DOI: 10.20998/2079-0023.2024.01.06
UDC 004.94

A. M. KOPP, Doctor of Philosophy (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Head of Software Engineering and Management Intelligent Technologies Department, Kharkiv, Ukraine, e mail: andrii.kopp@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-3189-5623
D. L. ORLOVSKYI, Candidate of Technical Sciences (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine, e mail: dmytro.orlovskyi@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-8261-2988
U. S. LITVINOVA, Candidate of Technical Sciences (PhD), National Technical University "Kharkiv Polytechnic Institute", Associate Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine, e mail: uliya.litvinova@khpi.edu.ua, ORCID: https://orcid.org/0000-0001-6680-662X

DEVELOPMENT AND RESEARCH OF SOFTWARE SOLUTION FOR BUSINESS PROCESS MODEL CORRECTNESS ANALYSIS USING MACHINE LEARNING

Poorly designed business process models are a source of errors and the subsequent costs associated with these errors, such as monetary costs, lost time, or even some harmful impact on people or the environment if the erroneous business process models are associated with critical industries. The BPM (Business Process Management) lifecycle usually consists of designing, implementing, monitoring, and controlling the business process execution, but it lacks continuous quality control of the created BPMN (Business Process Model and Notation) models. Thus, this paper considers the problem of business process models classification based on their correctness, which solution will ensure quality control of the designed business process models. Thus, this study aims to improve the quality of business process models by developing a software solution for business process models classification based on their correctness. The subject of the study is the process of business process models classification based on their correctness, which uses quality measures and thresholds, usually, complexity measures. The subject of the study is a software solution for business process models classification based on their correctness. Therefore, in this study, the algorithm to solve the problem of BPMN models classification using logistic regression, interface complexity, and modularity measures is proposed, the software requirements are determined, the software development tools are selected, the software for business process models classification based on their correctness is designed, the corresponding software components are developed, the use of a software solution for solving the problem of business process models classification based on their correctness is demonstrated, the obtained results are analyzed and discussed. The developed software indicates high performance of BPMN models classification based on their correctness, achieving high accuracy (99.14 %), precision (99.88 %), recall (99.23 %), and F-score (99.56 %), highlighting the high performance of modeling errors detection.

Keywords: business process modeling, model quality, complexity measures, machine learning, logistic regression, software solution.

Introduction. A business process is a sequence of actions performed within an organization to achieve a specific goal or result. There are several definitions of a business process:

- a business process is a sequence of interrelated actions performed to achieve a specific goal within an organization [1];
- a business process is a sequence of interrelated actions and decisions performed to achieve a specific goal within an organization [2];
- a business process is a sequence of actions performed within an organization to create, produce, or deliver a product or service [3].

Business Process Management (BPM) is an approach that combines management and Information Technology (IT) methodologies to achieve operational excellence in an organization. The primary approach of BPM is business process modeling, which facilitates communication and interaction between business users and IT providers responsible for designing, implementing, and maintaining information systems in the organization [4].

Business process modeling is the graphical representation of the sequence of actions and interdependencies that occur within an organization. It is a critical component of BPM that helps ensure interaction between business and IT users responsible for developing and maintaining information systems in the enterprise. Business process models are used to gather requirements for supporting software systems and to identify and optimize bottlenecks in business operations. Information about business process models can be used for training new employees and for internal quality control in the organization [5].

Business process models are used in BPM to document business processes and define requirements for supporting information systems. As a result, poor quality business process models can make them difficult to understand and maintain, potentially leading to errors that increase time and cost [5].

In addition, poorly developed business process models can reflect inefficiencies in real-world processes. As a result, the quality of created business process models must be carefully controlled to detect and prevent errors at the design stage before they become real errors in the organization's business processes and supporting IT systems, which can lead to unanticipated time and cost expenditures, or even dangerous impacts on people and the environment in critical sectors [5].

Thus, the goal of this work is to improve the quality of business process models by developing a software solution for business process models classification based on their correctness.

The subject of the study is the process of business process models classification based on their correctness.

The subject of the study is a software solution for business process models classification based on their correctness.

Related work. Today, the most widely used notation for modeling business processes is the Business Process

Model and Notation (BPMN), which is also an Object Management Group (OMG) standard [6].

The Business Process Model and Notation (BPMN) is a standard for modeling business processes that provides a graphical notation for specifying business processes on a Business Process Diagram (BPD) based on traditional flowcharting methods. BPMN aims to support business process modeling for both technical and business users by providing a notation that is intuitive to business users and capable of representing the complex semantics of processes. The BPMN 2.0 specification also provides execution semantics and alignment between the graphical notation and other execution languages, including the Business Process Execution Language (BPEL) [7].

The BPMN notation is designed to be understood by all business stakeholders, including business analysts who create and refine processes, technical developers who implement them, and business managers who monitor and control them. In this way, BPMN serves as a common language that bridges the communication gap that often exists between business and IT domains [7].

Flow objects are the most commonly used elements in BPMN business process models, consisting of events, tasks, and gateways [8]:

- Events represent occurrences that happen during a process and change its flow. Events could be receiving a message, encountering an error, or reaching a certain point in time. In culinary terms, events can be state changes (bubbling butter or translucent onions) or time-based triggers (flipping wings after 15 minutes) [8];
- any work performed within a business process, Task, or Sub-process can be considered an Activity, with the most common being tasks [8];
- gateways are used to control the flow of processes, but do not represent work performed. For example, an AND gateway indicates that multiple paths can be selected simultaneously, an XOR gateway indicates that only one of the following paths can be selected, while an OR gateway indicates that one or more paths can be selected [8].

Data Objects represent information flowing through the process, such as reports, emails, or other business documents. Some process activities may require input data before they can be executed [8].

Connecting objects are used to connect flow objects. Solid lines represent Sequence Flows, dashed lines represent Message Flows, and dotted lines represent Associations [8]. Data Associations are used to move data between data objects and between task and process inputs and outputs [8].

Pools and Lanes define the boundaries of the process and its participants. Pools typically represent different organizations, while Lanes divide structures within a single organization. However, there are no hard and fast rules for assigning Pools and Lanes, as this depends on the specific purpose of the business process model [8].

An example of the main elements of the BPMN standard is demonstrated in fig. 1 [9].

In their study [10], the authors emphasize the detrimental effects that high complexity in business process models can have on their quality. Therefore, they advocate

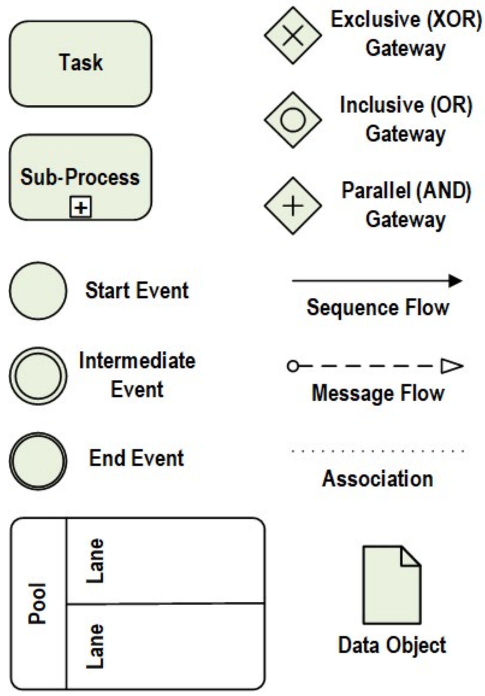

Fig. 1. Basic elements of the BPMN standard [9]

the use of specific complexity measures originated from software engineering [10].

1. Interface Complexity (IC):

$$IC = length \cdot \left[\sum_{n \in N} fan-in(n) + \sum_{n \in N} fan-out(n) \right]^2, \quad (1)$$

where:

- $length$ is the coefficient equal to 1 for tasks and 3 for sub-processes;
- n is the business process element, $n \in N$;
- $fan-in(n)$ is the number of input control flows of the element, $n \in N$;
- $fan-out(n)$ is the number of output control flows of the element, $n \in N$.

2. Modularity (M):

$$M = \sum_{n \in N} \left[fan-in(n) \cdot fan-out(n) \right]^2. \quad (2)$$

In their work [11], the authors identify complexity as a critical quality attribute of business process models that reflects their comprehensibility and maintainability. They define model complexity as the degree to which a model has a structure that is complicated and difficult to understand and analyze [11].

Consequently, the considered problem includes the classification of business process models with respect to their correctness based on the considered complexity measures IC and M, identified as fundamental quality characteristics of business process models.

Methods. Let us review existing methods that can be used to classify business process models for correctness.

Classification algorithms are a class of supervised machine learning methods used to predict or classify data into different classes or categories based on a set of input features. The goal of a classification algorithm is to train a model that can accurately predict the class labels of new, unseen instances based on patterns found in the training data [12].

This type of algorithm is widely used in various fields, including healthcare, finance, and marketing, as well as in applications such as customer churn modeling, email spam filtering, image classification, and many more [12].

Here are some of the most common classification algorithms [13]:

- logistic regression is a statistical technique used to analyze the relationship between a dependent variable and one or more independent variables, it is used to predict the probability of a particular outcome based on input data;
- Support Vector Machine (SVM) is a powerful classification algorithm that can be used for both linear and nonlinear classification, it works by finding the best boundary between different classes in the input data;
- Naive Bayes is the algorithm based on Bayes theorem, it assumes that the input features are independent of each other, it is often used for text classification;
- K-Nearest Neighbors (KNN) is a nonparametric algorithm that classifies new data points based on their proximity to the nearest neighbors in the training data;
- decision trees propose a simple but effective method of classification, they are easy to interpret and understand, and can be used to create rules that can be applied to new data;
- random forest is a popular ensemble learning algorithm that combines multiple decision trees to create a more accurate model, it works by generating a large number of decision trees, each of which makes a prediction about the input data.

Classification algorithms are a critical component of modern machine learning systems. The choice of algorithm depends on the specific task at hand, the nature of the data, and the computational resources available. By selecting and tuning the right classification algorithm, we can create powerful predictive models that automate many tasks and provide valuable recommendations [12].

Since business process models classification based on their correctness involves predicting the probability of errors in them, the logistic regression algorithm can be used for classification tasks in this work.

Let us define classification tasks where the independent variables are continuous in nature and the dependent variable takes a categorical form, such as classes like positive and negative. Real-world examples of classification include emails as spam classification, tumors as malignant or benign, and transactions as fraudulent or genuine. The answers to all of these questions are categorical, either "yes" or "no", making them binary classification tasks [14].

Logistic regression uses a sigmoid or logistic function. An explanation of logistic regression can begin with an explanation of the standard logistic function. A logistic function is a sigmoid function that takes any real value between zero and one.

The logistic (sigmoid) function is defined as [14]:

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}. \quad (3)$$

Plotting a graph of a sigmoidal function will result in an S-shaped curve, an example of which is demonstrated in fig. 2 [15].

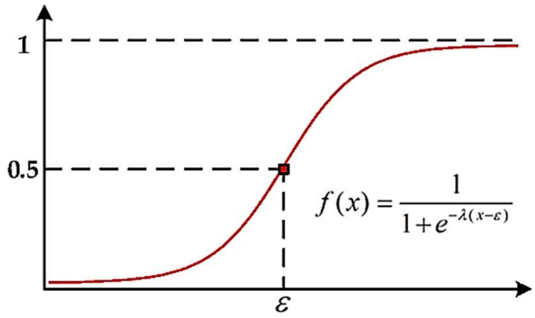

Fig. 2. Graph of the sigmoid function [15]

On the y-axis, the s-shape changes from 0 to 1, which is the probability of an output (fig. 2). For example, assuming that the probability of errors in a business process model is predicted, 1 means that the model is 100 % erroneous based on logistic regression, while 0 means that there is no reason to believe that the business process model is error-prone [15].

For example, let us consider a linear function in a one-dimensional regression model [14]:

$$t = \beta_0 + \beta_1 \cdot x, \quad (4)$$

where:

- x is the independent variable;
- β_0 and β_1 are the logistic regression coefficients.

Thus, the logistic equation that allows predicting the probability of a certain outcome based on the input data will take the following form [14]:

$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \cdot x)}}. \quad (5)$$

Now, when the logistic regression model encounters an outlier, the logistic equation will not allow to obtain values less than 0 or greater than 1 [14].

Features in a dataset often have different scales, measurement units, or ranges. If features are not scaled, those with larger scales can dominate the learning process and introduce errors. By scaling features, we ensure that they contribute equally to the analysis and prevent any particular feature from exerting undue influence [16].

Feature scaling, also known as normalization or data standardization, is a method used in machine learning and data preprocessing to bring different features of a dataset to the same scale. The goal of scaling is to ensure an equal contribution of all features to the learning process and to prevent the dominance of any particular feature or bias in the results [16].

Normalization, also known as Min-Max scaling, is a common technique used in machine learning to normalize feature values within a certain range [16].

The goal of normalization is to transform features so that they all have values in the same scale, usually between 0 and 1 [16]:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}. \tag{6}$$

Standardization is a common technique used in machine learning to normalize data. It transforms the data so that it has zero mean and unit variance. This process causes the data to follow a standard normal distribution, where the mean is 0 and the standard deviation is 1 [16]:

$$x' = \frac{x - \bar{x}}{\sigma}, \tag{7}$$

where:

- $\bar{x}$ is the mean value;
- σ is the standard deviation.

There is no clear rule for choosing an input scaling function for a machine learning model. However, in classification tasks based on logistic regression or other algorithms that use a gradient-based optimization to adjust the model, standardization gives better accuracy than normalization [16].

Therefore, the logistic regression model for business process models classification based on their correctness is using the considered complexity measures IC and M will be as follows:

$$p(IC', M') = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \cdot IC' + \beta_2 \cdot M')}}, \tag{8}$$

where:

- IC' is the standardized value of IC;
- M' is the standardized value of M.

Therefore, business process models for which $p(IC', M') > 0.5$ will be considered error-prone and will require measures to improve their quality.

Interface complexity (IC) and modularity (M) are quantitative measures that reflect important characteristics of BPMN process models.

These measures can be calculated for each process model and used as input to a logistic regression model. The logistic regression model learns the relationship between the input features (IC' and M') and a binary outcome variable representing correctness (e.g., whether the BPMN model is correct or not).

By learning from a dataset of labeled BPMN process models, a logistic regression model can then generalize its learning to classify new process models based on their IC' and M' scores, providing a predictive tool for assessing correctness in BPMN process modeling.

Thus, the algorithm for business process models classification based on their correctness will include the following steps (see fig. 3):

1) process a collection of BPMN model files, parse business process elements;
2) calculate the IC and M measures for each BPMN model in the dataset, find standardized values of the considered features;
3) create a dataset with the IC and M values for each BPMN model and the corresponding binary target variable indicating correctness;
4) apply logistic regression to the prepared training set (75 %), define a relationship between features (IC and M) and correctness labels;
5) evaluate the performance of the trained logistic regression model on a test set (25 %) to ensure that it can analyze new business process models;
6) deploy the logistic regression model to evaluate the correctness of new BPMN models based on their IC and M measures;
7) use the deployed logistic regression model to classify new BPMN models based on their IC and M scores.

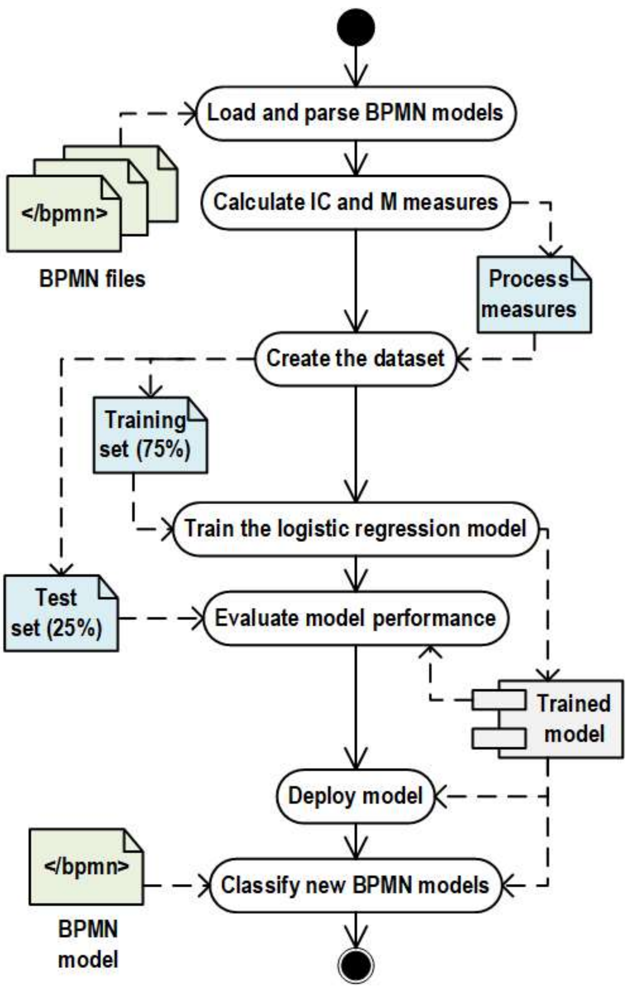

Fig. 3. Algorithm for business process models classification based on their correctness

System design. Functional requirements define the capabilities and functions a system must have to meet user expectations and business needs. They describe what the system should do and typically include functions such as data processing, computation, and user interaction [17].

Formally, the functional requirements for a software solution for business process models classification based on their correctness are presented in a UML (Unified Modeling Language) use case diagram (see fig. 4).

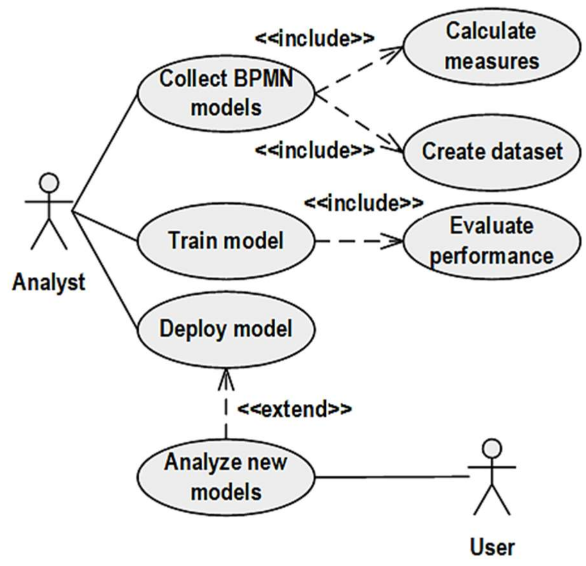

Fig. 4. Use case diagram of a software solution for business process models correctness classification

The selection of the technology stack for the BPMN model classification software tool was made with a focus on achieving a balance between efficiency, versatility, and usability:

- Node.js [18] is the backend JavaScript runtime, which facilitates parallel query processing.
- TensorFlow [19] is a powerful machine learning library with robust capabilities in implementing the logistic regression algorithm for model training and classification.
- Bpmn.io [20] is a BPMN modeling toolkit, which provides a user interface for model visualization.
- MongoDB is a NoSQL database [21], used for efficient data storage and retrieval, it meets scalability and flexibility requirements.

The software for BPMN diagrams classification based on their correctness is built using a three-tier client-server architecture that uses considered technologies to integrate with various system components (see fig. 5) [22].

Results and discussion. The collection of BPMN business process models based on the GitHub repository "bpmn-for-research" contains 3729 models [23].

Boxplot diagrams (fig. 6) were used to set thresholds for the IC and M measures. We focused on the values of the third quartile (Q3) obtained from the boxplot analysis.

By observing the Q3 values, the upper thresholds for IC and M were determined:

- $IC \leq 62$;
- $M \leq 19$.

Any data points that exceeded these thresholds were considered outliers, indicating potential inefficiencies in the BPMN models. Thus, a criterion was established by which BPMN models with IC or M values exceeding the upper threshold were classified as inefficient. This approach made it possible to identify and mark cases where high complexity or modularity could potentially hinder the efficiency or reliability of BPMN models.

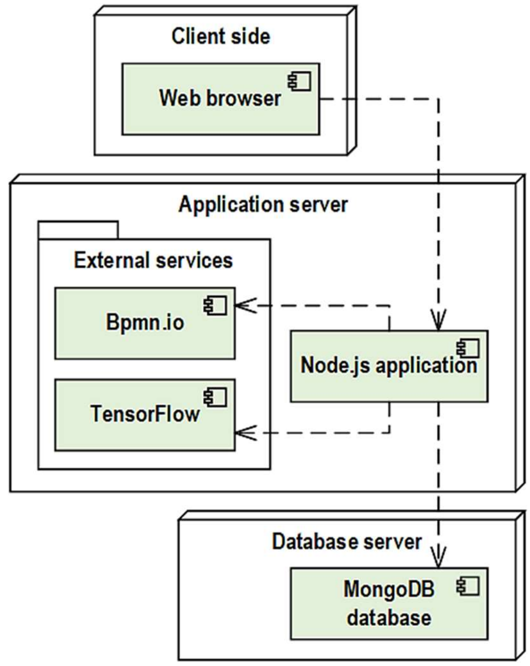

Fig. 5. Deployment diagram of a software solution for business process models correctness classification

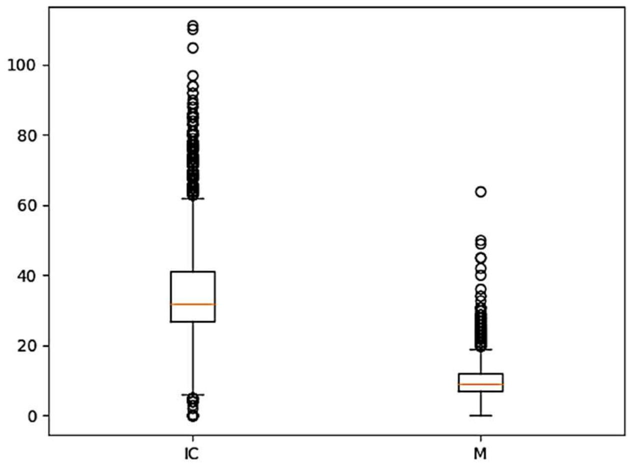

Fig. 6. Boxplot diagram to determine thresholds for IC and M measures

The provided confusion matrix (see fig. 7) illustrates the performance of a logistic regression classification model, typically in the considered BPMN models binary classification scenario. In this matrix [24]:

- top left cell (907) represents the number of True Negative (TN) results, indicating cases that the model correctly classified as negative;
- top right cell (1) demonstrates the number of False Positives (FP), which are cases that the model incorrectly classified as positive;
- bottom left cell (7) displays the number of False Negative (FN) results, representing cases that the model incorrectly classified as negative;
- bottom right cell (18) reflects the number of True Positive (TP) results, indicating cases that the model correctly classified as positive.

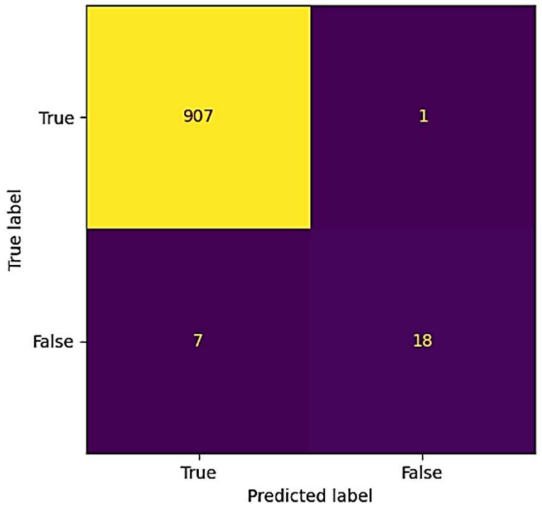

Fig. 7. Confusion matrix of the BPMN models classification

The results of the software solution for business process models classification based on their correctness demonstrate a high level of performance:

- out of 925 total predictions made, all were correct, indicating the effectiveness of the classification model in accurately identifying incorrect BPMN models;
- only 8 predictions were incorrect, indicating a low number of misclassifications and a high level of model accuracy;
- accuracy score of 0.9914 indicates the high ability of a software solution to correctly classify business process models in 99.14 % cases;
- precision score of 0.9988 indicates that when a solution predicts a model to be correct, it is correct 99.88 % of the time;
- recall value of 0.9923 indicates the classification model correctly identifies 99.23 % of the correct business process models.
- F-score of 0.9956 indicates the overall 99.56 % performance of the classification model and, respectively, the developed software solution.

The results of the classification of business process models in terms of their correctness based on logistic regression are demonstrated in fig. 8.

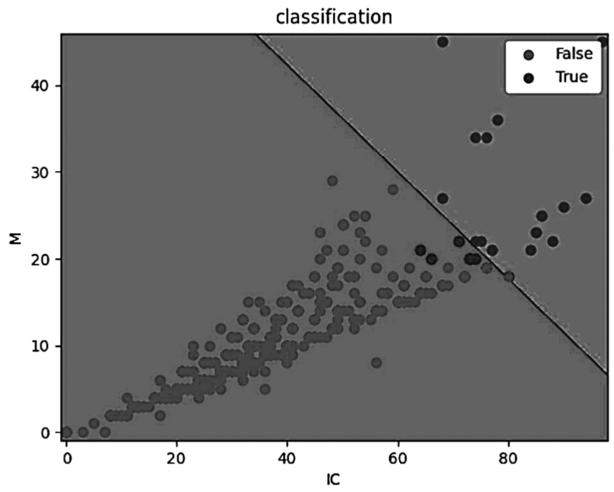

Fig. 8. Results of business process models correctness classification based on logistic regression

The deployed logistic regression model, implemented in the form of the developed software solution, was finally used to classify new BPMN models based on their IC and M scores (see fig. 9).

Conclusion. This paper solves the relevant practical problem of business process models classification based on their correctness, which allows to ensure quality control of the business process models being developed.

The goal of the study was to improve the quality of business process models by developing a software solution for business process models classification based on their correctness. Therefore, to achieve this goal, the following tasks were solved in this work:

1) the algorithm for solving the problem of business process models classification based on their correctness is developed;
2) the requirements for a software solution are determined;
3) the software solution development tools are selected;
4) the software solution for business process models classification based on their correctness is designed;
5) the corresponding software components of the proposed solution are developed;
6) the use of a software solution for solving the problem of business process models classification based on their correctness is demonstrated;
7) the obtained results are analyzed and discussed.

The software, based on the logistic regression model, indicates high performance of BPMN models classification based on their correctness, achieving a high accuracy rate of 99.14 %. A precision of 99.88 % demonstrates the ability to accurately identify correct BPMN models while keeping a high recall rate of 99.23 %, which indicates the ability to efficiently detect the relevant instances in the dataset. In addition, the F-score of 99.56 % highlights the overall high performance in BPMN models classification, which helps to minimize errors in designed business process models.

References

1. *Business Process*. URL: https://www.ibm.com/docs/en/bpm/8.5.0?topic=processes-business (access date: 15.03.2024).
2. *Business Process*. URL: https://www.gartner.com/en/information-technology/glossary/business-process (access date: 15.03.2024).
3. *Business Process*. URL: https://www.techopedia.com/definition/26363/business-process (access date: 20.03.2024).
4. Dumas M., La Rosa M., Mendling J., Reijers H. A. *Fundamentals of Business Process Management*. URL: https://doi.org/10.1007/978-3-662-56509-4 (access date: 20.03.2024).
5. Kopp A., Orlovskyi D. *Towards Intelligent Technology for Error Detection and Quality Evaluation of Business Process Models*. URL: https://ceur-ws.org/Vol-3373/keynote1.pdf (access date: 27.03.2024).
6. *Business Process Model and Notation (BPMN), Version 2.0*. URL: https://www.omg.org/spec/BPMN/2.0/PDF (access date: 28.03.2024).
7. von Rosing M., White S., Cummins F., de Man H. *Business Process Model and Notation (BPMN)*. URL: https://doi.org/10.1016/B978-0-12-799959-3.00021-5 (access date: 28.03.2024).
8. van der Aals W. M. P. *Business Process Modeling Notation*. URL: https://doi.org/10.1007/978-1-4614-8265-9_1195 (access date: 04.04.2024).
9. Kopp A., Orlovskyi D. *Towards the Software Solution for Complexity Minimization of Business Process Models to Improve*

Understandability. URL: https://ceur-ws.org/Vol-3426/paper22.pdf (access date: 04.04.2024).
10. Ayech H. B. H., Ghannouchi S. A., El Hadj Amor E. A. *Extension of the BPM lifecycle to promote the maintainability of BPMN models*. URL: https://doi.org/10.1016/j.procs.2021.01.239 (access date: 09.04.2024).
11. Tsakalidis G., Vergidis K. *A Roadmap to Critical Redesign Choices That Increase the Robustness of Business Process Redesign Initiatives*. URL: https://doi.org/10.3390/joitmc7030178 (access date: 09.04.2024).
12. Taherdoost H. *Classification of Machine Learning Algorithms*. URL: https://doi.org/10.1007/978-981-16-8403-6_38 (access date: 10.04.2024).
13. Sekeroglu B., Hasan S. S., Abdullah S. M. *Comparison of Machine Learning Algorithms for Classification Problems*. URL: https://doi.org/10.1007/978-3-030-17798-0_39 (access date: 10.04.2024).
14. Bisong E. *Understanding Logistic Regression*. URL: https://doi.org/10.1007/978-1-4842-4470-8_20 (access date: 12.04.2024).
15. Han W., Wang R., Huang D., Xu C. *Large-Scale ALS Data Semantic Classification Integrating Location-Context-Semantics Cues by Higher-Order CRF*. URL: https://doi.org/10.3390/s20061700 (access date: 15.04.2024).
16. Ahsan M. et al. *Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance*. URL: https://doi.org/10.3390/technologies9030052 (access date: 15.04.2024).
17. *Functional and Nonfunctional Requirements*. URL: https://doi.org/10.1007/978-1-4302-0528-9_4 (access date: 15.04.2024).
18. *Node.js*. URL: https://nodejs.org/en/ (access date: 18.04.2024).
19. *TensorFlow*. URL: https://www.tensorflow.org/ (access date: 18.04.2024).
20. *bpmn.io*. URL: https://bpmn.io/ (access date: 18.04.2024).
21. *MongoDB*. URL: https://www.mongodb.com/ (access date: 18.04.2024).
22. Reussner R. et al. *Managed Software Evolution*. URL: https://doi.org/10.1007/978-3-030-13499-0 (access date: 18.04.2024).
23. *BPMN for research*. URL: https://github.com/camunda/bpmn-for-research (access date: 20.04.2024).
24. Ming Ting K. *Confusion Matrix*. URL: https://doi.org/10.1007/978-0-387-30164-8_157 (access date: 21.04.2024).

References (transliterated)

1. *Business Process*. Available at: https://www.ibm.com/docs/en/bpm/8.5.0?topic=processes-business (accessed: 15.03.2024).
2. *Business Process*. Available at: https://www.gartner.com/en/information-technology/glossary/business-process (accessed: 15.03.2024).
3. *Business Process*. Available at: https://www.techopedia.com/definition/26363/business-process (accessed: 20.03.2024).
4. Dumas M., La Rosa M., Mendling J., Reijers H. A. *Fundamentals of Business Process Management*. Available at: https://doi.org/10.1007/978-3-662-56509-4 (accessed: 20.03.2024).
5. Kopp A., Orlovskyi D. *Towards Intelligent Technology for Error Detection and Quality Evaluation of Business Process Models*. Available at: https://ceur-ws.org/Vol-3373/keynote1.pdf (accessed: 27.03.2024).
6. *Business Process Model and Notation (BPMN), Version 2.0*. Available at: https://www.omg.org/spec/BPMN/2.0/PDF (accessed: 28.03.2024).
7. von Rosing M., White S., Cummins F., de Man H. *Business Process Model and Notation (BPMN)*. Available at: https://doi.org/10.1016/B978-0-12-799959-3.00021-5 (accessed: 28.03.2024).
8. van der Aals W. M. P. *Business Process Modeling Notation*. Available at: https://doi.org/10.1007/978-1-4614-8265-9_1195 (accessed: 04.04.2024).
9. Kopp A., Orlovskyi D. *Towards the Software Solution for Complexity Minimization of Business Process Models to Improve Understandability*. Available at: https://ceur-ws.org/Vol-3426/paper22.pdf (accessed: 04.04.2024).
10. Ayech H. B. H., Ghannouchi S. A., El Hadj Amor E. A. *Extension of the BPM lifecycle to promote the maintainability of BPMN models*. Available at: https://doi.org/10.1016/j.procs.2021.01.239 (accessed: 09.04.2024).
11. Tsakalidis G., Vergidis K. *A Roadmap to Critical Redesign Choices That Increase the Robustness of Business Process Redesign Initiatives*. Available at: https://doi.org/10.3390/joitmc7030178 (accessed: 09.04.2024).
12. Taherdoost H. *Classification of Machine Learning Algorithms*. Available at: https://doi.org/10.1007/978-981-16-8403-6_38 (accessed: 10.04.2024).
13. Sekeroglu B., Hasan S. S., Abdullah S. M. *Comparison of Machine Learning Algorithms for Classification Problems*. Available at:

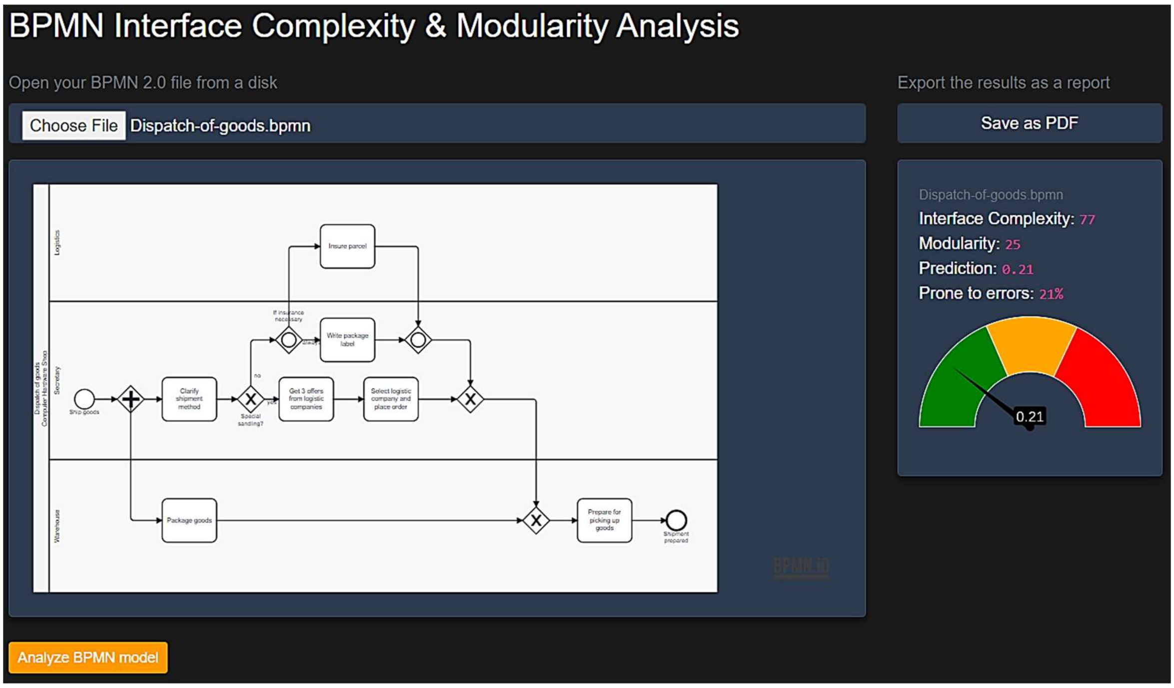

Fig. 9. Example of the new BPMN model assessment using the developed software

https://doi.org/10.1007/978-3-030-17798-0_39 (accessed: 10.04.2024).
14. Bisong E. *Understanding Logistic Regression*. Available at: https://doi.org/10.1007/978-1-4842-4470-8_20 (accessed: 12.04.2024).
15. Han W., Wang R., Huang D., Xu C. *Large-Scale ALS Data Semantic Classification Integrating Location-Context-Semantics Cues by Higher-Order CRF*. Available at: https://doi.org/10.3390/s20061700 (accessed: 15.04.2024).
16. Ahsan M. et al. *Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance*. Available at: https://doi.org/10.3390/technologies9030052 (accessed: 15.04.2024).
17. *Functional and Nonfunctional Requirements*. Available at: https://doi.org/10.1007/978-1-4302-0528-9_4 (accessed: 15.04.2024).
18. *Node.js*. Available at: https://nodejs.org/en/ (accessed: 18.04.2024).
19. *TensorFlow*. Available at: https://www.tensorflow.org/ (accessed: 18.04.2024).
20. *bpmn.io*. Available at: https://bpmn.io/ (accessed: 18.04.2024).
21. *MongoDB*. Available at: https://www.mongodb.com/ (accessed: 18.04.2024).
22. Reussner R. et al. *Managed Software Evolution*. Available at: https://doi.org/10.1007/978-3-030-13499-0 (accessed: 18.04.2024).
23. *BPMN for research*. Available at: https://github.com/camunda/bpmn-for-research (accessed: 20.04.2024).
24. Ming Ting K. *Confusion Matrix*. Available at: https://doi.org/10.1007/978-0-387-30164-8_157 (accessed: 21.04.2024).

Received 05.05.2024

УДК 004.94

А. М. КОПП, доктор філософії (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», завідувач кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: andrii.kopp@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-3189-5623
Д. Л. ОРЛОВСЬКИЙ, кандидат технічних наук (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», професор кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: dmytro.orlovskyi@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-8261-2988
Ю. С. ЛІТВІНОВА, кандидат технічних наук (PhD), Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: uliya.litvinova@khpi.edu.ua, ORCID: https://orcid.org/0000-0001-6680-662X

РОЗРОБКА ТА ДОСЛІДЖЕННЯ ПРОГРАМНОГО РІШЕННЯ ДЛЯ АНАЛІЗУ КОРЕКТНОСТІ МОДЕЛІ БІЗНЕС-ПРОЦЕСІВ З ВИКОРИСТАННЯМ МАШИННОГО НАВЧАННЯ

Погано розроблені моделі бізнес-процесів є джерелом помилок і подальших витрат, пов’язаних з цими помилками, таких як грошові витрати, втрачений час або навіть певний шкідливий вплив на людей або навколишнє середовище, якщо помилкові моделі бізнес-процесів пов’язані з критично важливими галузями. Життєвий цикл BPM (Business Process Management) зазвичай складається з проєктування, впровадження, моніторингу та контролю виконання бізнес-процесів, але йому бракує безперервного контролю якості створених моделей BPMN (Business Process Model and Notation). Таким чином, у даній роботі розглядається задача класифікації моделей бізнес-процесів на основі їх коректності, вирішення якої забезпечить контроль якості розроблених моделей бізнес-процесів. Таким чином, метою даного дослідження є підвищення якості моделей бізнес-процесів шляхом розробки програмного рішення для класифікації моделей бізнес-процесів на основі їх коректності. Об’єктом дослідження є процес класифікації моделей бізнес-процесів на основі їх коректності, який використовує показники якості та порогові значення, як правило, показники складності. Предметом дослідження є програмне рішення для класифікації моделей бізнес-процесів на основі їх коректності. Отже, у даній роботі запропоновано алгоритм розв’язання задачі класифікації моделей BPMN з використанням логістичної регресії, показників складності інтерфейсу та модульності, визначено вимоги до програмного забезпечення, обрано засоби розробки програмного забезпечення, спроєктовано програмне забезпечення для класифікації моделей бізнес-процесів на основі їх коректності, розроблено відповідні програмні компоненти, продемонстровано використання програмного рішення для розв’язання задачі класифікації моделей бізнес-процесів на основі їх коректності, а також проаналізовано та обговорено отримані результати. Розроблене програмне забезпечення демонструє високу ефективність класифікації моделей BPMN на основі їх коректності, досягаючи високих показників точності (99,14 %), влучності (99,88 %), повноти (99,23 %) та F-міри (99,56 %), що підкреслює високу ефективність виявлення помилок моделювання.

Ключові слова: моделювання бізнес-процесів, якість моделі, показники складності, машинне навчання, логістична регресія, програмне рішення.

Повні імена авторів / Author's full names

Автор 1 / Author 1: Копп Андрій Михайлович / Kopp Andrii Mykhailovych
Автор 2 / Author 2: Орловський Дмитро Леонідович / Orlovskyi Dmytro Leonidovych
Автор 3 / Author 3: Літвінова Юлія Сергіївна / Litvinova Uliya Serhiivna