Developing the OntoMath^{Edu} Ecosystem for Educational Applications

Konstantin S. Nikolaev, Olga A. Nevzorova and Marina V. Falileeva

Kazan Federal University, Kremlyovskaya ul., 18, Kazan, 420008, Russia

Abstract

In the modern world, remote technologies have gained great popularity and demand. This happened due to the fact that many educational institutions were forced to switch to a distance learning format in response to the epidemiological situation. Kazan Federal University has a distance learning system based on LMS Moodle. As a component of this system for teaching school mathematics, the OntoMath^{Edu} ecosystem is being developed, the main goal of which is to personalize the process of distance learning in school mathematics. The article also describes the generation of some types of tasks in school geometry, based on the structure and concepts of the ontology of the same name, which contains a cross-section of knowledge on planimetry. The types of tasks covered in this article are as follows: theoretical tasks with the choice of the correct answer, an interactive task to restore the relationships between a certain set of ontology concepts, as well as a task to choose a suitable relationship between two concepts.

In addition, one of the methods for improving the structure of the ontology is highlighted to further improve the quality of the task generator and other components of the ecosystem.

Keywords

Paper template, OntoMath^{Edu} ontology, software service, educational application, test generator, test question.

1. Introduction

As in any large university, Kazan Federal University has its own distance learning system. Our goal is to expand the functionality of this system by adding components based on the ontology of school mathematics OntoMath^{Edu}.

The ontological approach is often used to organize training in a more structured format, as well as to help the student form a holistic picture of the current discipline, to form in the student's mind the relationship between concepts and rules.

There are a number of projects that use ontologies as a means of structuring educational content. For example, Conde et al. [1] developed a project in which educational ontology is used to describe the topics to be mastered by students and the pedagogical relationships between the topics.

In the MKMSE project [2], the ontology is used to store mathematical knowledge. In [3], the ontology is used to represent the semantic description of document resources and the relationship between document resources and other data.

Since the main practical result of this article is the generation of tasks in school mathematics, it is necessary to consider works aimed at generating tasks or components of tasks (for example, correct and incorrect answers for a pre-known question). We are interested in works that perform such generation in automatic mode, since it is very time-consuming to manually create large collections of tasks.

One of the promising approaches to automatic question generation is corpus approach, in which questions are generated from a set of annotated or non-annotated texts. This approach is widely used in the field of language learning [4–8] and some other fields, such as biology and medicine [9].

DTTL-2021: International Workshop on Digital Technologies for Teaching and Learning, March 22-28, 2021, Kazan, Russia
EMAIL: konnikolaeff@yandex.ru(K. S. Nikolaev); onevzoro@gmail.com(O. A. Nevzorova); mmwwff@yandex.ru(M. V. Falileeva)
ORCID: 0000-0003-3204-238X(K. S. Nikolaev); 0000-0001-8116-9446(O. A. Nevzorova); 0000-0003-2228-7551(M. V. Falileeva)

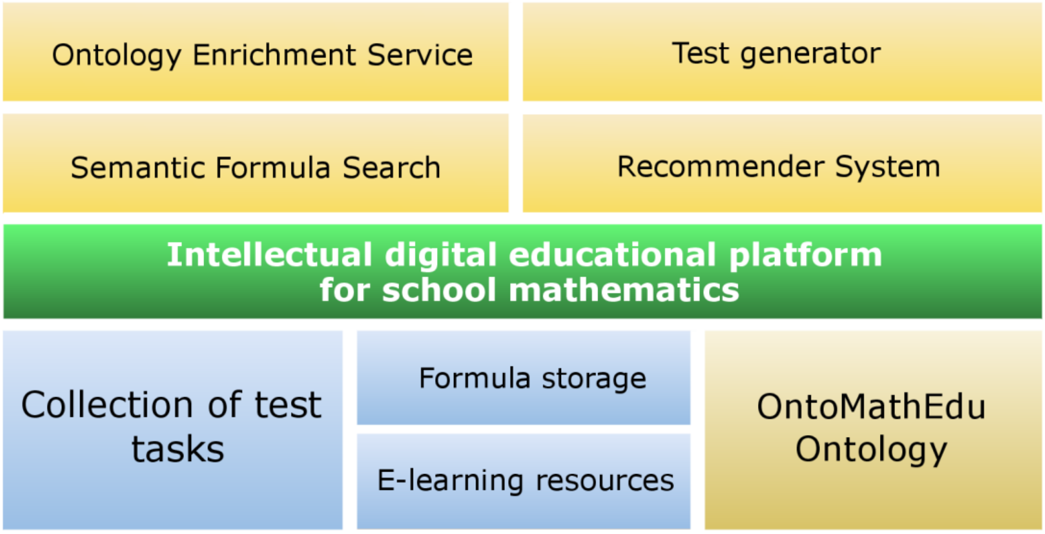

Figure 1. Components of the OntoMath^{Edu} ecosystem

However, as far as we know, corpus approaches were not used in the problems of generating mathematical tasks. Of course, there are works in which the generation is based on a set of templates [10–17] or manually created knowledge bases [18, 19].

2. Structure of the OntoMath^{Edu} ecosystem

The article discusses the structure and main tasks of the infrastructure of software components built around the ontology of school mathematics OntoMath^{Edu} [20, 21]. The OntoMath^{Edu} ontology is a set of concepts and relations between them that describe the necessary knowledge in the course of school planimetry. The main components of the OntoMath^{Edu} ecosystem are shown in Figure 1. We will briefly describe each of them:

1. Intellectual digital educational platform for school mathematics. This component is central to the ecosystem. It is used in teaching school mathematics, with the help of ontological and semantic technologies.
2. Collection of questions. This collection is the output data of the "Test generator" component. Questions stored in this collection are used in the formation of sets of questions when monitoring the level of knowledge of students.
3. Formula storage [22]. This component consists of formulas extracted from school geometry textbooks, presented in various formats (plain text, LaTeX, OpenMath). The conversion of formulas from plain text to formalized formats is done manually. Importantly, the concepts included in the formalization in OpenMath contain references to the corresponding concepts in the OntoMath^{Edu} ontology.
4. Digital educational resources. This component combines all auxiliary data sources located on the Internet
5. OntoMath^{Edu} ontology [20, 21]. The OntoMath^{Edu} ontology is a reflection of the level of knowledge corresponding to the level of school mathematics. It serves as the main repository of concepts and their relationships that are used by other services. The concepts of the ontology are expressed by the English, Russian and Tatar labels and were interlinked with the external lexical resources from the Linguistic Linked Open Data cloud [23], including, WordNet [24], BabelNet [25], RuThes Cloud [26] and Russian-Tatar Thesaurus [27].
6. Ontology enrichment service. This component includes a set of methods that are used to clarify the relationships between ontology concepts and improve the horizontal connectivity of ontologies.
7. Test generator [28]. This component is used to automatically create new tasks based on the analysis of the structure and concepts of existing tasks. In addition, this module has an ability to generate tasks based solely on the structure and concepts of the ontology.

8. Semantic formula search [29]. This is a search module that performs a semantic search for mathematical texts present in the ecosystem.
9. Recommendation system [30]. This module helps users to learn concepts related to their current learning process, thereby ensuring consistent learning.

3. Types of tasks

One of the most important components of the OntoMath^{Edu} ecosystem is the "Test generator". This component will automatically create tasks that belong to the types listed in the classification below.

As a classifier of test tasks, we have developed a set of correlations between the types of tasks from LMS Moodle and the levels of acquisition, according to which all questions can be divided into five types (Table 1).

It is obvious that questions requiring a detailed open answer are the most difficult type for automation and require the participation of the applicant. Therefore, at this stage, the result of the test component of the question generator will be questions of the types "Multiple Choice", "Matching task", "Open Answer", "Calculated answer".

In addition to the above classification, we also divided the questions according to the level of assimilation of mathematical entities (axioms, definitions, theorems, mathematical problems).

In addition to the main types of questions listed in Table 1, we have also developed several types of tasks based on the structure and concepts of ontology, namely:

1. Interactive task on restoring the structure between ontology concepts;
2. The task of determining the type of connection between pairs of concepts.

The first task is formed as follows:

1. A concept that is explicitly present in the hierarchical structure of the ontology is randomly selected from the ontology.
2. The subtree of the ontology is selected, the root of which is the selected concept (the depth and maximum width are set by the parameters).
3. The student is offered a tree structure of a subtree with empty nodes, and a list of concepts that need to be placed in the corresponding nodes. The order of concepts of the same level and concepts in the same branch is not checked (since the student is not required to know the order of these concepts in the ontology).
4. Upon completion of the concept placement, the solution is automatically checked.
5. If the student has made a random mistake and wants to cancel the placement of the concept, he has the option to click on the posted concept and return it to the list of unplaced concepts. Figure **2** shows an example of a partially solved task of the 1st type.

The second task is formed as follows:

1. Several random triplets of the <subject – predicate – object > format are selected from the ontology, without a limitation on predicate type.
2. For the student, pairs of concepts are displayed, separated by a drop-down list with all possible types of predicates present in the ontology.
3. The student selects the predicate types for all the given pairs of concepts and runs the check.

Figure 3 shows an example of a solved task of type 2.

Table 1
Types of tasks (according to our classification)

Type of formalized question (required answer)	Level of acquisition
Multiple choice	student (basic)
Matching task	student (basic), algorithmic level
Open answer	algorithmic, heuristic level
Calculated answer	creative level

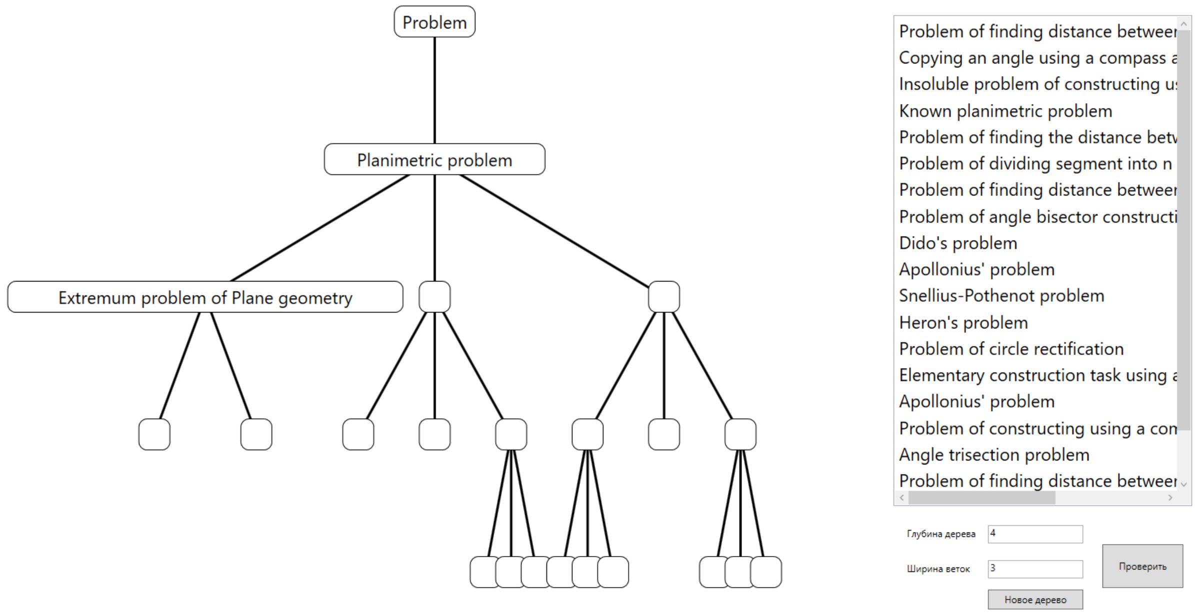

Figure 2. Example of an ontological task of the 1st type

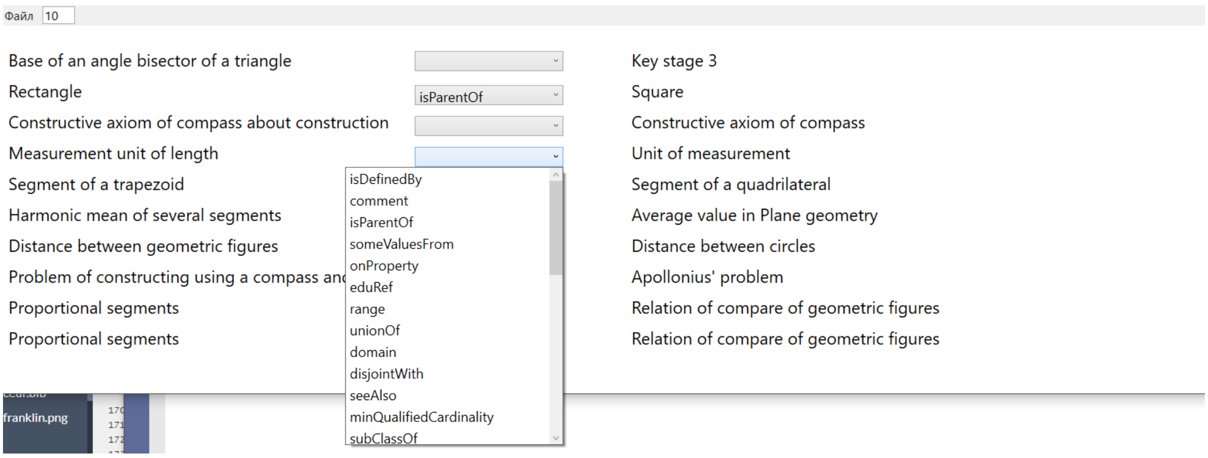

Figure 3. Example of an ontological task of the 2nd type

4. Ontology enrichment service

The development of the OntoMath^{Edu} ontology is currently in the active stage, and contains more than 900 concepts and more than 20 relationships. The task of introducing new relations between concepts is very important.

We have developed a method for introducing new relations between concepts, based on the use of a set of definitions of theorems. Our main idea is as follows. Each theorem contains some ontology concepts, and they can be divided into two types – concepts that are given in the problem condition and those that need to be obtained in the answer (for example, in the theorem "The area of a quadrilateral is half the product of the diagonals by the sine of the angle between them", the input concept is the "Diagonal of a quadrilateral", and the output concept is the "Area of a quadrilateral"). We assume that there must be a chain of concepts between the input and output concepts, and the length of such a chain must be less than 6 concepts (an empirically discovered limitation). If the chain between concepts is long or passes through concepts that are not related to the examined ones in any way, such a pair is passed for analysis to an expert group in order to introduce a direct ontological connection between the concepts (or between the concepts in the pair and some common concept present in the chain).

The concept chain search algorithm is essentially a breadth-first search algorithm on a graph. When composing the adjacency matrix, we consider all one-sided relationships as two-sided. In this way, we guarantee the processing of directional relationships between concepts.

We present some results of experiments on detecting chains between concepts in Table 2. It shows the theorem from which the concepts were distinguished, the concepts themselves and the chains between them, as well as possible corrections that will be made to the ontology. Table 2 shows the step-by-step process of ontology enrichment on the example of "The area of a quadrilateral is half the product of the diagonals by the sine of the angle between them" theorem. The list of concepts that are present in this theorem includes "Quadrilateral", "Diagonal of a quadrilateral", "Square of a quadrilateral", etc. Current state of connections between these concepts is shown in the "Concept chain" column of the table. We provide some actions that can make the ontology better. After linking the elements in the concept pairs ("Quadrilateral", "Diagonal of a quadrilateral") and ("Quadrilateral", "Area of a quadrilateral"), we get a short connection between the concepts ("Area of a quadrilateral", "Diagonal of a quadrilateral").

5. Conclusion

This article describes the structure of the OntoMath^{Edu} ecosystem, that is designed to support the process of personalized teaching of school mathematics. The components of the task generation ecosystem are presented in detail, and new task types based on ontology are proposed. In addition, an algorithm for enriching the ontology was given, which contributes to improving the horizontal connectivity of the ontology.

Table 2
Examples of improvements for the OntoMathEdu ontology

Source theorem	First concept	Second concept	Concept chain	Actions
The area of a quadrilateral is equal to half the product of the diagonals and the sine of the angle between them	Quadrilateral	Area of a quadrilateral	Quadrilateral -> Polygon -> Bounded part of the plane -> Part of the plane -> Un-limited part of the plane -> Angle... -> Area of the bounded part of the plane -> Area of a Polygon -> Area of a quadrilateral	The relationship “ontologically depends” is built
The area of a quadrilateral is equal to half the product of the diagonals and the sine of the angle between them	Quadrilateral	Diagonal of a quadrilateral	Quadrilateral -> The described quadrilateral -> The described polygon -> Polygon -> Bounded part of the plane -> Part of the plane -> Geometric shape on a plane -> Line -> Cut -> Polygon segment -> Diagonal of the polygon	The relationship “ontologically depends” is built
The area of a quadrilateral is equal to half the product of the diagonals and the sine of the angle between them	Area of a quadrilateral	Diagonal of a quadrilateral	Area of a quadrilateral -> Quadrilateral -> Diagonal of a quadrilateral	Connection is optimal

Future work is related to the implementation of the personalization component in all implemented test generators, as well as the overall development of these components. In addition, it is planned to develop a component for automatic analysis of a detailed response by drawing up a framework (scheme) of the solution, and filtering tasks by the types of these frameworks.

6. Acknowledgments

The research was funded by RFBR according to the project № 19-29-14084.

7. References

[1] A. Conde, M. Larranaga, A. Arruarte, J. A. Elorriaga, A Combined Approach for Eliciting Relationships for Educational Ontologies Using General-Purpose Knowledge Bases, IEEE Access 7 (2019) 48339–48355. doi:10.1109/ACCESS.2019.2910079.

[2] Y. Martinez-Ramirez, A. Ramirez-Noriega, M. Zayas-Esquer, S. Miranda-Mondaca, J. Armenta-Bojorquez, M. Quintero-Fonseca, M. Gonzialez-Videgaray, C. Cortes-Velazquez, Architecture of mathematical knowledge management system in education: Ontologybased and case-based, in: 2018 International Symposium on Computers in Education, Proceedings, 2018, pp. 1–5. doi:10.1109/SIIE.2018.8586771.

[3] J. Hai, An efficient semantic retrieval method for network education information resources, in: Proceedings - 2019 11th International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2019, 2019, pp. 522–526. doi:10.1109/ICMTMA.2019. 00121.

[4] A. Y. Satria, T. Tokunaga, Automatic generation of english reference question by utilising nonrestrictive relative clause, in: CSEDU 2017 – Proceedings of the 9th International Conference on Computer Supported Education, volume 1, INSTICC, SciTePress, 2017, pp. 379–386. doi:10.5220/0006320203790386.

[5] Y. Susanti, R. Iida, T. Tokunaga, Automatic generation of english vocabulary tests, in: CSEDU 2015 - 7th International Conference on Computer Supported Education, Proceedings, volume 1, INSTICC, SciTePress, 2015, pp. 77–87. doi:10.5220/0005437200770087.

[6] T. Soonklang, S. Pongpinigpinyo, W. Muangon, S. Kaewjamnong, Automatic question generation system for English exercise for secondary students, in: W. Chen (Ed.), Proceedings of the 25th International Conference on Computers in Education, ICCE 2017 – Main Conference Proceedings, APSCE, 2017, pp. 890–895.

[7] K. Sakaguchi, Y. Arase, M. Komachi, Discriminative approach to fill-in-The-blank quiz generation for language learners, in: ACL 2013 - 51st Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, volume 2, Association for Computational Linguistics (ACL), 2013, pp. 238–242.

[8] J. Araki, D. Rajagopal, S. Sankaranarayanan, S. Holm, Y. Yamakawa, T. Mitamura, Generating questions and multiple-choice answers using semantic analysis of texts, in: COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers, Association for Computational Linguistics, ACL Anthology, 2016, pp. 1125–1136.

[9] W. Wang, T. Hao, W. Liu, Automatic question generation for learning evaluation in medicine, in: H. Leung, F. Li, R. Lau, Q. Li (Eds.), Lecture Notes in Computer Science, volume 4823, Springer, 2008, pp. 242–251. doi:10.1007/978-3-540-78139-4_22.

[10] M. Agarwal, P. Mannem, Automatic Gap-fill Question Generation from Text Books, in: Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications, June, 2011, pp. 56–64.

[11] G. Kumar, R. E. Banchs, L. F. D'Haro, Automatic fill-the-blank question generator for student self-assessment, in: Proceedings - Frontiers in Education Conference, FIE, volume 2015, 2015, pp. 1–3. doi:10.1109/FIE.2015.7344291.

[12] N. Afzal, R. Mitkov, Automatic generation of multiple choice questions using dependency-based semantic relations, Soft Computing 18 (2014) 1269–1281. doi:10.1007/s00500-013-1141-4.

[13] A. P. Tomás, J. P. Leal, Automatic generation and delivery of multiple-choice math quizzes, in: Lecture Notes in Computer Science, volume 8124, Springer, 2013, pp. 848–863. doi:10.1007/ 978-3-642-40627-0_62.
[14] K. Wang, T. Li, J. Han, Y. Lei, Algorithms for Automatic Generation of Logical Questions on Mobile Devices, in: IERI Procedia, volume 2, 2012, pp. 258–263. doi:10.1016/j.ieri.2012.06.085.
[15] A. Moiasova, J. Pocsova, Automatic test generator for analytic geometry, in: Proceedings of the 2019 20th International Carpathian Control Conference, ICCC 2019, IEEE, 2019, pp. 1–6. doi:10.1109/CarpathianCC.2019.8765961.
[16] R. Singhal, M. Henz, K. McGee, Automated generation of geometry questions for high school mathematics, in: CSEDU 2014 – Proceedings of the 6th International Conference on Computer Supported Education, volume 2, SCITEPRESS – Science and and Technology Publications, 2014, pp. 14–25. doi:10.5220/0004795300140025.
[17] R. Singhal, M. Henz, S. Goyal, A framework for automated generation of questions based on First-Order logic, in: Lecture Notes in Computer Science, volume 9112, Springer, 2015, pp. 776–780. doi:10.1007/978-3-319-19773-9_114.
[18] S. Williams, Generating mathematical word problems, AAAI Fall Symposium - Technical Report FS-11-04 (2011) 61–64.
[19] N. A. Khodeir, H. Elazhary, N. Wanas, Generating story problems via controlled parameters in a web-based intelligent tutoring system, The International Journal of Information and Learning Technology 35 (2018) 199–216. doi:10.1108/IJILT-09-2017-0085.
[20] A. Kirillovich, O. Nevzorova, M. Falileeva, E. Lipachev, L. Shakirova, OntoMath^{Edu}: Towards an Educational Mathematical Ontology, in: E. Brady, et al. (Eds.) Workshop Papers at the 12th Conference on Intelligent Computer Mathematics (CICM-WS 2019), CEUR Workshop Proceedings (2020) 2634. URL: http://ceur-ws.org/Vol-2634/WiP1.pdf.
[21] A. Kirillovich, O. Nevzorova, M. Falileeva, E. Lipachev, L. Shakirova, OntoMath^{Edu}: A linguistically grounded educational mathematical ontology, in: Lecture Notes in Computer Science, volume 12236, Springer, 2020, pp. 157–172. doi:10.1007/978-3-030-53518-6_10.
[22] A. Kirillovich, O. Nevzorova, K. Nikolaev, K. Galiaskarova, Towards a Parallel Informal-Formal Corpus of Educational Mathematical Texts in Russian, in: Advances in Intelligent Systems and Computing, volume 1127, Springer, 2020, pp. 325–334. doi:10.1007/978-3-030-39216-1_29.
[23] P. Cimiano, C. Chiarcos, J.P. McCrae, J. Gracia, Linguistic Linked Open Data Cloud, in: P. Cimiano, et al. (Eds.) Linguistic Linked Data: Representation, Generation and Applications, Springer, 2020, pp. 29–41. doi:10.1007/978-3-030-30225-2_3.
[24] J.P. McCrae, C. Fellbaum, P. Cimiano, Publishing and Linking WordNet using Lemon and RDF, in: Proceedings of the 3rd Workshop on Linked Data in Linguistics (LDL-2014), 2014, pp. 13–16.
[25] M. Ehrmann, et al. Representing Multilingual Data as Linked Data: the Case of BabelNet 2.0, in: Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014), ELRA, 2014, pp. 401–408.
[26] A. Kirillovich, O. Nevzorova, E. Gimadiev, N. Loukachevitch, RuThes Cloud: Towards a Multilevel Linguistic Linked Open Data Resource for Russian, in: P. Różewski, C. Lange (Eds.) Communications in Computer and Information Science, volume 786, Springer, 2017, pp. 38–52. doi:10.1007/978-3-319-69548-8_4.
[27] A. Galieva, A. Kirillovich, B. Khakimov, N. Loukachevitch, O. Nevzorova, D. Suleymanov, Toward Domain-Specific Russian-Tatar Thesaurus Construction, in: Proceedings of the International Conference IMS-2017, ACM, 2017, pp. 120–124. doi:10.1145/3143699.3143716.
[28] K. Nikolaev, A. Kirillovich, O. Nevzorova, A Corpus-Based Approach to Elementary Geometry Knowledge Test Generation, in: L. Gómez Chova, et al. (Eds). Proceedings of the 14th International Technology, Education and Development Conference (INTED 2020), IATED, 2020, pp. 6342–6348. doi:10.21125/inted.2020.1710.
[29] A. Elizarov, A. Kirillovich, E. Lipachev, O. Nevzorova, Semantic Formula Search in Digital Mathematical Libraries, in: Proceedings of the 2nd Russia and Pacific Conference on Computer Technology and Applications (RPC 2017), 2017, pp. 39–43. doi:10.1109/RPC.2017.8168063.
[30] A.M. Elizarov, A.V. Kirillovich, E.K. Lipachev, A.B. Zhizhchenko, N.G. Zhil'tsov, Mathematical Knowledge Ontologies and Recommender Systems for Collections of Documents in Physics and Mathematics, Doklady Mathematics 93 (2016) 392–395. doi:10.1134/S1064562416020174.