Method for Automated Test Tasks Creation for Educational Materials

Olexander Barmak[1[0000-0003-0739-9678]], Olexander Mazurets[1[0000-0002-8900-0650]], Iurii Krak[2[0000-0002-8043-0785]], Anatolii Kulias[3[0000-0003-3715-1454]]

[1]Khmelnytskyi National University, Khmelnytskyi, Ukraine
alexander.barmak@gmail.com, exechong@gmail.com
[2]Taras Shevchenko National University of Kyiv, Ukraine
yuri.krak@gmail.com
[3]Glushkov Cybernetics Institute, Kyiv, Ukraine
kulyas@nas.gov.ua

Abstract. The article considers the method of automated creation of test tasks for educational materials, which does not require additional formalization of educational materials and uses the production model of knowledge representation for presentation of rules for creation of test tasks. A sufficient requirement for the formalization of educational materials is the presence in the input document of text content in the form of symbols and preferably a structure in the form of headings in the file. As a result of the method of automated creation of test tasks receive the set of test tasks that are different in parameters (type of question, number of correct answers, the rule behind which the test task is formed, terms used in the task, etc.) and can be used to check the level of knowledge by existing educational environments and testing systems. The set contains test tasks that semantically, structurally, and parametrically cover the corresponding input informational education material. An important feature of the developed method is the binding of the created test tasks to all levels of the semantic structure of the informational education material, which ensures its complete coverage and enables adaptive control of the level of acquired knowledge.

Keywords: test tasks, information technology, keywords, key terms

1 Introduction

Computer-based means of knowledge testing play an important role in addressing the problem of effective control of the level of acquired knowledge that emerges with the development of new technologies and education [1–4]. Computer-based testing is one of the main ways of controlling knowledge in educational information systems, such as "Moodle" [5] or "ATutor" [6]. Information technologies make it possible to significantly reduce labor costs for the creation of test tasks with the possibility of their constant updating, which forms the current direction of scientific research [7–10].

Education course of discipline uses informational education material (IEM) as the main information in the education course and test education material (TEM) necessary to assess the level of knowledge of IEM. TEM contains test exercises of varying complexity which allow to determinate the level of knowledge of IEM, identify gaps in knowledge, causes of wrong actions of the subject studying the education course [11]. In the context of narrow specialization of courses, their numbers and intensive updating, the only way to provide courses of disciplines with representative and discriminatory TEM is to automate the formation of sets of test tasks.

Many scientific works are devoted to various aspects of testing, development and application of educational and testing environments using modern information technologies, and to the development of software systems of knowledge level testing [12–15]. Most of them carried out research in the field of testing, filling out the database of test tasks with the help of means to support manual creation of test tasks, assessment of complexity of test tasks, safety of the process of testing and reproduction of results.

Among the known means of automation of the creation of test tasks, it is necessary to note the method of parameterized tasks, the method of generation of test tasks by conceptual-thesis model and the method of generation of test tasks by formalization of structured text elements including classification and clustering ones for effective data processing. The solutions developed are effective for use in certain cases, but require substantial and time-consuming preparation of IEM. Much of the content of the IEM of many courses is predominantly textual content, which is characterized by consistency and semantic coherence of the presentation [16]. This feature cuts the way to the development of a method for the automated creation of test tasks, which does not require significant pre-processing of IEM.

The purpose of the work is to develop a method of automated creation of test tasks for educational materials, which does not require additional formalization of educational materials and uses a production model of knowledge representation to represent the rules of formation of test tasks. The output data of the method is a set of test tasks and metadata necessary to bind the created test tasks to the levels of the semantic structure of the educational materials, which provides a complete coverage of the educational material and enables adaptive control of the level of acquired knowledge.

2 Information Model of the Semantic Structure of the Educational Course

The method of automated creation of test tasks for educational materials determines the elements of a number of sets of information model of the semantic structure of educational course. The information model of the semantic structure of educational course is a complete representation of the semantic structure of the educational course [11]. The model is formalized by presenting part of the course elements as a set of entities (headings, key terms, test tasks, relations). The structure of the educational course (EC) is presented in the form of:

$$\{IEM, TEM\} \subset EC, \tag{1}$$

where *IEM* – informational education material), *TEM* – testing education material.

The semantic structure of the *EC* as a union of *IEM* and *TEM* is represented as:

$$IEM \bigcup TEM = \{M_{Heading} \bigcup M_{Term} \bigcup M_{TestEx} \bigcup M_{\mathrm{Re}\,l}\}, \tag{2}$$

where $M_{Heading}$ – set of headings, M_{Term} – set of key terms, M_{TestEx} – set of test tasks, M_{Rel} – set of relations.

According to the types of elements in (2) that are related with the elements of set M_{Rel}, its structure can be represented as:

$$M_{\mathrm{Re}\,l} = M_{\mathrm{Re}\,l:H-H} \bigcup M_{\mathrm{Re}\,l:H-T} \bigcup M_{\mathrm{Re}\,l:H-TE} \bigcup M_{\mathrm{Re}\,l:T-TE}, \tag{3}$$

where $M_{Rel:H-H}$ – set of relations between headings and headings, $M_{Rel:H-T}$ – set of relations between headings and key terms, $M_{Rel:H-TE}$ – set of relations between headings and test tasks, $M_{Rel:T-TE}$ – set of relations between key terms and test tasks.

From here, according to (2) and (3), the semantic structure of the IEM and TEM education course is (Fig. 1):

$$IEM \bigcup TEM = (M_{Heading} \bigcup M_{Term} \bigcup M_{TestEx} \bigcup M_{\mathrm{Re}\,l:H-H} \bigcup M_{\mathrm{Re}\,l:H-T} \bigcup M_{\mathrm{Re}\,l:H-TE} \bigcup M_{\mathrm{Re}\,l:T-TE}). \tag{4}$$

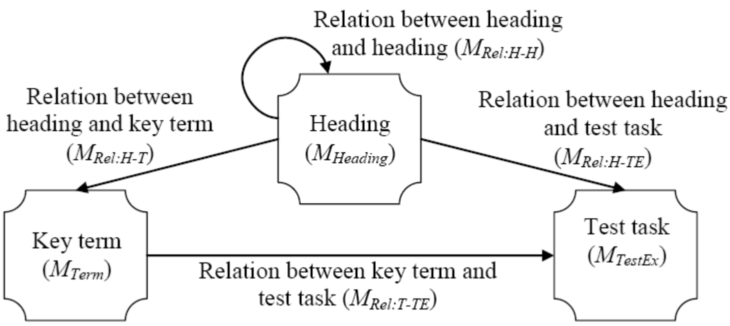

Fig. 1. Relationships between model parameters

The semantic structure of the *IEM* can be represented as:

$$(M_{Heading} \bigcup M_{Term} \bigcup M_{\mathrm{Re}\,l:H-H} \bigcup M_{\mathrm{Re}\,l:H-TE}) \subset IEM \subset EC. \tag{5}$$

The semantic structure of the *TEM* can be represented as:

$$(M_{Heading} \bigcup M_{Term} \bigcup M_{TestEx} \bigcup M_{\mathrm{Re}\,l:H-TE} \bigcup M_{\mathrm{Re}\,l:T-TE}) \subset TEM \subset EC. \tag{6}$$

Each of the *IEM* and *TEM* components in the *EC* model has its own representation and structure.

Previous works have considered the method of forming the structure of educational materials and searching for key terms in them [9], which defines the elements of the sets *IEM*: $M_{Heading}$ – set of headings, M_{Term} – set of terms, $M_{Rel:H-H}$ – set of relations between headings and headings, $M_{Rel:H-T}$ – set of relations between headings and key terms [11]. This data [17] is input to the method of creation of test tasks for educational materials, which finds the elements of the following sets of *TEM*: set of test tasks M_{TestEx}, set of relations between headings and test tasks $M_{Rel:H-TE}$, set of relations between key terms and test tasks $M_{Rel:T-TE}$.

Each element of the set of test tasks $m_{TestEx} \in M_{TestEx}$ is a cortege of the form:

$$m_{TestEx} = (ID, Type, TEContent, Answers)\,, \tag{7}$$

where *ID* – unique identifier of the element, *Type* – type of question, *TEContent* – content of the test task, *Answers* – number of answers.

Each element of the set of relation between headings and test tasks $m_{\mathrm{Re}\,l:H-TE} \in M_{\mathrm{Re}\,l:H-TE}$ is a cortege of the form:

$$m_{\mathrm{Re}\,l:H-TE} = (3, Obj1, Obj2, Cont)\,, \tag{8}$$

where 3 – identifier for this type of relation; *Obj*1– first entity of the relation, the element of the set of $M_{Heading}$; *Obj*2 – second entity of the relation, an element of the set M_{TestEx}; *Cont* – number of the sentence used.

Each element of the set of relation between key terms and test tasks $m_{\mathrm{Re}\,l:T-TE} \in M_{\mathrm{Re}\,l:T-TE}$ is a cortege of the form:

$$m_{\mathrm{Re}\,l:T-TE} = (4, Obj1, Obj2, Loc)\,, \tag{9}$$

where 4 – identifier for this type of relation; *Obj*1– first entity of the relation, an element of the set of key terms M_{Term}; *Obj*2 – second entity of the relation, an element of the set M_{TestEx}; *Loc* – numeric indicator obtained from the use of the rule of creation of test tasks and is a pointer to the type or place of use of the term in the test task.

3 Production Rules for the Creation of Test Tasks

Procedural knowledge or rules are a set of procedures for transforming knowledge as data [19].

The production model best reflects the procedural nature of knowledge [20]. The basic constructive element of such model is the production rule [21–23], which can be represented as follows: IF <condition> THAN <action>, so the rule consists of a conditional and an effective part.

The condition (premise, antecedent) is some sentence-template, which is used for searching, and action (result, conclusion, consequence) is an algorithm for converting the sentence into the content of the components of the test task, which are executed with successful search results (Fig. 2).

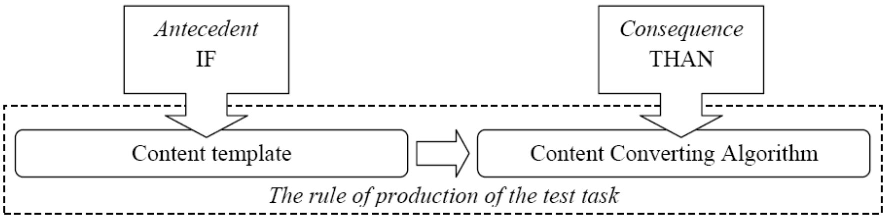

Fig. 2. Scheme of production rule for the creation of test tasks

An example of a production rule for the creation of a test tasks prototype can be shown in Fig. 3. In this case, the antecedent defines three requirements for the sentence, in the case of simultaneous implementation of which the rule is activated. When using an antecedent, the active term and the set of relation fragments are used. The consequence determines the four-step sequence required to formulate the content of the test task. When applying the consequence, the content of the sentence, the active term, and the set of key terms for the given fragment are used.

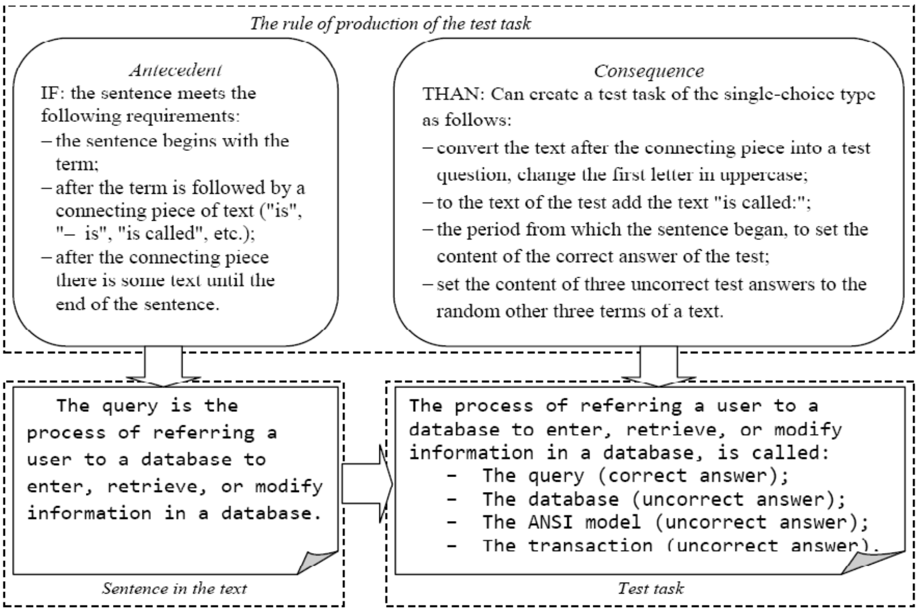

Fig. 3. An example of the production rule for generating of test task prototype

Thus, set of production rules M_{Rule} is the primary mechanism for creating test tasks. Each rule of production of the test tasks $\forall m_{Rule} \in M_{Rule}$ is a cortege of two elements – the antecedent and the consequence that form the implication:

$$m_{Rule} = (a \Rightarrow c), \quad (10)$$

where a – antecedent of the rule, c – consequence of the rule.

A set of 2 antecedents has been created to describe all the sentences that are potentially suitable for creating test tasks. The set of 11 consequences covers all algorithms for creation test tasks of the types used in educational environments: logical type, single choice, multiple choice and text input tasks. Sets of antecedents and consequences form a set of 17 possible production rules that allow to create all possible test tasks for all potentially suitable sentences.

4 Scheme of the Method of Automated Creation of Test Tasks

The scheme of the method of automated creation of test tasks is presented in Fig. 4. The input data of the method is the content of information educational material or its defined element of the $M_{Heading}$ structure and the corresponding set of key terms M_{Term} $\cup$ $M_{Rel:H\text{-}T}$; the output data is a set of test tasks M_{TestEx}, as well as a sets of relations between headings and test tasks $M_{Rel:H\text{-}TE}$ and between terms and test tasks $M_{Rel:T\text{-}TE}$. The method requires many rules for the production of test tasks M_{Rule}, created separately and in advance.

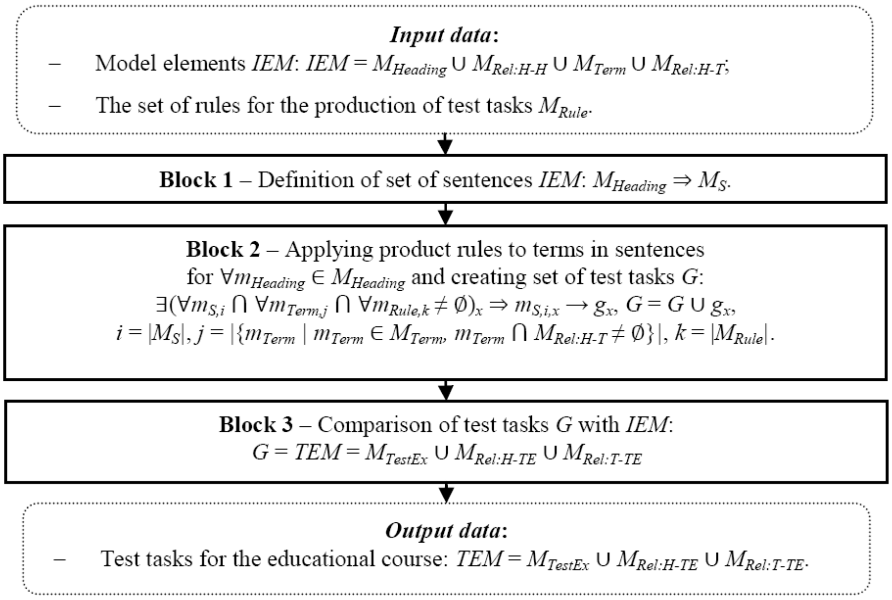

Fig. 4. General scheme of the method of automated creation of test tasks

First (*Block 1*), by parsing the content of the selected IEM element (the *HContent* attribute of the elements of set $M_{Heading}$), a set of fragments M_S is formed, each of which is a sentence or in some cases (like lists) a set of sentences. The fragments localize potential content to create separate test tasks.

To create a set of test tasks G (*Block 2*), each element $m_S \in M_S$ from each heading of the document $m_{Heading} \in M_{Heading}$ is checked for the presence of each key term

$m_{Term} \in M_{Term}$, mapped to this heading $m_{Term} \bigcap M_{\mathrm{Re}\,l:H-T} \neq \varnothing$. If the term m_{Term} is present in the fragment m_S, then the production rules M_{Rule} are searched for compliance with the antecedent of the rule. Every case of compliance $\exists\left(\forall m_{S,i} \bigcap \forall m_{Term,j} \bigcap \forall m_{Rule,k} \neq \varnothing\right)_x$ results in the automatic creation of a new test task g_x. The effective part of the production rule (consequence) defines the algorithm for converting the content of the fragment m_S into a test problem task content g.

By searching the fragments, terms, and production rules, the antecedent of the selected rule is searched for in the IEM content fragment. If compliance is established – a new test task is formed in accordance with the consequence of this rule, is checked for the minimum number of elements required for this type of test task, and is added to the set of test tasks M_{TestEx}. Then (*Block 3*), the corresponding relations are formed as elements of the set of relations between headings and test tasks $M_{Rel:H\text{-}TE}$ and the set of relations between terms and test tasks $M_{Rel:T\text{-}TE}$, which is necessary for further adaptive testing [24].

The output data of the method of automated creation of test tasks are all attributes of elements of sets of TEM: $G = TEM = M_{TestEx} \bigcup M_{\mathrm{Re}\,l:H-TE} \bigcup M_{\mathrm{Re}\,l:T-TE}$.

Method of automated creation of test tasks allows to create test tasks of the types used in educational environments: logical type, single choice, multiple choice and text input tasks.

Each test task has a set of answers, each of which has parameters: content of the answer, evaluation of the answer. The answer estimation is automatically determined from the following calculation. If the maximum base for the correct answer to the test task is B and the number of correct answers is N_{True}, then each correct answer receives a $B_{True} = B / N_{True}$ score. Accordingly, if the number of false answers is N_{False}, then each false answer receives a score of $B_{False} = -B / N_{False}$.

To improve the generated test tasks, it is possible to manually edit the content of each test task and to automatically edit the total number of test tasks.

5 Method Efficiency Research – Content Coverage

The research was conducted to determine the part of the IEM content, that is used to create the test tasks, and accordingly the level of knowledge that can be verified created using this method. 203 elements of the test sample educational materials were used for the research. The test software was designed for the automated formation of test tasks. It has been estimated that on average in 97.8% of cases of presences of a key term in the content at least one test task of each type are created (Fig. 5).

The calculation of the average number of rules used for the production of test tasks for the presences of key terms in the content of IEM on a test sample of 203 documents get the results shown in Table 1 (97.8% of cases were taken into account, when the presence of the term triggered at least one antecedent of product rules).

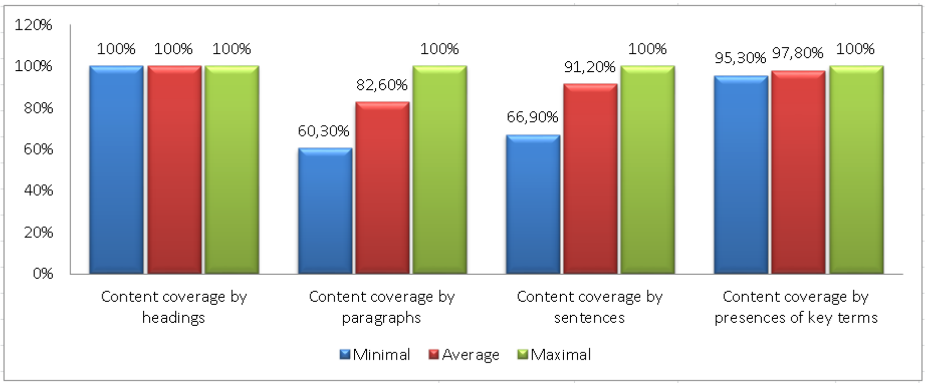

Fig. 5. Average and limit values of IEM content coverage by test tasks set

Table 1. Number of created test tasks (parentheses indicate the number of product rules used) by types and in general by the presences of key terms in IEM content.

№	*Type of test task*	*Number of test tasks created, pcs.*		
		Minimal	*Average*	*Maximal*
1	Logical type	2 (2)	2,23 (2,28)	8 (3)
2	Single choice	2 (2)	3,19 (2,61)	17 (3)
3	Multiple choice	1 (1)	1,95 (1)	5 (1)
4	Text input	1 (1)	1,06 (1)	3 (1)
5	General by one presence	6 (6)	8,43 (6,87)	14 (8)

Combining the same antecedents and different consequences in the production rules reaches the minimal required equality of distribution of test tasks by type and by key terms used.

6 Method Efficiency Research – Further Actions

A separate research was conducted to determine the difference in time required to creation a set of test tasks, by determining the difference between the time required for this job to perform it manually and the time required to obtaining a similar result by using the developed method and successive manually correction the automatically created test tasks set. The research material used elements of disciplines in the test sample, total 41 samples. The task is to develop the same number of test tasks for each sample of IEM (40–100 tasks). For the automated creation of test tasks, a software product developed according to the following method (Fig. 6) was used. Moodle was used to manually development and correction of the test tasks. The subjects of the work on the development and correction of the test tasks were used by teachers of the KhNU (total 5 persons).

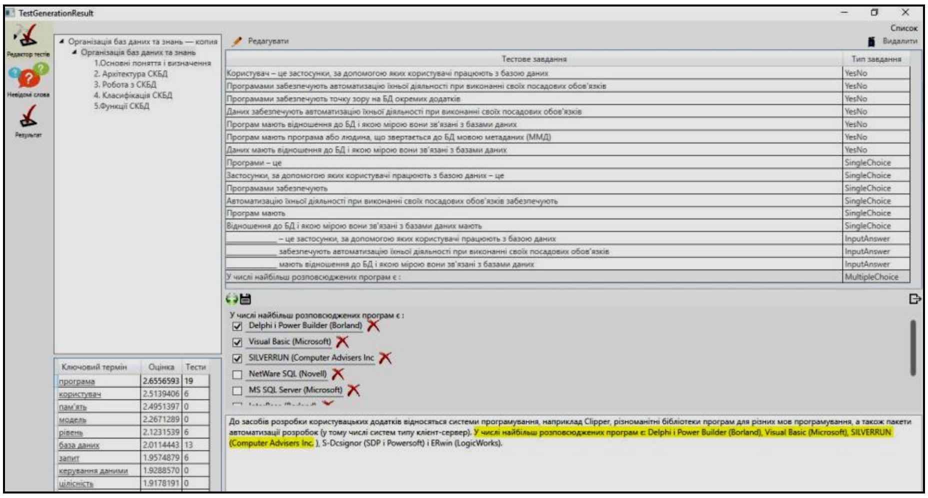

Fig. 6. Software system interface for automated test creation

The following results obtained. The average value of the effect of reducing the time on the creation of a set of test tasks was 60.25%, minimum value was 26.91% and maximum value was 74.53% (Fig. 7).

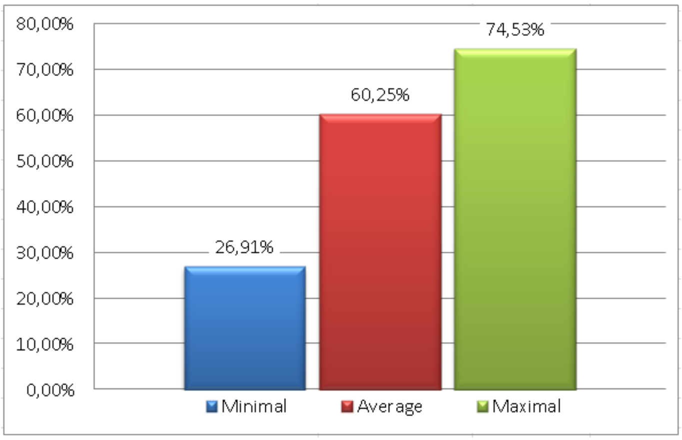

Fig. 7. The average and the limit values of the reducing the time for creation of test tasks sets

The average percentages of the number of test tasks in the further actions of the test developer was: 23.65% included in the resulting set without changes, 27.14% included in the resulting set and was manually adjusted, 49.21% deleted as incorrect or redundant (Fig. 8).

The average effect of reducing the time for creation of test tasks set 60.25% indicates that the creation of test tasks set using the developed method and subsequent manual adjustment of the automatically created test tasks set allows to reach the goal more quickly.

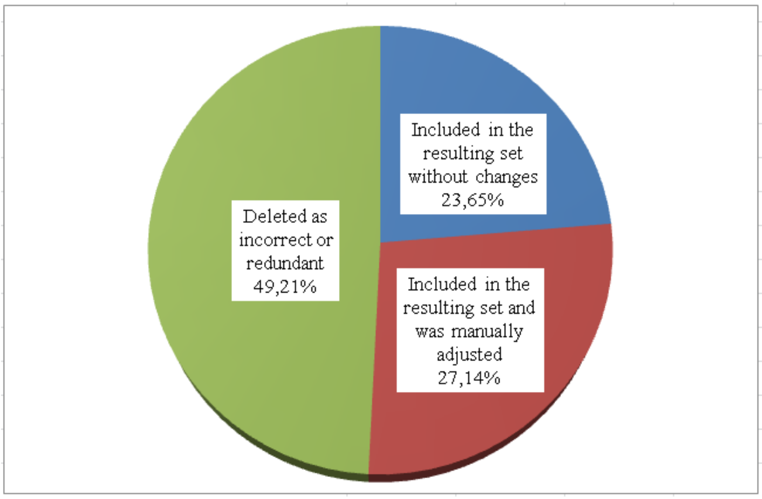

Fig. 8. Average percentages of the number of test tasks by the further actions of test developer

The available percentage of the number of deleted test tasks 49.21% is explained mainly by the excess number of automated test tasks created, rather than their incorrectness, since in no case did the teacher need to create new test tasks. The correction made to a number of automated test tasks of 27.14% concerned mainly the syntactic alignment of test task elements and editing of distractors. However, a significant number of test tasks, 23.65%, which is 46.56% of the total number of tasks in the resulting set, were used without adjustments and changes.

7 Conclusion

The method of automated creation of test tasks for educational materials is considered, which does not require additional formalization of educational materials and uses the production model of knowledge representation for presentation of rules for creation of test tasks. A sufficient requirement for the formalization of educational materials is the presence in the input document of text content in the form of symbols and preferably a structure in the form of headings in the file. As a result of the method of automated creation of test tasks receive the set of test tasks that are different in parameters (type of question, number of correct answers, the rule behind which the test task is formed, terms used in the task, etc.) and can be used to check the level of knowledge by existing educational environments and testing systems. The set contains test tasks that semantically, structurally, and parametrically cover the corresponding input IEM.

An important feature of the developed method is the binding of the created test tasks to all levels of the semantic structure of the IEM, which ensures its complete coverage and enables adaptive control of the level of acquired knowledge.

References

1. Yang, C., Potts, R., Shanks, D. R.: Enhancing learning and retrieval of new information: a review of the forvard testing effect. Science of Learning, 3: 8 (2018)
2. Chen, J., Dosyn, D., Lytvyn, V., Sachenko, A.: Smart Data Integration by Goal Driven Ontology Learning, Advances in Big Data, 529, pp. 283-292 (2016).
3. Brusilovsky, P., Rollinger, C., Peyl, C.: Adaptive and Intelligent Technologies for Web-based Education. Special Issue on Intelligent Systems and Teleteaching, Konstliche Intelligenz 4, pp. 19-25 (1999).
4. Saed, M. R.: Methods and Applications for Advancing Distance Education Technologies: International Issue and Solutions. IGI Global (2009)
5. Moodle – Open-source learning platform. https://moodle.org/
6. ATutor – Expert Tutoring Advice. https://atutor.ca/
7. Krak, I., Barmak, O., Mazurets, O: The practice investigation of the information technology efficiency for automated definition of terms in the semantic content of educational materials. CEUR Workshop Proceedings, 1631, pp. 237-245 (2016)
8. Carpenter, S. K., Pashler, H., Wixted, J. T., Vul, E.: The effects of tests on learning and forgetting. Memory & Cognition, 36, pp. 438–448 (2008)
9. Cho, K. W., Neely, L. H., Crocco, S., Virando, D.: Testing enhances both encoding and retrieval for both tested and untested items. Quarterly Journal of Experimental Psychology, 70(7), pp. 1211–1235 (2017)
10. Pasichnyk, R., Melnyk, A., Pasichnyk, N., Turchenko, I.: Method of adaptive control structure learning based on model of test's complexity. Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS'2011, 2, pp. 692-695 (2011)
11. Barmak, O., Mazurets, O., Krak, I., Kulias, A., Smolarz, A., Azarova, L., Gromaszek, K., Smailova, S.: Information technology for creation of semantic structure of educational materials. Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments, 11176 (2019) https://doi.org/10.1117/12.2537064
12. Gutl, Ch., Lankmayr, K., Weinhofer, J., Hofler, M.: Enhanced Approach of Automatic Creation of Test Items to foster Modern Learning Setting. Electronic Journal of e-Learning, 9, pp. 23–38 (2011)
13. Gierl, M. J., Lai, H., Hogan, L. B., Matinovic D.: A Method for Generation Educational Test Items that are Aligned to the Common Core State Standards. Journal of Applied Testing Technology, 16(1), pp. 1-18 (2015)
14. Kryvonos, I. G., Krak, I. V., Barmak, O. V., Kulias, A. I.: Methods to Create Systems for the Analysis and Synthesis of Communicative Information. Cybernetics and Systems Analysis, 53(6), pp. 847–856 (2017)
15. Manziuk, E. A., Barmak, O. V., Krak, Iu. V., Kasianiuk, V. S.: Definition of information core for documents classification. Journal of Automation and Information Sciences, 50(4), pp. 25-34 (2018)
16. Aggarwal, C. C., Zhai, C.: Text Data. Springer (2012).

17. Krak, Y., Barmak, O., Mazurets, O.: The practice implementation of the information technology for automated definition of semantic terms sets in the content of educational materials, CEUR Workshop Proceedings, 2139, pp. 245-254 (2018).
18. Barmak, O., Krak, I., Mazurets, O., Pavlov, S., Smolarz, A., Wojcik, W.: Research of Efficiency of Information Technology for Creation of Semantic Structure of Educational Materials. Advances in Intelligent Systems and Computing, 1020, pp. 554-569 (2020)
19. Liu, H., Gegov, A., Cocea, M.: Rule-based systems: a granular computing perspective. Granular Computing, 4, pp. 259-274 (2016)
20. Moghimi, M., Varjani, A. Y.: New rule-based phishing detection method. Expert systems with applications, 53, pp. 231-242 (2016)
21. Hu, X., Pedrycz, W., Castillo, O., Melin, P.: Fuzzy rule-based models with interactive rules and their granular generalization. Fuzzy Sets and Systems, 307, pp. 1-28 (2017)
22. Baki, I., Sahraoui, H.: Multi-Step Learning and Adaptive Search for Learning Complex Model Transformations for Examples. ACM Transaction on Software Engineering and Methodology, 25(3), pp. 1-37 (2016)
23. Kehrer, T., Alshanqiti, A., Heckel R.: Automatic inference of rule-based specifications of complex in-place model transformations. International Conference on Theory and Practice of Model Transformations, pp. 92-107 (2017)
24. Durlach, P. J., Lesgold, A. M.: Adaptive Technologies for Training and Education. Cambridge University Press (2012)