Experience in Integrating Domain–Specific Knowledge Bases based on Semantic Web Standards

Victor Telnov[1][0000-0003-0176-5016] and Konstantin Odintsov[2]

[1] National Research Nuclear University MEPhI, Obninsk, Russia
[2] Lomonosov Moscow State University, Moscow, Russia
telnov@bk.ru

Abstract. The paper discusses the optimal machine learning algorithms for virtual integration of knowledge bases with external data sources in Russian and English. Data in the external sources can be presented in RDF, RDFS, OWL, XML, HTML, JSON, CSV formats, in the form of relational, graph databases, or not structured at all. The algorithms under consideration are designed to provide a methodological and technological basis for development of the problem–oriented knowledge bases as artificial intelligence systems, as well as prerequisites for the development of semantic technologies for acquiring new knowledge on the internet without direct human participation. Testing of the algorithms is carried out by the method of cross–validation on specialized text corpora. The novelty of the presented study is due to the application of the Pareto's optimality principle for multi–criteria evaluation and ranking of the studied algorithms in the absence of a priori information about the comparative significance of the criteria. The architecture of the semantic web portal, usage examples are given. The proposed software solutions are based on cloud computing using DBaaS and PaaS service models to ensure scalability of data warehouses and network services.

Keywords: Semantic Web, Knowledge Base, Machine Learning, Text Classification, Cloud Computing.

1 Introduction

Classification and integration of weakly structured knowledge, founded on ontologies, is a non-trivial task of informatics. It is usually not difficult for a human to understand whether two or more entities are related based on cognitive associations, while it is not always easy for a computer to do this. Endowing machines with common sense and domain understanding has been and continues to be an important challenge in the field of artificial intelligence. As of 2023, educational web portals of universities and open corporate knowledge management systems are isolated from each other, do not use the capabilities of the semantic web and machine learning methods. In distance learning, the popular educational platforms Coursera, edX, Moodle, etc. are prevailed, which are based on taxonomies, thesauri and are not very compatible with each other in terms of data formats. The project [1] under discussion is a step towards virtual integration of educational resources, initially in terms of university knowledge bases in computer science and programming. The long–term goal of the entire project is a public semantic educational web portal of a new generation, which is based on machine learning methods [2], Semantic Web standards and technologies [3]. Potential users of the project results are students, teachers, managers, experts, researchers and specialists in the field of computer science and programming.

In order to fulfill virtual integration of external data with already existing knowledge bases, it is necessary to somehow intelligently embed these external data into existing ontologies. In particular, new objects should not violate such fundamental properties of the ontology as satisfiability [4] and consistency [5]. The project under discussion [1] combines the advantages of the joint use of Semantic Web technologies, machine learning methods and Pareto's optimization [6]. By implementing virtual integration of knowledge, we programmatically select the best method of content classification each time, performing multi-criteria optimization of algorithms, provided that there is no a priori information about the comparative importance of criteria. One of the significant observations is that different algorithms are effective on different knowledge graphs (on different text corpora). We emphasize that optimization is performed according to any number of quantitative and qualitative criteria, without any preliminary ranking of these criteria. This is the property of Pareto's optima, which is often underestimated. Each of the tested methods of content classification has individual parameters for fine-tuning, which affect the quality of the algorithm. These parameters can also be optimization objects. The sound recommendations and hints are made programmatically about how a person can add new data to existing knowledge bases. The fully automating the process of integrating knowledge bases at this stage of their development seems premature.

2 Problem statement

The project [1] considered in the paper is aimed at creating, testing and introducing systems and methods for acquiring, presenting and virtual integration of knowledge in the domain of computer science and programming into the educational practice of universities. Virtual integration does not imply physical data consolidation, but means the ability to navigate external data sources using an RDF browser [9, 10]. Possible external data formats for virtual integration with knowledge bases are shown below in Fig. 1. There are a sufficient number of publicly available RDF adapters that perform mapping of relational and graph databases, as well as network data formats XML, HTML, JSON to RDF format. For example: OpenLink Virtuoso RDF Views, D2RQ-Map, SquirrelRDF, etc.

The tasks of the project at the 2023 stage are determined by the following functional requirements for the software: 1) interactive context-dependent search and selection of educational content on the Internet; 2) interactive selection of knowledge graphs that are used for classification and semantic annotation of the content; 3) interactive selection of optimal machine learning methods; 4) management of options for classification processes and semantic annotation of the content; 5) saving the results of classification and semantic annotation of content on the client computer; 6) management of remote access to knowledge bases by teachers and knowledge engineers, tools for joint editing of the ontologies.

The classification (categorization) of textual content there is the process of assigning text to one or another class of a certain knowledge graph. Semantic annotation there is the process of tagging documents with relevant concepts (classes) and individuals (objects) from the knowledge graph. The actual classifier used for classification can be viewed online on a client computer with any level of detail. To do this, one needs to enter into the knowledge graph using the reference [1] and open the class hierarchy. The usage of the Pareto's optimality principle provides multi-criteria optimization and ranking of algorithms for virtual integration of knowledge and machine learning, provided that there is no a priori information about the comparative importance of criteria.

The technological backlog of the project is represented by a prototype [1]. The software solutions being created are based on cloud computing according to the PaaS and DBaaS service models, which ensures the scalability of semantic repositories, network services and protects the server code from stalling during surges in the flow of network requests. To solve the problems of the content classification and its integration into the knowledge bases, the following machine learning methods are studied and applied:

1. Naive Bayes classifiers.
2. Softmax classifiers (maximum entropy model).
3. Classifiers based on the Support Vector Machine with SGD.
4. Classifiers using the Nearest Neighbors method.
5. Classifiers based on Terminological Decision Trees.

At the initial stage, the following seven knowledge graphs on the computer science and programming act as training and test sets:

1. Knowledge graph “Semantic Web”.
2. Knowledge graph “Programming technologies”
3. Knowledge graph “Object-Oriented Programming”.
4. Knowledge graph “Front-end web programming”.
5. Knowledge graph “Programming paradigms and patterns”.
6. Knowledge graph “Cloud services and technologies”.
7. Joint Graph of Knowledge.

The listed above knowledge graphs contain syllabuses and relevant learning resources which are created and taught by the authors of the paper. When solving the problems formulated above, both publicly available software Apache Jena, Stanford NLP, Scikit-learn, Weka and original software are used.

3 Methods and means of solving the problem

The main subject of research in the project [1] is algorithms for virtual integration of external data with the knowledge bases on the domain of computer science and programming. Research methods include software design, implementation, tuning and testing of algorithms. From a practical point of view, the project is implemented on the Jelastic cloud platform in Java and Python runtime environments. The development is based on Semantic Web standards and technologies: RDFS, OWL, SPARQL, as well as description logics ALC and SROIQ(D) [11]. Full text educational objects and media content are placed on the Internet in arbitrary remote data storage and at video hostings. The choice of a specific remote storage is not important, any repositories equipped with content display tools (Google Drive, Yandex.Disk, YouTube, etc.) can be used.

During the implementation of the project, the effectiveness of relatively simple, intuitive machine learning methods for solving the problem of automated filling from the Internet and virtual integrating knowledge bases without direct human participation was studied on seven corpora of specialized texts in computer science and programming. Each of the seven involved knowledge graphs contained about one thousand objects and about one hundred classes. Such moderate sizes of knowledge graphs are typical for the specialized domains. For instance, when a knowledge graph is created for learning purposes and includes a specific academic discipline.

Cross-validation was used to test classification algorithms in Russian and English. Each training set was randomly divided three times into three samples of approximately the same size. Each of the three samples was in turn declared a control sample, the remaining two samples were combined into a training sample. The classification algorithm was adjusted according to the training sample, and then it classified the objects of the control sample. The described procedure was repeated three times for each classification algorithm and for each knowledge graph.

To assess the quality of classification algorithms, macro-averages of well-known machine learning metrics *Accuracy*, *Precision*, *Recall*, *F1-score* were used. Specific metric values are averaged over all classes, regardless of the number of objects in these classes. The *Accuracy* metric shows the proportion of correctly classified objects. The *Precision* metric characterizes the ability of the algorithm to distinguish classes from each other. The *Recall* metric measures the algorithm's ability to detect a particular class at all. The *F1-score* metric is a derivative of the two previous metrics and is calculated as their harmonic mean. It is informative in cases where the values of other metrics differ significantly from each other.

Having the results of testing five classification algorithms on seven knowledge graphs in Russian and English, see Table 1, it is possible to calculate a set of Pareto-optimal algorithms that are the best in the aggregate of all computational experiments performed. The optimization problem is formulated as follows. It is required to choose the best classification method, taking into account all the calculated metrics, without making any a priori assumptions about the relative importance of these metrics. For this, in the class of transitive anti-reflexive binary relations, the Pareto's relation in the Euclidean space is considered. This relation is also called the dominance relationship. The essence of this relation is as follows. It is said that some element x from some set dominates another element y from the same set if x is not worse than y in all aspects (criteria) and at least one aspect x is superior to y. The set of all non-dominated elements is called the Pareto's set. The binary Pareto's relation provides a universal mathematical model of multi-criteria context-independent choice in the Euclidean space.

Table 1. Calculated metric values for five text classification methods tested on seven knowledge graphs.

Knowledge graphs as training and test sets	Text data classification method																			
	Naive Bayes Classifier				Maxent Classifier (Softmax)				SVM Classifier with SGD				Nearest Neighbors Classifier				Terminological Decision Trees			
	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F
Semantic Web	97	97	96	97	97	54	96	69	74	79	75	77	87	86	86	86	83	87	80	83
Programming technologies	96	98	31	48	97	46	41	43	74	79	74	76	79	86	72	78	81	88	68	77
Object Oriented Programming	96	97	39	56	97	53	52	53	75	81	75	78	87	99	68	81	83	77	85	81
Front–end web programming	96	98	39	56	96	54	46	49	74	80	74	77	77	67	63	65	83	78	86	82
Programming paradigms and patterns	97	97	44	60	98	67	71	69	76	82	76	79	85	90	79	84	74	83	63	71
Cloud services and technologies	95	97	61	75	96	60	67	64	75	81	75	78	83	76	76	76	77	71	87	78
Joint Graph of Knowledge	98	98	15	25	98	44	40	42	72	77	72	75	79	80	63	71	74	75	68	71

Note: A – *Accuracy* (%); P – *Precision* (%); R – *Recall* (%); F – *F1-score* (%).

Based on the Pareto's relation, a choice function is constructed that generates a set of elements that are the best in terms of all the calculated metrics, without any a priori assumptions about the comparative importance of these metrics. The corresponding mathematical formulas are presented in [12, 13]. The results of the Pareto's set calculations are shown below in Table 2.

The results obtained in the course of computational experiments allow us to draw the following conclusions. Among the five test methods for classifying texts on natural languages, the leader is the “Naive Bayes Classifier” method with dominance indexes of 7, 12, and 14. It is always included in the set of Pareto-optimal algorithms. The “Nearest Neighbors Classifier” method is not much inferior to it. The methods “Maxent Classifier (Softmax)” and “SVM Classifier with SGD” look like outsiders on the studied text corpora.

Table 2. Computed dominance indexes for five text classification methods tested on seven knowledge graphs.

Text data classification methods	Dominance indexes calculated when accounting *Precision* and *Recall* metrics	Dominance indexes calculated when accounting *Accuracy*, *Precision* and *Recall* metrics	Dominance indexes calculated when accounting *Accuracy*, *Precision*, *Recall* and *F1–score* metrics
Naive Bayes Classifier	**7**	**12**	**14**
Maxent Classifier (Softmax)	13	13	20
SVM Classifier with SGD	8	15	19
Nearest Neighbors Classifier	**7**	14	15
Terminological Decision Trees	8	15	16

4 Discussion and applied interpretation of the results

To verify the results presented in the previous section of the paper, we were compared with data obtained by independent researchers on other text corpora using advanced methods of deep machine learning. In a recent review [14], Table 1 on page 27 shows the results of testing a number of machine learning algorithms for solving text classification problems. In particular, the "Naive Bayes Classifier" in our study showed an average *Accuracy* of approximately 96%, while the same classifier on the SST-2 text corpus gave an *Accuracy* of 81.80%. As can be seen from the review [14], deep machine learning algorithms on the SST-2 text corpus give an average *Accuracy* of about 91%, which is by no means better than the *Accuracy* shown by the "Naive Bayes Classifier" on the data sets in our study. The given observation suggests that at present the "Naive Bayes Classifier", as well as the "Nearest Neighbors Classifier", provide sufficient competence of semantic knowledge bases as systems of artificial intelligence.

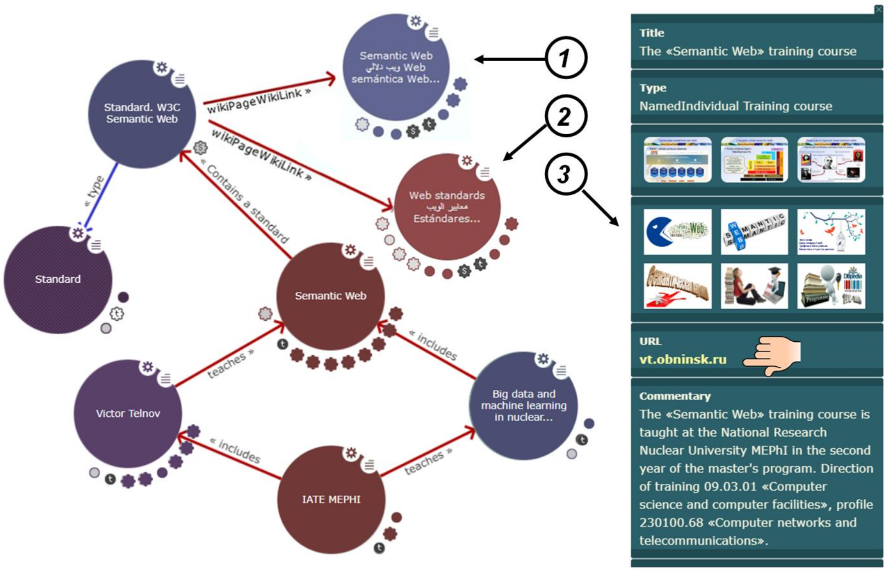
The educational product under construction is a semantic web portal on computer science and programming, designed in accordance with the MVC architectural pattern. Algorithms for processing and presenting data are separated from each other and from the actual data (learning objects). Learning objects are combined through ontologies into knowledge graphs, which are placed in semantic repositories on a cloud platform.

The general structure of the software product is as follows:

1. Semantic repository as a storage of knowledge graphs, equipped with the Apache Jena engine.
2. Search widgets for quick immersion in knowledge graphs. Their interface is similar to how the search query string works in popular search engines.
3. Intelligent RDF browser for interactive visual navigation through knowledge graphs. Intuitive visual way of navigation resembles a computer game adventure type “walker”, does not require special skills and is accessible to an inexperienced user. Students master the RDF browser within a few minutes.
4. Component for interactive context-sensitive search and selection of content on the Internet. Indirectly uses the capabilities of regular search engines.
5. A component for semantic annotation of network content in the interests of filling and updating knowledge graphs.
6. New software component “Semantic Classification” for automated classification of network content in the interests of filling, updating knowledge graphs and integrating knowledge bases.
7. Plugin component “Ontology Editor WebProtege” for managing remote access to knowledge graphs (ontologies) by teachers, knowledge engineers and for joint editing of ontologies.
8. Public access points to international knowledge bases.

The initial filling and subsequent updating of knowledge bases, their integration with third-party educational resources, is the prerogative of university professors and knowledge engineers. The educational portal can be replicated without restrictions, adapting the content of knowledge bases to new types of syllabuses or levels of education. The software is available worldwide where the Internet is available. It can be used in traditional and distance learning as the main and additional source of lecture material, textbooks, practical tasks, knowledge control tools, etc.

If the user of the knowledge base has the ability to seamlessly move from one knowledge graph to another graph in the RDF browser, it's called seamless graph integration. The figures show fragments of knowledge graphs illustrating seamless integration with external data, see Fig. 1 and Fig. 2 below.



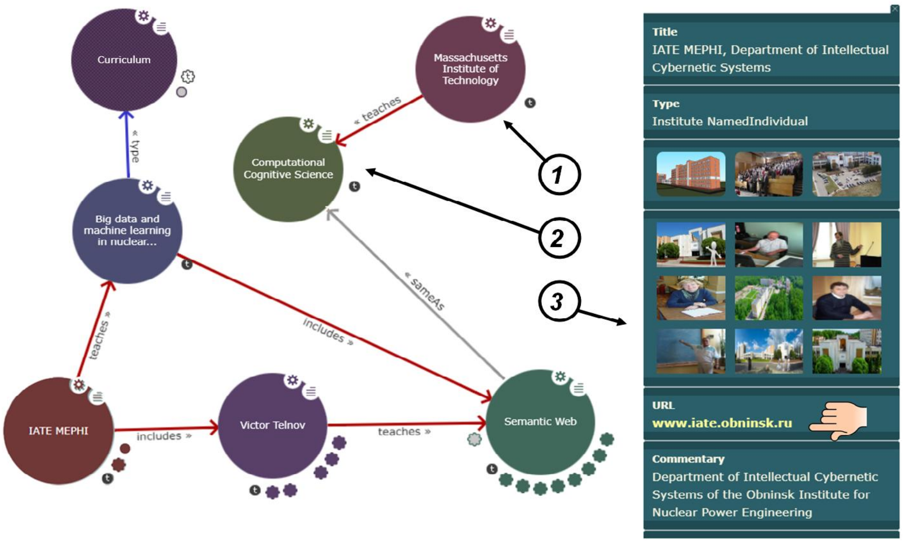
Title
The «Semantic Web» training course

Type
NamedIndividual Training course

URL
vt.obninsk.ru

Commentary
The «Semantic Web» training course is taught at the National Research Nuclear University MEPhI in the second year of the master's program. Direction of training 09.03.01 «Computer science and computer facilities», profile 230100.68 «Computer networks and telecommunications».

Fig. 1. Navigation in the RDF browser: an example of seamless integration of the "Semantic Web" knowledge graph and external semantic resources: 1 – the "Semantic Web" object from DBpedia; 2 – the "Web standards" object from DBpedia; 3 – metadata for the "Semantic Web" object.



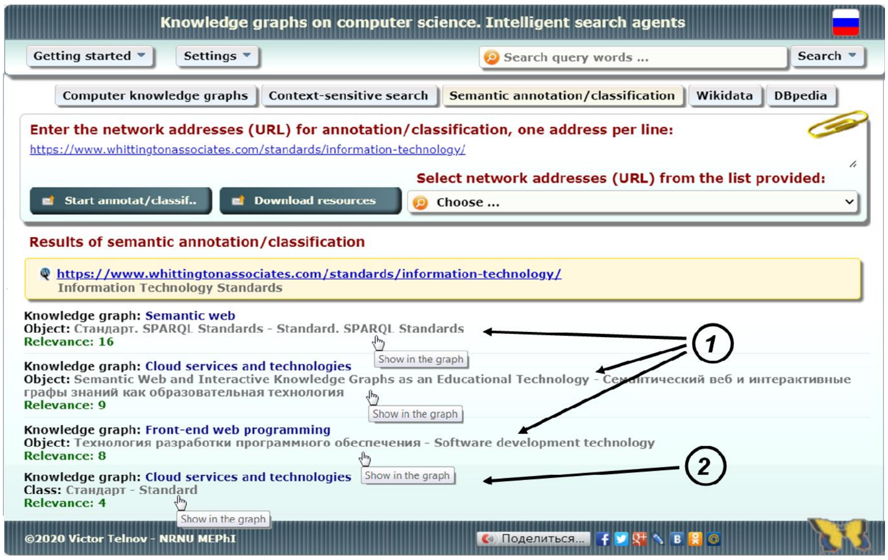
Title
IATE MEPHI, Department of Intellectual Cybernetic Systems

Type
Institute NamedIndividual

URL
www.iate.obninsk.ru

Commentary
Department of Intellectual Cybernetic Systems of the Obninsk Institute for Nuclear Power Engineering

Fig. 2. Navigation in the RDF browser: an example of seamless integration of the "Semantic Web" knowledge graph and external non–semantic resources: 1 – a third-party university "Massachusetts Institute of Technology"; 2 – a third-party syllabus "Computational Cognitive Science" related to the "Semantic Web" syllabus; 3 – metadata for the "IATE MEPHI" object.



Enter the network addresses (URL) for annotation/classification, one address per line:
https://www.whittingtonassociates.com/standards/information-technology/

Select network addresses (URL) from the list provided:

Results of semantic annotation/classification

https://www.whittingtonassociates.com/standards/information-technology/
Information Technology Standards

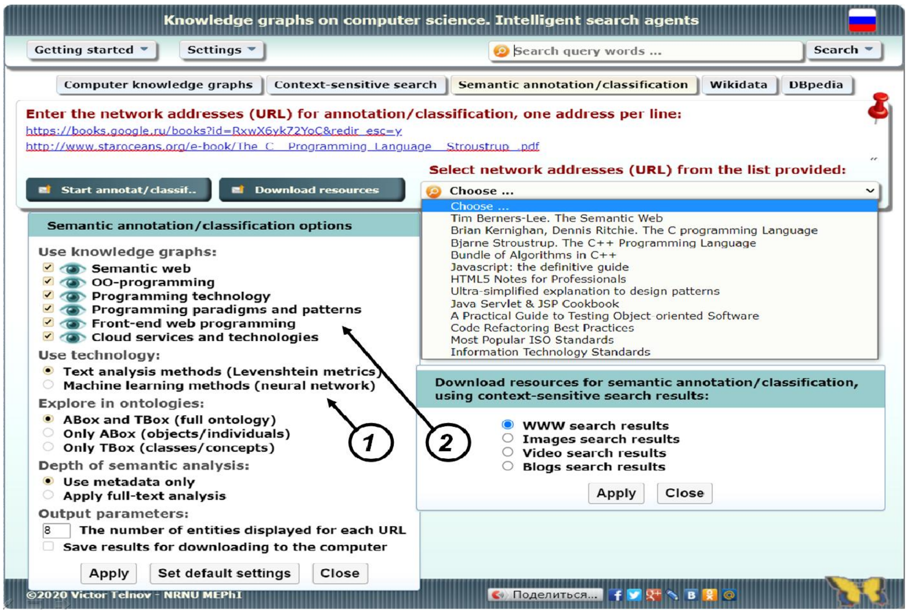
Knowledge graph: Semantic web
Object: Стандарт. SPARQL Standards - Standard. SPARQL Standards
Relevance: 16

Knowledge graph: Cloud services and technologies
Object: Semantic Web and Interactive Knowledge Graphs as an Educational Technology - Семантический веб и интерактивные графы знаний как образовательная технология
Relevance: 9

Knowledge graph: Front-end web programming
Object: Технология разработки программного обеспечения - Software development technology
Relevance: 8

Knowledge graph: Cloud services and technologies
Class: Стандарт - Standard
Relevance: 4

Fig. 3. Knowledge graphs: an example of issuing the results of semantic annotation and classification of network content: 1 – computed semantic annotations; 2 – classification results.



Enter the network addresses (URL) for annotation/classification, one address per line:
https://books.google.ru/books?id=RxwX6yk72YoC&redir_esc=y
http://www.staroceans.org/e-book/The_C__Programming_Language__Stroustrup_.pdf

Select network addresses (URL) from the list provided:

Choose ...
Tim Berners-Lee. The Semantic Web
Brian Kernighan, Dennis Ritchie. The C programming Language
Bjarne Stroustrup. The C++ Programming Language
Bundle of Algorithms in C++
Javascript: the definitive guide
HTML5 Notes for Professionals
Ultra-simplified explanation to design patterns
Java Servlet & JSP Cookbook
A Practical Guide to Testing Object-oriented Software
Code Refactoring Best Practices
Most Popular ISO Standards
Information Technology Standards

Semantic annotation/classification options

Use knowledge graphs:
☑ Semantic web
☑ OO-programming
☑ Programming technology
☑ Programming paradigms and patterns
☑ Front-end web programming
☑ Cloud services and technologies

Use technology:
● Text analysis methods (Levenshtein metrics)
○ Machine learning methods (neural network)

Explore in ontologies:
● ABox and TBox (full ontology)
○ Only ABox (objects/individuals)
○ Only TBox (classes/concepts)

Depth of semantic analysis:
● Use metadata only
○ Apply full-text analysis

Output parameters:
8 The number of entities displayed for each URL
□ Save results for downloading to the computer

Download resources for semantic annotation/classification, using context-sensitive search results:
● WWW search results
○ Images search results
○ Video search results
○ Blogs search results

Fig. 4. Knowledge graphs: panels for setting options and loading resources for semantic annotation and classification of network content: 1 – choice of classification and semantic annotation technology; 2 – selection of training and control sets for setting up classification and semantic annotation algorithms.

5 Conclusion

The created prototype of the knowledge base in the domain of computer science and programming is currently used at National Research Nuclear University MEPhI as part of master's syllabuses. Provided the introduction of such integrated knowledge bases into the educational practice of universities, one can expect the expansion of the professional horizons of students and the increase in the level of competence of teachers due to the emergence of a new unified channel for access to the educational objects allocated at third-party universities. The proposed educational product provides university teachers with a toolkit of the authoring that helps create new and update existing courses in the field of computer science and programming. The RDF smart browser provides for students, educators and stakeholders with seamless, interactive navigation through the knowledge bases of many universities around the world. This is expected to increase the attractiveness of university education in the eyes of entrants, students and third parties, and also create prerequisites for expanding the scope of knowledge bases as artificial intelligence systems.

Funding

The study was supported by the Russian Science Foundation grant No. 22-21-00182[1].

References

1. Knowledge graphs on computer science. Intelligent search agents, http://vt.obninsk.ru/s/, last accessed 2023/05/15.
2. Geron, A.: Hands–On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd edn. O'Reilly Media, Inc. Sebastopol (2019).
3. W3C Semantic Web, http://www.w3.org/standards/semanticweb, last accessed 2023/05/15.
4. Encyclopedia of Mathematics. Satisfiability, https://encyclopediaofmath.org/wiki/Satisfiability, last accessed 2023/05/15.
5. Encyclopedia of Mathematics. Consistency, https://encyclopediaofmath.org/wiki/Consistency, last accessed 2023/05/15.
6. Vilfredo Pareto, https://www.newworldencyclopedia.org/entry/Vilfredo_Pareto, last accessed 2023/05/15.
7. Fettach Y., Ghogho M., Benatallah B.: Knowledge Graphs in Education and Employability: A Survey on Applications and Techniques. IEEE Access 10, 80174–80183 (2022), DOI: 10.1109/ACCESS.2022.3194063.
8. Gutierrez C., Sequeda J.: Knowledge graphs. Communications of the ACM 54, 96–104 (2021), DOI: 10.1145/3418294.
9. Telnov V. P., Korovin Yu. A.: Programming of Knowledge Graphs, Reasoning on Graphs. Software Engineering 10(2), 59—68 (2019), DOI: 10.17587/prin.10.59-68.
10. Telnov V., Korovin Y.: Semantic Web and Interactive Knowledge Graphs as Educational Technology. In: Cloud Computing Security, Dinesh G. Harkut (eds.), IntechOpen London (2020), DOI: 10.5772/intechopen.83221.

[1] https://rscf.ru/project/22-21-00182/

11. Telnov V., Korovin Y.: Semantic web and knowledge graphs as an educational technology of personnel training for nuclear power engineering. Nuclear Energy and Technology 5(3), 273–280 (2019), DOI: 10.3897/nucet.5.39226.
12. Telnov V.P., Korovin Yu.A.: Application of Machine Learning Methods for Filling and Updating Nuclear Knowledge Databases. Izvestiya vuzov. Yadernaya Energetika 4, 122–133 (2022), DOI: 10.26583/npe.2022.4.11.
13. Telnov V.P., Korovin Y.A., Odintsov K.V.: On the Issue of Optimum Machine Learning Methods for Filling and Updating Nuclear Knowledge Graphs. Lobachevskii J. Math., 44(1), 227–236 (2023), DOI:10.1134/S1995080223010419.
14. Minaee S., Kalchbrenner N., Cambria E., Nikzad N., Chenaghlu M., Gao J.: Deep Learning Based Text Classification: A Comprehensive Review. ACM Computing Surveys 54(3), 1–40 (2022), DOI: 10.1145/3439726.