УДК 004.912

ALGORITHM FOR COMPREHENSIBILITY EVALUATION OF BUSINESS PROCESS MODELS USING NATURAL LANGUAGE PROCESSING

Shevelev V.D.[1], Kopp A.M.[2]

[1,2] National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine
ORCID: [1]http://orcid.org/0009-0005-2904-3420, [2]http://orcid.org/0000-0002-3189-5623
E-mail: vadym.sheveliev@cs.khpi.edu.ua[1], andrii.kopp@khpi.edu.ua[2]

DOI: https://doi.org/10.15673/atbp.v15i4.2721

Анотація. *Дане дослідження присвячене розв'язанню задачі оцінки якості моделей бізнес-процесів, зокрема, оцінці зрозумілості текстових позначень в цих моделях. Метою роботи є підвищення зрозумілості текстових позначень в моделях бізнес-процесів для покращення загальної якості цих моделей. Ця проблема є актуальною, оскільки погано зрозумілі моделі бізнес-процесів можуть спричиняти помилки при виконанні описаних бізнес-процесів або при їх аналізі для покращення процесів. Більше того, такі помилки можуть знизити ефективність бізнес-процесів, якщо залучені сторони неправильно розуміють необхідні завдання, і призвести до фінансових втрат або ще більш небезпечних наслідків для критично важливих бізнес-процесів. Таким чином, для досягнення поставленої мети необхідно створити алгоритмічне забезпечення для розв'язання поставленої задачі. В даній роботі визначено питання, які існуючі дослідження не вирішували або не розкривали належним чином, а також основні недоліки існуючих програмних рішень, призначених для розв'язання поставленої задачі. Запропонований алгоритм базується на методах обробки природної мови, таких як токенізація та визначення частин мови. Запропонований алгоритм враховує рекомендований «дієслівно-об'єктний» стиль маркування текстових міток діяльностей в моделях бізнес-процесів. За допомогою Python-реалізації запропонованого алгоритмічного рішення оцінено зрозумілість прикладу моделі бізнес-процесу. Проаналізовано отримані результати, зроблено висновки та сформульовано напрямки подальших досліджень.*

Abstract. *This research paper is devoted to solving the problem of assessing the quality of business process models, in particular, evaluating the comprehensibility of text labels in these models. The purpose of the work is to increase the understandability of text labels in business process models to improve the overall quality of these models. This problem is relevant, since poorly understandable business process models may cause errors when described workflows are executed, or when they are analyzed for process improvement. Moreover, such errors may decrease the performance of business processes if involved parties misunderstand required tasks and lead to monetary losses or even more dangerous consequences for critical business processes. Thus, to achieve the goal, it is necessary to create an algorithmic solution to solve the specified problem. In this work, the issues that existing studies did not resolve or did not reveal properly, and the main shortcomings of the existing software solutions that should solve the problem mentioned above are identified. The proposed algorithm is based on the natural language processing techniques, such as tokenization and part of speech tagging. The proposed algorithm considers the recommended "verb-object" labeling style of activity text labels in business process models. The comprehensibility of a sample business process model is evaluated using the Python implementation of the proposed algorithmic solution. Obtained results are discussed, a conclusion is made, and future research directions are formulated.*

Keywords: business process model comprehensibility, activity labels evaluation, business process model quality, natural language processing.

Ключові слова: зрозумілість моделей бізнес-процесів, оцінювання міток діяльностей, якість моделі бізнес-процесу, обробка природної мови.

Introduction

Nowadays, many companies document their business processes with the help of conceptual models. These models provide a framework for activities related to the business process management life cycle, such as process analysis, process redesign, process evaluation etc [1]. Many modeling initiatives for these processes have resulted in hundreds of models created by designers, all with varying backgrounds. One of the main obstacles to more effective use of these process models is the insufficient quality assurance. This lays the foundation for identifying automated analysis methods that can

provide such quality assurance.

In practice, a significant percentage of business process models have quality problems, with often 5% to 30% of models having reliability problems. The reason for at least some of them is the growing number of initiatives to model business process. Such poor development causes problems even at the stage of model creation and maintenance. More and more employees are involved in the modeling process. Many of these casual architects lack modeling experience and training, so the newly created models are not always of good quality.

Hence, poorly designed business process models, which are not understandable by involved participants and other stakeholders may lead to errors when such business processes are executed or analyzed with the purpose of further improvement. These errors caused by misunderstanding can cause workflow execution errors followed by monetary expenses or even literally disaster consequences for critical business processes that affect human activity and nature.

In addition, the fact that many companies maintain several thousand models requires automatic quality assurance, which is largely absent in modern tools. A promising direction for improving the quality of the process model is the automatic verification of business process models and their refactoring.

Related work analysis and problem statement

There are already many different solutions that try to automate the process of assessing the quality of business process models. One such solution is BPMNspector.

BPMNspector is a tool that provides static analysis for BPMN 2.0 process models [2]. It checks individual files or entire directories of BPMN (Business Process Model and Notation) files and reports violations of BPMN 2.0 restrictions.

BPMNspector currently supports the following:

- checking the scheme (ensuring correctness according to official specifications);
- reference checking (ensuring that all references used exist and that only valid types are used);
- verification of additional restrictions;
- import (checking the import of processes, WSDL and XSD files);
- automatic corrections of various violations.

Because BPMNspector uses Gradle, only Java 8 installation is needed. The required libraries are downloaded and configured on the fly. To use BPMNspector, running the initial script is only needed. When finished, an HTML report will automatically open. After that, all reports are stored in a separate directory [3].

Another solution is the BPMN Scorecard Validation Tool, which allows users to check the quality of the business process model for free. This tool is the industry benchmark for BPMN model quality. Using it, users can evaluate their business process models to develop reliable and error-free diagrams.

BPMN Scorecard is a free tool that automates multiple validations of Bizagi's business process modeling best practices. Using this validation tool, users can obtain a higher quality BPMN business process model and improve their BPMN business process modeling skills. This free tool is available as a lightweight web application for a web browser. After completing the questionnaire, a certain BPMN business process model will receive a score from 0 (bad) to 100 (excellent) [4].

The BPMN Quality Tool referred in [5] is a tool for assessing the quality of business process models. BPMN Quality is developed in Java and mainly consists of four modules: BPMN elements extractor, capability builder, metrics calculator, and interpreter.

The extractor receives as input BPMN converted to XMI (XML Metadata Interchange). This conversion aims to obtain BPMN in a standard exchange format. The information received by this module includes all elements contained in BPMN. The tool can be integrated into any other modeling tool that supports the standard.

The information provided by the extractor is passed to the capability constructor. Given an opportunity selected by the user, this module generates a tree from the BPMN data containing all elements belonging to that perspective.

The metrics calculator is a module that implements all metrics calculation algorithms. It uses the information provided by the capability designer and the metric category selected by the user to calculate and display the metric values corresponding to those selections and saves them as an XML (eXtensible Markup Language) file. The structure of this file is a DTD (Document Type Definition) derived from the proposed classification system

On the other hand, the interpreter module compares the obtained results with the threshold values of the metrics entered by the user. According to the quality parameters associated with the selected metrics, the interpreter gives an assessment of the quality of the analyzed business process.

After analyzing the solutions discussed above, it can be seen that none of them consider the clarity of text labels in business process models as a factor that affects the overall quality of these models. In addition, most of these tools have certain limitations, that is, in order to use them, it is necessary to download and install not only the programs themselves, but also external dependencies. Finally, not all of these solutions can work specifically with files in the BPMN format.

Returning to the work of [5], we can also consider the basis on which they rely when creating their software solution, namely metrics that can be used to measure the quality of business process models. In general, the purpose of this document is to classify the quality indicators proposed at the moment as a structure defined in terms of design perspectives and to implement this structure as a tool that helps in assessing the quality of business process models.

The framework for classifying quality indicators takes into account BPMN capabilities (e.g., informational, functional, organizational, and behavioral). This provides the designer with more efficient use of quality indicators. Indeed, depending on a particular point of view, they will examine only a subset of the metrics relevant to their own point of view: the selected metrics are defined in terms of the elements of the business process that are of particular interest. In

addition, when a decision is made, for example, to restructure a model in order to improve a certain metric, the designer knows the impact of his decision on other metrics related to other perspectives. That is, the BPMN elements involved in the studied metric provide traceability between different perspectives.

Several researchers have already identified the potential of business process indicators. Most of them were adapted from the field of software development. Following the classification of software engineering, these quality indicators are divided into three categories: connectivity, cohesion, and complexity.

Representing processes from different perspectives is a vital modeling approach to understand their complexity. A specific perspective can be developed based on analytical criteria or subjective needs. For example, a model must be able to provide various information elements, such as the activities that comprise a process, who performs these activities, when and where these activities are performed etc.

Another important work dealing with the quality of business process models offers a set of measures that can be used to assess the structural complexity of business process conceptual models. The main goal is to obtain useful metrics for use in the maintenance tasks of these models, hence to obtain higher quality models by early estimation of given model quality properties. Thanks to a number of experiments, it is now possible to identify a set of measures that can be useful for evaluating usability and supporting conceptual models of business processes [6].

Empirical research revealed that process models in practice do not always meet the conditions for naming activities, such as the “verb-object” style. There are three general classes of activity marking styles [7]. First, the verb-object style defines an activity label as a verb followed by a corresponding business object. Secondly, there are different ways of defining an activity label in the form of an “action-noun”. For such a case, the action is not formulated as a verb, but rather as a gerund or predicate verb. There is also a third category of activity labels that do not have any reference to the activity. An example is “information system”, which does not refer to action either as a verb or as a noun.

Having defined these categories, it should be noted that marking style is a factor with characteristics that are quite different from structural indicators. Although metrics can be measured on a metric scale, marking styles can only be distinguished nominally. This means that, in the simplest case, the input variable can be defined in binary form, distinguishing between the use of the “verb-object” style and the use of another style. In terms of defining quality parameters, this makes the task easier. While metrics require a threshold to distinguish between good and bad, marking styles can be directly compared to be better or worse.

An experiment reported in [7] uses activity labels of different labeling styles as interpretations to explore their potential ambiguity and their utility in facilitating domain understanding. Variance analysis tests show that “verb-object” labels are perceived much better in this respect, followed by “action-noun” labels. The marks of all other categories were recognized as the most ambiguous.

Despite the fact that the use of labeling style is well covered in the literature, there are still various problems in working with the terminology. From a quality point of view, the terms should have a clear meaning. Therefore, synonyms and homonyms should be avoided when modeling a process. This problem is recognized in various papers, but there is currently no proper solution for automatic quality assurance.

Experiments and best practices in the field indicate that preference should be given to the “verb-object” marking style. The challenge here is to achieve an accurate analysis of the different marking styles so that they can be transformed. Accurate parsing of labels in English is difficult for two reasons. First, many nouns in English are derived from verbs in ways that are morphologically equivalent to those verbs. Second, the activity labels of the process model usually do not cover the complete grammatically correct sentence structure. These difficulties were found using standard Natural Language Processing (NLP) tools, such as the Stanford parser.

The approach referred in [8] uses various contextual information to match a label and its correct labeling style. Once the marking style is known, tools such as WordNet can be used to find a verb that corresponds to an action formulated as a noun. This approach has been shown to work accurately for three different simulation collections from practice, including a total of over 10,000 activity labels [8].

The state-of-the-art review was given in [9], where problems of the quality evaluation of text labels in business process models were outlined.

Therefore, the main research topic of this work is how NLP tools can be used to evaluate the comprehensibility of business process models by analyzing their activity labels. This will help to create clear workflow descriptions understandable by process participants, owners, analysts, and other stakeholders to prevent process errors.

Research goal and tasks

The growth of many process modeling initiatives involving dozens of architects with varying expertise in building and maintaining thousands of models raises the question of how quality assurance can be defined and implemented in an automated manner. Understanding the comprehensibility factors of a process model provides the basis for creating such automatic methods.

The object of research is the process of assessing the quality of business process models based on their syntactic characteristics.

The subject of the research is an algorithm for evaluating the comprehensibility of text labels in business process models.

The purpose of the study is to increase the understandability of text labels in business process models to improve the overall quality of these models.

The tasks that must be solved to achieve the goal are as follows:

- create an algorithmic solution to solve the given problem;
- create the software implementation of the proposed algorithm;
- check the functionality of the proposed algorithm using the developed software implementation.

Materials and methods

Let us describe the business process model given using the BPMN notation as the following tuple:

$$BPMN = \langle N, A, L, \varphi \rangle, \quad (1)$$

where:

- N is the set of business process model elements (or "nodes");
- A is the set of business process model sequence flows (or "arcs");
- L is the set of text labels of business process model elements and sequence flows;
- $\varphi : N \cup A \to L$ is the function that defines the mapping between business process model objects (elements and sequence flows) and their text labels.

Therefore, the set of business process model elements (1) can be decomposed into respective subsets:

$$N = F \cup E \cup G, \quad (2)$$

where:

- F is the set of business process activities that includes tasks and sub-processes;
- E is the set of business process events that includes start events, end events, and intermediate events;
- G is the set of business process gateways that define logical rules (AND, OR, and XOR) of the workflow splits and joins.

Thus, for each activity (2) we have to define its text label:

$$\varphi_F : F \to L_F \subseteq L, \quad (3)$$

where:

- φ_F is the function that defines the mapping between business process model activities and their text labels;
- L_F is the subset of text labels that belong to business process model activities.

Hence, according to the proposed approach, when processing a BPMN file it is necessary to extract the set of text labels that belong to business process model activities (3):

$$L_F = \{l_{Fi}, i = \overline{1,n}\}, \quad (4)$$

where n is the number of business process activities and, consequently, the number of textual labels that match these activities (3).

Therefore, the classification of activity labels (4) by their correspondence to the "verb-object" style can be formally defined using the following formula:

$$\rho : L_F \to \{0,1\}, \quad (5)$$

where:

- $\rho(l_{Fi}) = 1$ if l_{Fi} label starts with the verb, $i = \overline{1,n}$;
- $\rho(l_{Fi}) = 0$ if l_{Fi} label does not start with the verb, $i = \overline{1,n}$.

For example, when using the WordNet corpus [6] mentioned above, the mapping function (5) can be described using the following formula:

$$\forall i = \overline{1,n} : \rho(l_{Fi}) = \begin{cases} 1, & \omega(t_1^{Fi}) = 1, \\ 0, & \omega(t_1^{Fi}) = 0, \end{cases} \quad (6)$$

where:

- t_1^{Fi} is the first element of the set $T_{Fi} = \{t_j^{Fi}, j = \overline{1,m_{Fi}}\}$ obtained after the label l_{Fi}, $i = \overline{1,n}$ tokenization, which consists of m_{Fi} elements;
- $\omega : T_{Fi} \to \{0,1\}$ is the function that defines the mapping between single tokens, into which labels l_{Fi}, $i = \overline{1,n}$ were split, and binary values: 1 – for verbs and 0 – for other parts of speech.

Finally, using (6), the degree of business process model comprehensibility can be defined using the following formula:

$$C(L_F) = \frac{1}{|L_F|} \sum_{i=1}^{n} \rho(l_{Fi}). \quad (7)$$

The degree (7) produces values in the range between 0 and 1, where 0 signalizes the very bad correspondence of business process activity labels to the "verb-object" labeling style and, therefore, the very bad comprehensibility. Whereas, 1 signalizes the very good correspondence of business process activity labels to the "verb-object" labeling style and, respectively, the very good comprehensibility.

Thus, the initial algorithmic solution, which can be proposed in this study, is the following:

- collect all elements of "Task" type from the BPMN file;
- get text labels of all collected elements;

- split each text label into separate words, getting the first one for each label;
- check if each first word is a verb with the help of NLP tools;
- get the total correspondence to verb value for all first words;
- find the average correspondence value for the whole BPMN model.

The algorithm of the proposed solution is given in Fig. 1 below.

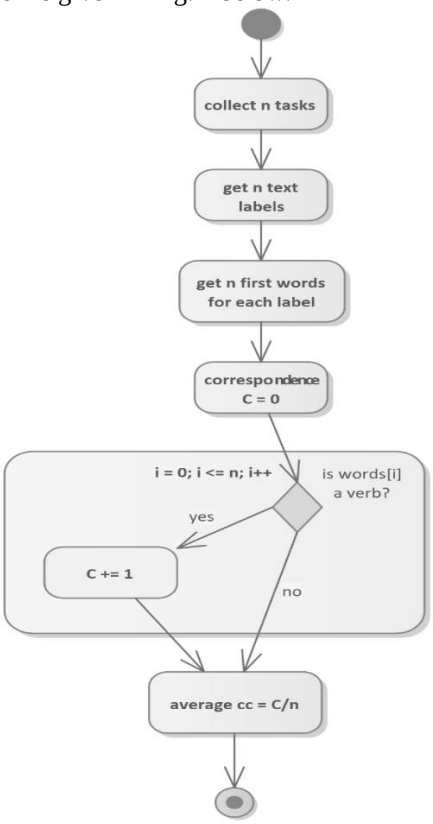

Fig. 1. Algorithm for comprehensibility evaluation of business process models

The long-term goal of this project is to design and develop a software tool for end-users capable of processing business process models in the BPMN format and evaluating the quality of their text labels.

The software must extract all names of "Task" elements from the BPMN file. Thus, for the business process model, a set of activity names that they contain will be obtained. After that, the program checks the compliance of each activity name with the recommended "verb-object" format using NLP tools and calculates the quality score of the text labels.

Then the obtained value can be interpreted as the degree of compliance of activity text labels in the business process model with the rules, which affects the comprehensibility of the business process model during its further use in organizational design or software development.

To collect and specify requirements and document them, an interview method was used, where the interested party was asked certain questions. As a result, the answers to these questions were used to form the software requirements specification.

The main goal of this project is to improve the quality assurance process of BPMN models by evaluating the comprehensibility of text activity labels. To automate this process and reduce the number of errors in each model, we need to create a universal software capable of handling labels regardless of their structure and language.

Thus, it was found that the future software solution has the following functional requirements:

- REQ-1: the system should have a field where the user can upload a BPMN file;
- REQ-2: the system should have a button that starts the analysis of the attached file;
- REQ-3: the system should present the results of the analysis, showing whether they meet the standard;
- REQ-4: the system should store the analysis results in the database for further use;
- REQ-5: the system should suggest how action names can be changed to conform to the standard.

In addition, the system also has the following non-functional requirements:

- PE-1: the system must handle large BPMN files that contain up to 100 different activities;
- PE-2: the system should be able to process any BPMN file for no longer than 1 minute;
- PE-3: the system should start up completely in no longer than 2 seconds;
- PE-3: the system should generate evaluation results in no longer than 5 seconds;
- PE-4: the system must save the analysis results to the database for no longer than 1 second;
- SE-1: the system should not allow direct access to database data, but only provide it through the system's user interface;

– SE-2: the user must log in to one of the accounts to access all system functions.

The system has a single basic user role (Fig. 2), which allows access to all of its functionality.

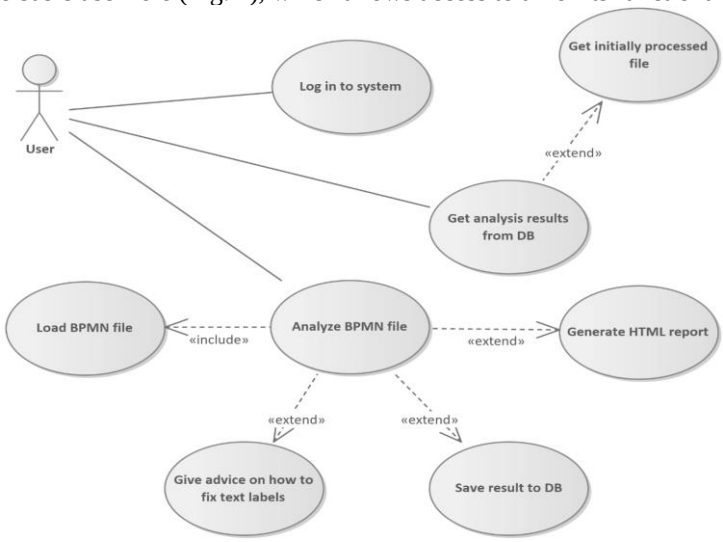

Fig. 2. UML use case diagram

The basic usage scenario is as follows: a user downloads a BPMN file and begins the process of evaluating the quality of activity labels; when the process is complete, the system presents the evaluation results and stores them in the database for further investigation. Optionally, the system can provide advice to the user on how to correct the activity labels so that they are correct.

The user interface (UI) of the main working area should allow users to attach BPMN files, click the respective button to analyze uploaded business process models, and get the estimated text labels quality with the list of incorrect labels that do not follow the "verb-object" naming style (Fig. 3).

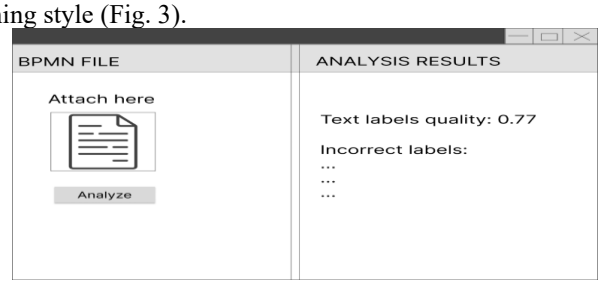

Fig. 3. Expected UI of the main working area

Research results and discussion

Let us use business process BPMN models available in the Camunda GitHub repository [10] to perform experimental calculations. All of these models were created by participants of Camunda's training sessions. One of these BPMN models, which describes the goods dispatch business process, is given in Fig. 4 below.

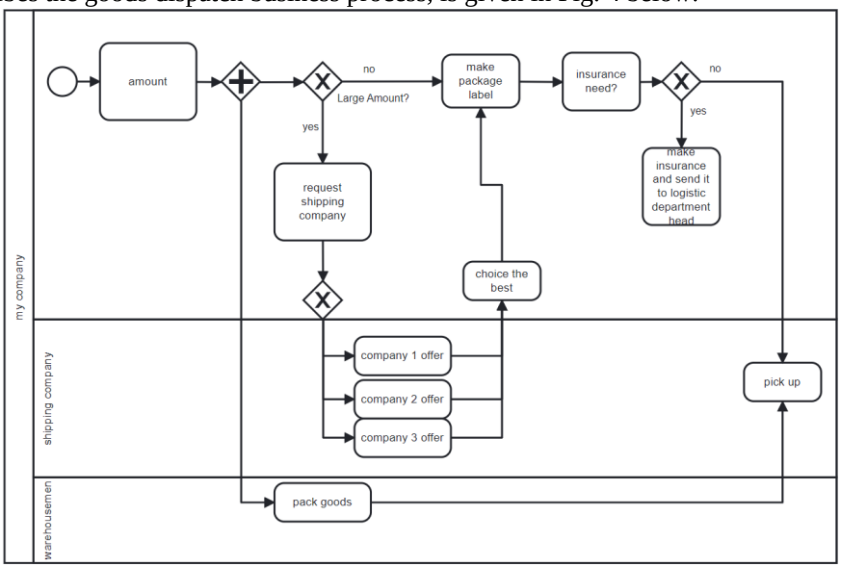

Fig. 4. BPMN model example

Experimental calculations are handled using the Python programming language with the NLTK (Natural Language Toolkit) [11] package. In the prototype algorithm's implementation, we do not process BPMN files directly, but use the prepared list of business process activity labels (Fig. 5).

```
from nltk.tokenize import word_tokenize
from nltk.corpus import wordnet
```

```
bpmn_activities = ['amount', 'request shipping company',
                   'pack goods', 'company 1 offer',
                   'company 2 offer', 'company 3 offer',
                   'make package label', 'choice the best',
                   'insurance need?',
                   'make insurance and send it to logistic department head',
                   'pick up']
print(bpmn_activities)
```

```
['amount', 'request shipping company', 'pack goods', 'company 1 offer', 'compan
y 2 offer', 'company 3 offer', 'make package label', 'choice the best', 'insura
nce need?', 'make insurance and send it to logistic department head', 'pick u
p']
```

Fig. 5. Business process activity labels preparation

Then, using Python's NLTK [11] package we apply the proposed approach by tokenizing activity labels, extracting their first words, and using the WordNet, similarly to the idea proposed in [12] for similarity measurement of food entities, for checking if the activity label begins with the verb (Fig. 6).

```
pos_first_words = []

for activity in bpmn_activities:
    words = word_tokenize(activity)
    first_word = words[0]
    is_verb = False
    synsets = wordnet.synsets(first_word)

    for synset in synsets:
        if synset.pos() == 'v':
            is_verb = True
            break

    pos_first_words.append((first_word, is_verb))

print(pos_first_words)
```

```
[('amount', True), ('request', True), ('pack', True), ('company', True), ('comp
any', True), ('company', True), ('make', True), ('choice', False), ('insuranc
e', False), ('make', True), ('pick', True)]
```

Fig. 6. Experimental implementation of the proposed approach

Finally, the degree of business process model comprehensibility is calculated using the formula (7) as the ratio between the number of activity labels that correspond to the "verb-object" labeling style and the total number of activities present in the processed BPMN model (Fig. 7).

```
verb_object_correspondence = 0

for pos_first_word in pos_first_words:
    if pos_first_word[1]:
        verb_object_correspondence += 1

verb_object_correspondence /= len(pos_first_words)
verb_object_correspondence = round(verb_object_correspondence, 2)
print(verb_object_correspondence)
```

```
0.82
```

Fig. 7. Evaluation of the business process model comprehensibility

According to the obtained results (Fig. 7), the degree of comprehensibility of the considered goods dispatch BPMN model (Fig. 4) is 0.82. This means that labels of only 82% of business process activities correspond to the "verb-object" naming style [5] recommended for achieving better clarity of business process models and, hence, better understandability for their readers (process participants, owners, managers, and other stakeholders). Thus, the proposed algorithm (Fig. 1) can be used in the software tool for the comprehensibility analysis of BPMN business process models.

Conclusions

In this paper, we considered the problem of comprehensibility evaluation of business process models and proposed the respective algorithmic solution. Therefore, the following conclusions can be formulated:

– poorly described business process models can become sources of workflow errors, because process participants, owners, and analysts, may not clearly understand depicted workflows;

– errors caused by misunderstood business process scenarios may lead to monetary losses or, even worse, more serious damage to society or the environment if these are critical business processes;

– the state-of-the-art analysis shows that most popular tools for quality analysis of BPMN models do not consider their textual quality, while the "verb-object" style is considered as the most appropriate way to label business process activities;

– the comprehensibility evaluation of business process models is possible using NLP techniques, such as tokenization and part-of-speech tagging, so the corresponding algorithm was proposed and its efficiency was verified on the sample BPMN model;

– experimental results show that the degree of business process model comprehensibility can be used to estimate the general clarity of business process models, as well as to compare BPMN models that describe the same workflow;

In the future, formulated software requirements and the proposed algorithm will be used to develop the software tool for end-users to help them build high-quality and understandable BPMN models. Also, the limitation related to the improper part of speech recognition should be considered to improve the accuracy of the proposed approach.

Список використаних джерел

1. Grisold T., Groß S., Stelzl K., vom Brocke J., Mendling J., Röglinger M., Rosemann M. The five diamond method for explorative business process management. *Business & Information Systems Engineering*. 2022. Vol. 64, No. 2. P. 149–166.
2. Corradini F., Muzi C., Re B., Rossi L., Tiezzi F. BPMN 2.0 OR-Join Semantics: Global and local characterisation. *Information Systems*. 2022 Vol. 1, No. 105. P. 101934.
3. Static Analysis of BPMN 2.0 process models. URL: http://bpmnspector.org/ (дата звернення: 30.01.2023)
4. BPMN Scorecard Validation Tool. URL: https://freebpmnquality.github.io/bpmn-scorecard.html (дата звернення: 30.01.2023)
5. Fakorede O., Davies P. Quantitative Measurable Concepts to Visualize Business Process Improvement. *Proceedings of the 3rd International Conference on Big Data Technologies*. 2020. P. 147–154.
6. Wang W., Chen T., Indulska M., Sadiq S., Weber B. Business process and rule integration approaches – An empirical analysis of model understanding. *Information Systems*. 2022. Vol. 104. P. 101901.
7. Avila D. T., dos Santos R. I., Mendling J., Thom L. H. A systematic literature review of process modeling guidelines and their empirical support. *Business Process Management Journal*. 2020. Vol. 27, No. 1. P. 1–23.
8. Sola D., van der Aa. H., Meilicke C., Stuckenschmidt H. Exploiting label semantics for rule-based activity recommendation in business process modeling. *Information Systems*. 2022. Vol. 108. P. 102049.
9. Kopp A., Orlovskyi D., Orekhov S. Towards Understandability Evaluation of Business Process Models using Activity Textual Analysis. *CEUR Workshop Proceedings*. 2022. Vol. 3312. P. 200–211.
10. BPMN for research. URL: https://github.com/camunda/bpmn-for-research (дата звернення: 30.01.2023)
11. NLTK: Natural Language Toolkit. URL: https://www.nltk.org/ (дата звернення: 30.01.2023)
12. Butt S., Bakhtyar M., Noor W., Baber J., Ullah I., Ahmed A., Basit A., Kakar M. S. Semantic similarity based food entities recognition using WordNet. *Journal of Intelligent & Fuzzy Systems*. 2022. Vol. 43, No. 2. P. 2069–2078.

References

1. T. Grisold, S. Groß, K. Stelzl, J. vom Brocke, J. Mendling, M. Röglinger, and M. Rosemann, "The five diamond method for explorative business process management," *Business & Information Systems Engineering*, vol. 64, no. 2, pp. 149–166, 2022.
2. F. Corradini, C. Muzi, B. Re, L. Rossi, and F. Tiezzi, "BPMN 2.0 OR-Join Semantics: Global and local characterisation," *Information Systems*, vol. 1, no. 105, pp. 101934, 2022.
3. "Static Analysis of BPMN 2.0 process models." Bpmnspector. http://bpmnspector.org/ (accessed: Jan. 30, 2023)
4. "BPMN Scorecard Validation Tool." Freebpmnquality. https://freebpmnquality.github.io/bpmn-scorecard.html (accessed: Jan. 30, 2023)
5. O. Fakorede, P. Davies, "Quantitative Measurable Concepts to Visualize Business Process Improvement," in *Proceedings of the 3rd International Conference on Big Data Technologies*, pp. 147–154, 2020.
6. W. Wang, T. Chen, M. Indulska, S. Sadiq, B. Weber, "Business process and rule integration approaches – An empirical analysis of model understanding," *Information Systems*, vol. 104, pp. 101901, 2022.
7. D. T. Avila, R. I. dos Santos, J. Mendling, L. H. Thom, "A systematic literature review of process modeling guidelines and their empirical support," *Business Process Management Journal*, vol. 27, no. 1, pp. 1–23, 2020.
8. D. Sola, H. van der Aa, C. Meilicke, H. Stuckenschmidt, "Exploiting label semantics for rule-based activity recommendation in business process modeling," *Information Systems*, vol. 108, pp. 102049, 2022.
9. A. Kopp, D. Orlovskyi, S. Orekhov, "Towards Understandability Evaluation of Business Process Models using Activity Textual Analysis," *CEUR Workshop Proceedings*, vol. 3312, pp. 200–211, 2022.
10. "BPMN for research." Github. https://github.com/camunda/bpmn-for-research (accessed: Jan. 30, 2023)
11. "NLTK: Natural Language Toolkit." Nltk. https://www.nltk.org/ (accessed: Jan. 30, 2023)
12. S. Butt, M. Bakhtyar, W. Noor, J. Baber, I. Ullah, A. Ahmed, A. Basit, M. S. Kakar, "Semantic similarity based food entities recognition using WordNet," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2069–2078, 2022.

Отримана в редакції 01.11.2023. Прийнята до друку 04.12.2023. Received 01 November 2023. Approved 12 December 2023. Available in Internet 03 January 2024