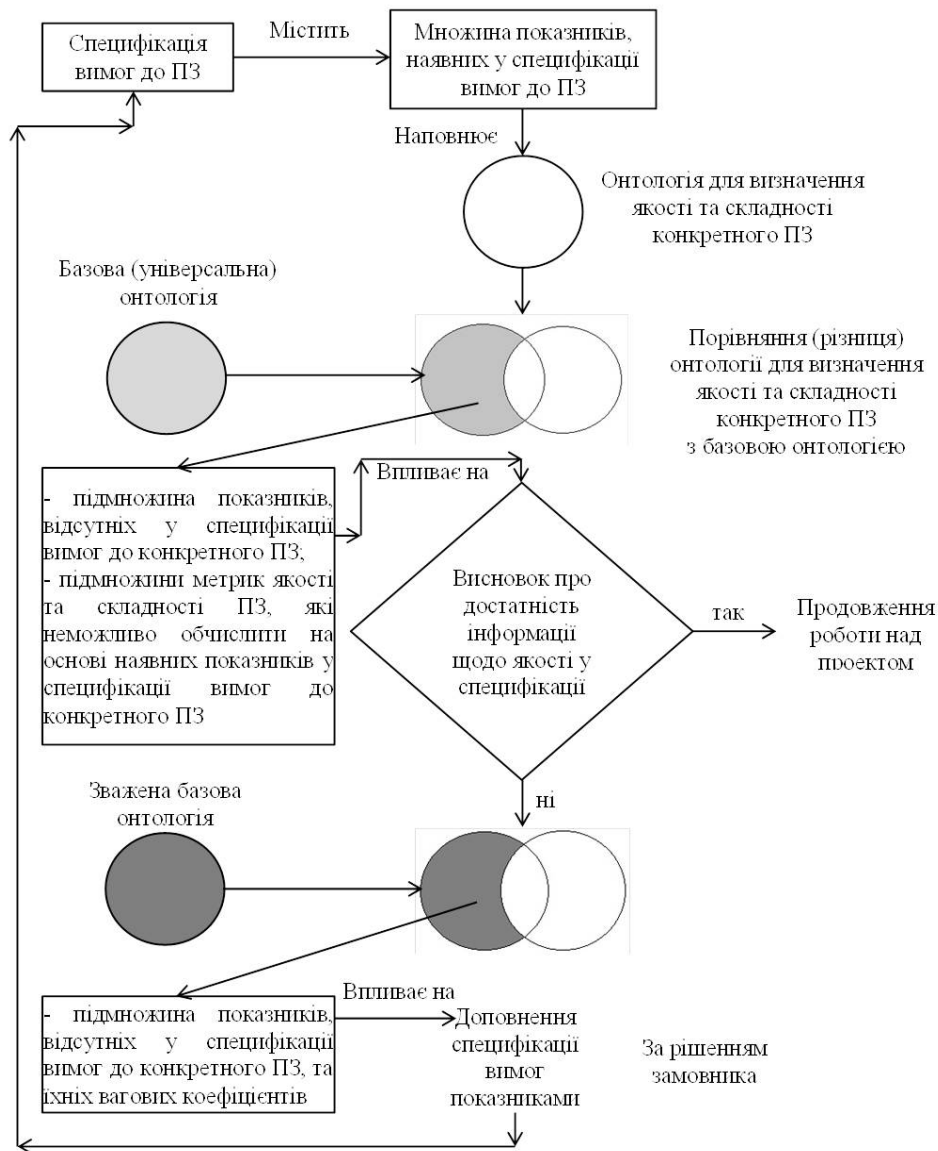


Рис. 2. Інформаційна модель процесу оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу

Розроблені моделі процесу оцінювання достатності інформації для визначення якості та складності ПЗ з використанням результатів метричного аналізу призначені для підтримки процесу оцінювання якості ПЗ на ранніх етапах життєвого циклу, забезпечують висновок про достатність інформації щодо якості у специфікації вимог та множину показників, які необхідно доповнити у специфікацію в разі недостатності інформації, а також є теоретичним підґрунтям для розроблення методів та засобів оцінювання достатності інформації щодо якості у специфікаціях вимог до ПЗ.

### Модель процесу оцінювання результатів проектування та прогнозування характеристик програмного забезпечення на основі штучної нейронної мережі

Для оцінювання і прогнозування складності та якості ПЗ на основі метричного аналізу потрібно розв'язати завдання визначення взаємозв'язку між значеннями метрик та такими характеристиками, як якість і складність проекту ПЗ, а також прогнозована складність та якість розробленого ПЗ. Отже, для розв'язання задачі оцінювання результатів проектування та прогнозування характеристик якості програмного забезпечення на основі метричного аналізу потрібно розв'язати системи рівнянь, представлені у [18, 20], для яких невідомими є функції залежності якості та складності ПЗ від метрик, які за кількісними оцінками метрик якості та складності ПЗ надаватимуть коректну кількісну оцінку якості та складності ПЗ.

Універсальними структурами, що забезпечують можливість розв'язати задачу представлення функції довільного вигляду, узагальнити інформацію та виявити залежності між вхідними і результуючими даними, є штучні нейронні мережі (ШНМ), отже, для реалізації вище зазначених систем рівнянь буде використовуватись саме ШНМ. Тоді слід розробити ШНМ, яка опрацюватиме множини показників специфікації, здійснюватиме апроксимацію показників та надаватиме прогнозовані оцінки характеристик програмного проекту.

Для формування вхідних даних ШНМ знадобляться множина метрик етапу проектування

архітектури з точними значеннями (множина  $TMP = \{tmp_e \mid e = 1..9\} = (SQM_{exv} \cup SCXM_{exv})$ ) та множина метрик етапу проектування архітектури з прогнозованими значеннями (множина  $PMP = \{pmp_l \mid l = 1..15\} = (SQM_{prv} \cup SCXM_{prv})$ ), які були виділені та визначені у [18].

З одержаних множин  $TMP, PMP$  формуються вектори, які подаються на вхід штучної нейронної мережі (ШНМ). Результатом роботи ШНМ є оцінка складності та якості проекту на основі точних метрик етапу проектування архітектури та прогнозовані оцінки складності та якості майбутнього програмного забезпечення на основі прогнозованих метрик етапу проектування архітектури [18].

Вибір структури та архітектури нейронної мережі здійснюється відповідно до особливостей і складності задачі. Для розв'язку окремих типів задач вже існують оптимальні конфігурації. Для вибору архітектури для аналізу метрик етапу проектування архітектури ПЗ та прогнозу характеристик якості ПЗ було проведено аналіз відомих архітектур штучних нейронних мереж [18], в результаті якого було обрано багатошаровий перцептрон як архітектуру для аналізу метрик етапу проектування архітектури ПЗ та прогнозування характеристик якості ПЗ [18]. Структура цієї ШНМ представлена на рис. 3.

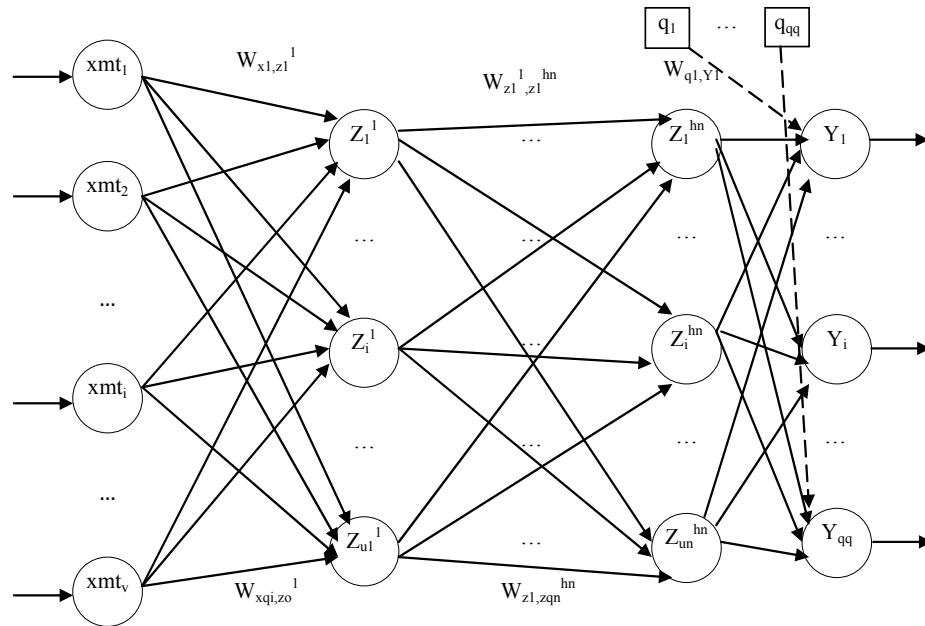


Рис. 3. Модель процесу оцінювання результатів проектування та прогнозування характеристик програмного забезпечення на основі штучної нейронної мережі

Нейрони ( $i$  відповідно входи) рецепторного шару визначено множиною  $\overline{XMT} = \{xmt_i\}, i = (\overline{1,v})$ , де  $xmt_i$  –  $i$ -й нейрон (вхід) рецепторного шару,  $v$  – кількість входів (кількість метрик якості та складності ПЗ з точними та прогнозованими значеннями).

Нейрони першого прихованого шару прямонапрявленого перцептрона визначено множиною  $\overline{Z^1} = \{z_i^1\}, i = (\overline{1,u1})$ , де  $z_i^1$  –  $i$ -й нейрон шару  $z^1$ ,  $u1$  – кількість нейронів першого прихованого шару. Рецепторні входи цього шару визначено множиною:  $\overline{X^{z^1}} = \{x^{z^1}\}, i = (\overline{1,u1})$ , де  $x^{z^1}$  –  $i$ -й вхід, пов'язаний з  $z_i^1$ -м нейроном.

Нейрони другого прихованого шару прямонапрявленого перцептрона визначено множиною  $\overline{Z^2} = \{z_i^2\}, i = (\overline{1,u2})$ , де  $z_i^2$  –  $i$ -й нейрон шару  $z^2$ ,  $u2$  – кількість нейронів другого прихованого шару. Рецепторні входи цього шару визначено множиною:  $\overline{X^{z^2}} = \{x^{z^2}\}, i = (\overline{1,u2})$ , де  $x^{z^2}$  –  $i$ -й вхід, пов'язаний з  $z_i^2$ -м нейроном.

Нейрони  $hn$ -го прихованого шару прямонапрявленого перцептрона визначено множиною  $\overline{Z^{hn}} = \{z_i^{hn}\}, i = (\overline{1,un})$ , де  $z_i^{hn}$  –  $i$ -й нейрон шару  $z^{hn}$ ,  $un$  – кількість нейронів  $hn$ -го прихованого шару. Рецепторні входи цього шару визначено множиною  $\overline{Z^{z^{hn}}} = \{z^{z^{hn-1}}\}, i = (\overline{1,unp})$ , де  $z^{z^{hn-1}}$  –  $i$ -й нейрон попереднього ( $hn - 1$ )-го шару, що пов'язаний з  $z_i^{hn}$ -м нейроном,  $unp$  – кількість нейронів ( $hn - 1$ )-го прихованого шару.

Вектор порогових величин зміщень множини нейронних елементів визначено як:



$\bar{q} = \{q_i\}$ ,  $i = (\overline{1, qq})$ , де  $q_i$  – зміщення  $i$ -го нейронного елемента,  $qq$  – кількість зміщень нейронних елементів.

Ваги зв'язків відображено ваговими матрицями:  $\overline{W}_{xmt, z^1} = \langle w_{x_i, z_j^1} \rangle$ ,  $i = (\overline{1, v}), j = (\overline{1, u1})$ , де  $w_{x_i, z_j^1}$  – ваговий коефіцієнт зв'язку між  $xmt_i$ -м входом та  $z_j^1$ -м нейроном першого прихованого шару;  $\overline{W}_{z^1, z^2} = \langle w_{z_i^1, z_j^2} \rangle$ ,  $i = (\overline{1, u1}), j = (\overline{1, u2})$ , де  $w_{z_i^1, z_j^2}$  – ваговий коефіцієнт зв'язку між  $z_i^1$ -м нейроном 1-го прихованого шару та  $z_j^2$ -м нейроном 2-го прихованого шару;  $\overline{W}_{z^1, z^{hn}} = \langle w_{z_i^1, z_j^{hn}} \rangle$ ,  $i = (\overline{1, u1}), j = (\overline{1, un})$ , де  $w_{z_i^1, z_j^{hn}}$  – ваговий коефіцієнт зв'язку між  $z_i^1$ -м нейроном 1-го прихованого шару та  $z_j^{hn}$ -м нейроном  $hn$ -го прихованого шару;  $\overline{W}_{z^{hn-1}, z^{hn}} = \langle w_{z_i^{hn-1}, z_j^{hn}} \rangle$ ,  $i = (\overline{1, unp}), j = (\overline{1, un})$ , де  $w_{z_i^{hn-1}, z_j^{hn}}$  – ваговий коефіцієнт зв'язку між  $z_i^{hn-1}$ -м нейроном  $(hn-1)$ -го прихованого шару та  $z_j^{hn}$ -м нейроном  $hn$ -го прихованого шару;  $\overline{W}_{q, Y} = \langle w_{q_i, Y_i} \rangle$ ,  $i = (\overline{1, qq})$ , де  $w_{q_i, Y_i}$  – ваговий коефіцієнт зв'язку між  $q_i$ -м зміщенням та  $Y_i$ -м нейроном вихідного шару.

Множина вихідних функціоналів ефекторного шару ШНМ становить  $\bar{Y} = \{Y_{ki}\}$ ,  $ki = (\overline{1, qq})$ , де  $Y_{ki}$  –  $ki$ -й функціонал ефекторного шару багатшарового персеプトрону,  $qq$  – кількість виходів ШНМ (кількість прогнозованих характеристик ПЗ).

Формула для визначення  $ki$ -го функціоналу ефекторного шару ШНМ  $Y_{ki}$  має вигляд:

$$Y_{ki} = fs \left( fan(z^1) \cdot \left( \sum_{i=1}^o \sum_{j=1}^{qn2} (z_i^1 \cdot w_{z_i^1, z_j^2}) + \dots + \sum_{i=1}^o \sum_{j=1}^{qn} (z_i^1 \cdot w_{z_i^1, z_j^{hn}}) \right) + \dots + fan(z^{hn-1}) \cdot \left( \sum_{i=1}^{qnm} \sum_{j=1}^{qn} (z_i^{hn-1} \cdot w_{z_i^{hn-1}, z_j^{hn}}) \right) + \sum_{i=1}^{qki} (xmt_i \cdot w_{x_i, z_i}) - q_{ki} \cdot w_{q_i, Y_{ki}} \right) \quad (9)$$

де  $fan(z^1)$  – активаційна функція нейронів 1-го прихованого шару,  $fan(z^{hn-1})$  – активаційна функція нейронів  $(hn-1)$ -го прихованого шару,  $fs$  – активаційна функція нейронів ефекторного (вихідного) шару ШНМ,  $qki$  – кількість метрик, які впливають на  $ki$ -й вихід ШНМ (на  $ki$ -у прогнозовану характеристику ПЗ),  $xmt_i$  –  $i$ -й вхід ШНМ, який впливає на  $ki$ -й вихід ШНМ.

Активаційною функцією нейронів прихованих (асоціативних) шарів є функція гіперболічного тангенсу. Активаційною функцією нейронів ефекторного шару є лінійна функція. Результати лінійної активаційної функції нейронів ефекторних шарів лежать в інтервалі  $[0;1]$ .

Вперше запропонована модель процесу оцінювання результатів проектування та прогнозування характеристик ПЗ на основі ШНМ відрізняється від існуючих тим, що дає можливість враховувати важливість кожної метрики якості та складності ПЗ, а також взаємний вплив метрик в межах кожної характеристики програмного забезпечення. Вихідні функціонали ШНМ, що відповідають значенням характеристик ПЗ (якість програмного проекту, складність програмного проекту, прогнозована якість ПЗ, прогнозована складність ПЗ), дають можливість оцінити сумарний вплив метрик якості та складності на характеристики ПЗ.

### Висновки

Розроблено моделі якості ПЗ за стандартом ISO 25010 та моделі якості і складності ПЗ з використанням результатів метричного аналізу на основі онтологій та зважених онтологій, які, за рахунок ґрунтування на онтологіях для відображення знань щодо взаємозв'язку та кореляції характеристик і підхарактеристик за атрибутами, метрик за показниками, дають змогу виконати моделювання процесу оцінювання достатності інформації щодо якості у специфікаціях вимог до ПЗ.

Вперше розроблено інформаційні моделі процесу оцінювання достатності інформації для визначення якості ПЗ на основі стандарту ISO 25010:2011 та для визначення якості і складності ПЗ з використанням результатів метричного аналізу, які відображають особливості процесу оцінювання достатності інформації щодо якості у специфікації вимог до ПЗ, а також причинно-наслідковий зв'язок між моделями якості ПЗ за стандартом ISO 25010 на основі онтологій, між моделями якості і складності ПЗ з використанням результатів метричного аналізу на основі онтологій. Вперше розроблено моделі процесу оцінювання достатності інформації для визначення якості ПЗ на основі стандарту ISO 25010:2011 та для визначення якості і складності ПЗ з використанням результатів метричного аналізу, які формалізують процес оцінювання достатності інформації щодо якості у специфікації вимог до ПЗ та забезпечують теоретичне підґрунтя для розроблення методів та засобів оцінювання достатності інформації щодо якості у

специфікація вимог до ПЗ.

Основним внесок усіх розроблених моделей є те, що моделі дають можливість оцінити наявність інформації щодо якості для конкретних проектів (є вона чи взагалі відсутня), виявити, якої інформації недостатньо, і виявити джерела, де шукати недостатню інформацію.

Розроблена модель процесу оцінювання результатів проектування та прогнозування характеристик ПЗ на основі ШНМ відрізняється від відомих тим, що дає можливість враховувати ваги та взаємовпливи метрик якості та складності ПЗ в межах кожної характеристики ПЗ, а також оцінити сумарний вплив метрик на характеристики ПЗ (якість і складність програмного проекту, прогнозовані якість і складність ПЗ).

### Література

1. Hastie Shane. Standish Group 2015 Chaos Report – Q&A with Jennifer Lynch [Electronic resource] / Shane Hastie, Stéphane Wojewoda. – Electronic data. – Mode of access : <http://www.infoq.com/articles/standish-chaos-2015> (viewed on November 14, 2017). – Title from the screen.
2. Latest study shows rise in project failures [Electronic resource] : [Web-site]. – Electronic data. – Mode of access : <http://kinzz.com/resources/articles/91-project-failures-rise-study-shows>. – (viewed on November 14, 2017). – Title from the screen.
3. Maedche A. Software for people: fundamentals, trends and best practices (Management for professionals) / A. Maedche, A. Botzenhardt, L. Neer. – Springer-Verlag Berlin Heidelberg, 2012. – 293 p.
4. Cost of a bug within a software lifecycle [Electronic resource] : [Web-site]. – Electronic data. – Mode of access : <http://www.testically.org/2012/02/09/cost-of-a-bug-within-a-software-lifecycle/>. – (viewed on November 14, 2017). – Title from the screen.
5. Макконнелл С. Совершенный код. Мастер-класс / С. Макконнелл. – М. : Издательство "Русская редакция", 2013. – 896 с.
6. Шамие К. Системная инженерия для «чайников» / К. Шамие. – New York : John Wiley & Sons, 2014. – 76 с.
7. Levenson N. G. Engineering a safer world: systems thinking applied to safety / N. G. Levenson. – MIT Press, 2012. – 560 p.
8. Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). System and software quality models: ISO/IEC 25010:2011. – [Introduced 01.03.2011]. – Geneva (Switzerland): ISO, 2011. – 34 p. – (International standard).
9. Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). Measurement of system and software product quality: ISO 25023:2016. – [Introduced 31.03.2016]. – Geneva (Switzerland): ISO, 2016. – 45 p. – (International standard).
10. Fenton N. Software metrics: a rigorous and practical approach / N. Fenton, J. Bieman. – CRC Press, 2014. – 617 p.
11. Shostak I. Ontology approach to realization of information technology for normative profile forming at critical software certification / I. Shostak, I. Butenko // Збірник наукових праць Військового інституту КНУ ім. Т. Г. Шевченка. – 2012. – № 38. – С. 250–253.
12. Литвин В. Моделирование интеллектуальных систем поддержки принятия решений с использованием онтологического подхода / В. Литвин // Радиоелектроніка, інформатика, управління. – 2011. – № 2. – С. 93–101.
13. Henderson-Sellers B. An ontology for ISO software engineering standards: 1) Creating the infrastructure / B. Henderson-Sellers, C. Gonzalez-Perez, T. McBride, G. Low // Computer Standards & Interfaces. – 2014. – Vol. 36. – Issue 3. – P. 563–576.
14. Ruy F. B. An ISO-based software process ontology pattern language and its application for harmonizing standards / F. B. Ruy, R. A. Falbo, M. P. Barcellos, G. Guizzardi, G. K. S. Quirino // ACM SIGAPP Applied Computing Review. – 2015. – Vol. 15. – Issue 2. – P. 27–40.
15. Hamri M. M. Building an ontology for the metamodel ISO/IEC24744 using MDA process / M. M. Hamri, S. M. Benslimane // International Journal on Modern Education and Computer Science. – 2015. – Vol. 8. – P. 48–60.
16. Jin D. Ontology-based software analysis and reengineering tool integration: The OASIS service-sharing methodology / D. Jin, J. R. Cordy // The 21-st IEEE International Conference on Software Maintenance, September 25–30, 2005 : Proceedings. – Budapest (Hungary), 2005. – P. 613–616.
17. Bajnaid N. O. An ontological approach to model software quality assurance knowledge domain / N. O. Bajnaid, R. Benlamri, A. Pakstas, Sh. Salekzamankhani // Lecture Notes on Software Engineering. – 2016. – Vol. 4. – No. 3. – P. 193–198.
18. Говорущенко Т. О. Методологія оцінювання достатності інформації для визначення якості програмного забезпечення : монографія / Т. О. Говорущенко. – Хмельницький : Хмельницький національний університет, 2017. – 310 с.
19. Hovorushchenko T. Methodology of evaluating the sufficiency of information for software quality assessment according to ISO 25010 / T. Hovorushchenko // Accepted for publication in Journal of Information and Organizational Sciences. – 2017.
20. Hovorushchenko T. Models and methods of evaluation of information sufficiency for determining the

software complexity and quality based on the metric analysis results / T. Hovorushchenko // Central European Researchers Journal. – 2016. – Vol. 2. – Issue 2. – P. 42–53.

#### References

1. Hastie Shane. Standish Group 2015 Chaos Report – Q&A with Jennifer Lynch [Electronic resource] / Shane Hastie, Stéphane Wojewoda. – Electronic data. – Mode of access: <http://www.infoq.com/articles/standish-chaos-2015> (viewed on November 14, 2017). – Title from the screen.
2. Latest study shows rise in project failures [Electronic resource]: [Web-site]. – Electronic data. – Mode of access: <http://kinzz.com/resources/articles/91-project-failures-rise-study-shows> viewed on November 14, 2017). – Title from the screen.
3. Maedche A. Software for people: fundamentals, trends and best practices (Management for professionals) / A. Maedche, A. Botzenhardt, L. Neer. – Springer-Verlag Berlin Heidelberg, 2012. – 293 p.
4. Cost of a bug within a software lifecycle [Electronic resource]: [Web-site]. – Electronic data. – Mode of access: <http://www.testically.org/2012/02/09/cost-of-a-bug-within-a-software-lifecycle/> (viewed on November 14, 2017). – Title from the screen.
5. McConnell S. Code complete / S. McConnell. – Microsoft Press, 2013. – 896 p.
6. Shamieh C. Systems Engineering for Dummies / C. Shamieh. – Wiley Publishing, 2011. – 74 p.
7. Levenson N. G. Engineering a safer world: systems thinking applied to safety / N. G. Levenson. – MIT Press, 2012. – 560 pp.
8. Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). System and software quality models: ISO/IEC 25010:2011. – [Introduced 01.03.2011]. – Geneva (Switzerland): ISO, 2011. – 34 p. – (International standard).
9. Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). Measurement of system and software product quality: ISO 25023:2016. – [Introduced 31.03.2016]. – Geneva (Switzerland): ISO, 2016. – 45 p. – (International standard).
10. Fenton N. Software metrics: a rigorous and practical approach / N. Fenton, J. Bieman. – CRC Press, 2014. – 617 p.
11. Shostak I. Ontology approach to realization of information technology for normative profile forming at critical software certification / I. Shostak, I. Butenko // Herald of the Military Institute of Kiev National University named after Taras Shevchenko. – 2012. – No. 38. – Pp. 250–253.
12. Lytvyn V. Modeling of intelligent decision support systems using the ontological approach / V. Lytvyn // Radio electronics, computer science, management. – 2011. No. 2. – Pp. 93-101 (2011) [in Ukrainian]
13. Henderson-Sellers B. An ontology for ISO software engineering standards: 1) Creating the infrastructure / B. Henderson-Sellers, C. Gonzalez-Perez, T. McBride, G. Low // Computer Standards & Interfaces. – 2014. – Vol. 36. – Issue 3. – Pp. 563-576.
14. Ruy F. B. An ISO-based software process ontology pattern language and its application for harmonizing standards / F. B. Ruy, R. A. Falbo, M. P. Barcellos, G. Guizzardi, G. K. S. Quirino // ACM SIGAPP Applied Computing Review. – 2015. – Vol. 15. – Issue 2. – Pp. 27-40.
15. Hamri M. M. Building an ontology for the metamodel ISO/IEC24744 using MDA process / M. M. Hamri, S. M. Benslimane // International Journal on Modern Education and Computer Science. – 2015. – Vol. 8. – Pp. 48-60.
16. Jin D. Ontology-based software analysis and reengineering tool integration: The OASIS service-sharing methodology / D. Jin, J. R. Cordy // The 21-st IEEE International Conference on Software Maintenance, September 25-30, 2005 : Proceedings. – Budapest (Hungary), 2005. – Pp. 613-616.
17. Bajnoid N. O. An ontological approach to model software quality assurance knowledge domain / N. O. Bajnoid, R. Benlamri, A. Pakstas, Sh. Salekzamankhani // Lecture Notes on Software Engineering. – 2016. – Vol. 4. – No. 3. – Pp. 193-198.
18. Hovorushchenko T. Methodology of evaluating the sufficiency of information for software quality assessment: monograph / T. Hovorushchenko. – Khmelnytsky National University, 2017. – 310 p.
19. Hovorushchenko T. Methodology of evaluating the sufficiency of information for software quality assessment according to ISO 25010 / T. Hovorushchenko // Accepted for publication in Journal of Information and Organizational Sciences. – 2017.
20. Hovorushchenko T. Models and methods of evaluation of information sufficiency for determining the software complexity and quality based on the metric analysis results / T. Hovorushchenko // Central European Researchers Journal. – 2016. – Vol. 2. – Issue 2. – Pp. 42-53.

Рецензія/Peer review : 20.11.2017 р.

Надрукована/Printed :06.12.2017 р.

Рецензент: д.т.н., професор кафедри КНІТ ХНУ, О.В.Бармак