Computing Education & Technological Trends: A Systematic Review Study

Oleksandr Nesterenko

International European University, Glushkov av., 42, Kyiv 03127, Ukraine

Abstract

According to international recommendations, the quality of computing education is influenced by the consideration of modern industrial trends and IT tendency, as well as the methodology of computer curricula. The transition to a new level of industry, characterized by robotics and cyber-socialization, brings a wave of new, never-before-seen challenges to the field of IT and programming. Under such conditions, it is impossible to achieve the desired synergistic effect of learning without the formation and actualization of the appropriate information environment in higher education institutions. One of the sources of the content of such an environment can be the publications of the world scientific community registered in international scient metric databases. On these sources, it becomes possible not only to provide effective digital educational content, but also to choose and use the latest technological means of learning supporting. A targeted review of primary sources in such a database allowed for the identification, selection, synthesis and evaluation of evidence related to the following research questions: (i) what industrial technological trends are analyzed in research, (ii) what information technologies are considered in research, (iii) what strategies are reflected in publications to overcome the above challenges of computing education. According to results of this study conducted on example of software engineering, it is proposed to improve the methodological support of computing education based on ontological modeling of the requirements integration a modern development trends in industry and IT sector, international recommendations and the capabilities of technologies learning tools. This approach is a good basis for the processes of intellectualization of educational environment information systems and the construction of appropriate technological educational tools on their basis.

Keywords

Digital transformation, quality of education, information technologies, educational technologies, ontologies, educational environment, software engineering

1. Introduction

In our time of digital transformations more and more attention has been paid to improving the quality of computing education in the processes of forming the intellectual potential of humankind. These approaches align with the Global Computing Education paradigms outlined in the latest release of the CC2020 Computing Curricula report series from the Association for Computing Machinery (ACM) and the IEEE Computer Society (IEEE-CS). Among the undergraduate programs covered by this document, software engineering (SE) is a key discipline that is becoming the backbone of technological progress. Software development is influenced by many factors, such as technology, consumer preferences, and market factors. That is why SE is significantly different from other educational disciplines in its versatility due to the integration of technical and technological dominants with engineering approaches, economic categories and provisions of management. Due to

the constant increase in the number of technologies in the industry, the coming years will also see an increase in trends in software development. Therefore, in order to develop modern and innovative technologies, it is necessary to understand the latest trends and take into account the peculiarities of SE. In turn, this causes certain difficulties in the organization of the process of software engineering education (SEE) and requires constant updating of approaches, methodological support, methods and learning tools. That is why it is expedient to consider the topic of research in the example of software engineering.

Analysis of the problem research in the world scientific literature shows that the development of software engineering is significantly influenced by modern industrial trends, as well as trends in information technologies and programming [1-3]. The transition to a new level of industry - "Industry 4.0", characterized by robotics and cyber-socialization, brings a wave of new, never-before-seen challenges in the IT and programming spheres. Due to the complexity and rapidity of the development of software technologies, the basis for acquiring knowledge about SE and applying innovations in this area should be a shift in emphasis to the paradigm of a deep understanding of modern technological processes and the acquisition of professional skills based on practice. Obviously, in such conditions, it is impossible to achieve the desired synergistic effect without the formation of an appropriate informational and educational environment in higher education institutions, which is provided for by the relevant standards of higher education and requirements for internal quality assurance of education. When designing and forming such an environment, the priority should be the implementation of means of intellectualization, the principles of open education and the European research space [4]. At the same time, it is expedient to introduce a transdisciplinary approach into this process as the most generalized and universal, which overcomes the limits and shortcomings of disciplinary fragmentation, provides a more complete picture of the world and relationships [5]. However, within the framework of the general problem of setting up an environment for educational activities in the field of SE based on such systems, the task of forming information resources that should reflect the above-mentioned technological trends is relevant. Based on them, it becomes possible not only to provide effective digital educational content, but also to choose and use the latest technological means of learning supporting.

One of the sources of such content can be the publications of the world scientific community registered in international scientometric databases, such as, for example, Scopus and Web of Science. To provide a comprehensive view of what was said on a certain topic and by whom, the systematic review method is effective. Such a high-level review of primary research on a targeted question identifies, selects, synthesizes and evaluates all high-quality research evidence relevant to the question at hand. As IT transforms and evolves so rapidly, such studies should be continually replicated to examine how these changes affect the gap between curriculums and current industry expectations.

The research goal is a systematic study of sources for finding informational, technological and methodical ways to solve the current problems of ensuring the improvement of the quality of computing education on the example of software engineering in the conditions of digital transformation.

2. Analysis of recent research and publications

For more than half a century of development of software engineering, many publications ranging from fundamental editions to current publications in scientific journals and conference proceedings provide a broad panorama of topics covering all stages and technologies of developing software systems. Software engineering has changed significantly over the decades, while being a new engineering discipline it is still evolving. This is confirmed by a systematic analysis [1] of a number of publications over a certain period, each of which offered its own sets of principles for SE development.

In [2], based on a systematic review, the main phenomena influencing evolution of SE practices and which helped the author to assess how this field got to where it was and where it is now are determined. This article also identifies some of the major sources of change that will affect the practice of software development over the next several decades and identifies some strategies for assessing and adapting to these sources of change.

In [3], a classification of many articles was carried out based on studies of trends in software development, processes and practices, approaches to learning, and the evolution of IP over time. It is shown that many trends other than Agile Software Development are relatively underrepresented in academic organization. The study's findings point to existing gaps between the software industry and education, providing useful insights for researchers, educators, and practitioners. An example of one such publication is [6], in which, based on a comprehensive survey of IT professionals and companies, it is determined that balancing technical/non-technical skills and adjusting curricula to better prepare students for industry is a constant challenge for higher education institutions.

At the same time, the proliferation and widespread use of agile methods have created enormous challenges for innovation in SEE. These pedagogical innovations should be aimed at strengthening the skills of students to achieve optimal results in the industry. However, recent research still points to differences in what the industry requires. The main goal of the article [7] is to collect and analyze scientific data on the skills necessary for the formation of flexible high-performance teams. In this work, a systematic literature review was carried out to get a visualization of the scientific contributions available in this topic.

Taking into account the fact that in software development, knowledge is constantly updated, a systematic review study [8] shows the need to focus attention on providing suggestions for tutorials updating. The results of the work indicate the top 20 main topics in the educational programs of SE field. At the same time, work [9] conducted a study of the problems and prospects of SEE based on a questionnaire survey of teachers and experts. The results showed that the most used in software engineering lectures is the problem-based learning approach, followed by gamification methods and role-playing games, which are new trends to engage students. These conclusions are also confirmed in the article [10], which provides a systematic reflection of the issues of gamification in software engineering education: Based on literary sources, it is also shown that the application of gamification in SEE is now more aimed at attracting students and, to a lesser extent, improving student knowledge.

Another systematic review [11] aims to characterize the teaching and learning of the "Software Requirements" discipline introduced as mandatory in most software engineering curricula. It is noted that the process of teaching and studying this discipline is associated with a number of problems for both teachers and students. To mitigate these problems, some review articles suggest techniques such as providing students with realistic and applied environments to create more engaging and practical learning.

Systematic reviews can provide useful knowledge for the practice of software engineering by consolidating and synthesizing empirical studies related to a specific topic. The work [12] used data obtained by of a tertiary study, i.e. conducting a systematic review of published systematic reviews. It is argued that the use of such more rigorous studies can lend greater authority to the findings of systematic reviews and help build a body of robust empirical evidence that is relevant to teaching and practice.

Thus, the scientific and methodical prerequisites of education in SE field require further research in the context of the priorities of modern technological trends. The conceptual basis of the study of these issues is progressing, and the features of the educational environment are constantly developing. Therefore, the relevance of the task consists in taking into account these processes in the methodical principles of teaching in the design and development of a quality software product and ensuring their wider coverage in educational activities

3. The theoretical backgrounds research methodology

The starting point of the research, which follows from the above analysis of sources, is the recognition of the tangible impact on the content, methods and organizational forms of training in SE of the characteristic features of the industry and the IT sphere, which at the current stage consist in the growth of the dynamics of technological changes, in a sharp increase in the amount of information needed for processing and learning.

Researchers and specialists who offer innovative approaches to support SE educational process, in their majority, rely on relevant models and methods, among which the prevailing ideas are: a harmonious combination of related disciplines and various industry technologies; the importance of

creating appropriate training programs and choosing a suitable teaching method on the relevant topics of technological trends; cooperation of subjects of the industry regarding the common goals of education.

After a long hiatus from the release of the SE 2014 (Software Engineering Curricular Volume), international organizations have returned to considering the guiding principles of bachelor's programs in computer professions, in particular SE. One of the four main principles followed by the task force in the development of CC2020 is related to the need to take into account future trends and visions from industry in the educational process, to monitor changes in technology and research from the entire spectrum of society.

From the analysis of this document and other sources, it can be concluded that all computer disciplines, especially SE, should focus on professional know-how of industrial and information technology trends. On the other hand, computer disciplines, like no other, need the support of the educational process precisely by means of information technologies. Actually, these two factors raise questions about the need to understand what attention are paid by scientists of the world community to these areas, as well as which publications indicate promising research in this area.

This systematic literature review followed the guidelines proposed by [13, 1] and other authors should include at least three main stages: planning, conducting and reporting. At these stages, it is advisable to implement the same steps. First, to identify the research questions related to a specific topic, including the definition of components and acceptance criteria. Next, define a comprehensive strategy for the systematic search of research candidates and implement it. The necessary data should be extracted from the received corpus, and the results should be interpreted and presented. Finally, it is advisable to evaluate the validity of the conclusions based on the relevant bibliography.

Research questions are the starting point of a systematic review, as research directions and methods of data analysis are determined based on them. This study has in mind the following research questions:

RQ1: What technological trends are analyzed in the research?

RQ2: What information technologies are considered in the studies?

RQ3: What strategies are reflected in the publications to overcome the above challenges of computing education, teaching, and learning?

As inclusion and exclusion criteria, at least at the beginning of the study, it is suggested to rely on the relevance and pertinence of search results for various search queries. The study suggests looking at Elsevier search engine results in the Web of Science Core Collection when searching by publication title, keywords, and abstract. In addition, the criterion of the number of publications found for the search terms is used.

Searching for sources begins with the formation of a text corpus of research based on search queries that reflect the subject area, using relevant keywords such as software engineering, training, education, etc., and their combinations. Publications are considered for the generally accepted time span of 5 years, i.e., for 2017-2021.

Further, in the obtained data array, it is proposed to select publications that consider modern technological trends that already exist or are expected in the near future and cause a constant increase in the information load in the field of SEE. The list of trends is prepared based on CC2020 and information from leading IT companies, as well as work [2], which identifies some of the main sources of changes that will affect the practice of software development over the next few decades, as well as some strategies for evaluating and adapting to these sources of change. On this basis, Ishikawa diagrams (Figure 1) were developed to identify the most significant factors (reasons) that affect the final result. This chart helps identify the main trends that make the most significant contribution to the vision of the picture.

The next step is to select publications in a different direction, namely by technologies and learning tools, the totality of which is determined based on the modern stage of their development and application (Figure 2).

In the future, it is necessary to work out the statistics of the results in order to find answers to the proposed research questions.
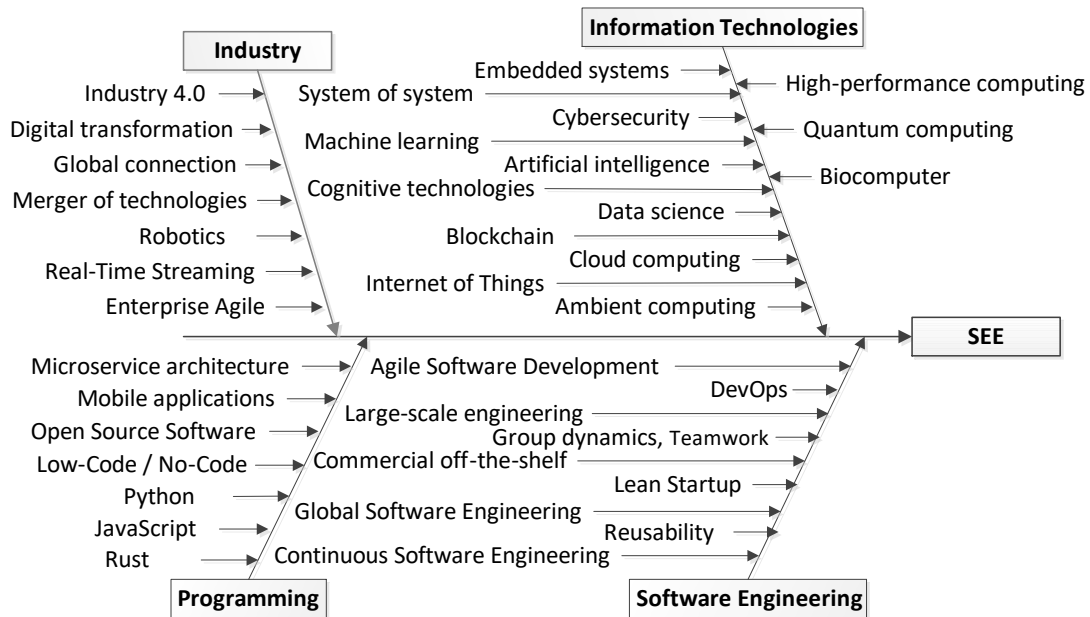
Figure 1: Ishikawa diagrams of the main trends that have the most significant impact on Software Engineering Education
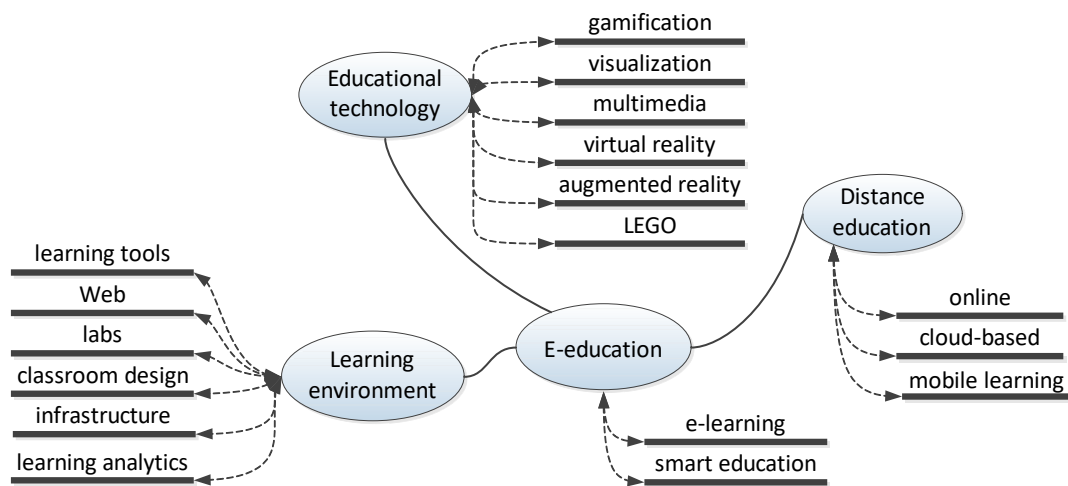

Figure 2: Diagrams of information technologies and teaching resources in Software Engineering Education

4. Research findings
4.1. Approach to publications selection

The implementation of the proposed methodology was started with a search query (educati* OR learning OR teaching OR training OR curriculum* OR course) AND software engineering, which provided a significant number of results. Their preliminary analysis showed low relevance of the search. Further refinements resulted in a search for teaching AND "software engineering" that produced a report of 746 results, of which 568 were conference proceedings and 170 were articles. The conducted analysis has already shown a sufficiently high pertinence of the results, as a result of which a decision was making to conduct further research on this corpus.

Regarding the general characteristics of the obtained material, it should be noted that the number of citations of the selected works at the time of the research was 1718, that is, on average; it is not much – only 2,3 per document. From the chart of publications and citations due to a decrease in the number of publications (Figure 3), it follows that starting from 2019, a significant decline in the number of publications, and later in the number of citations, began. To some extent, this can be explained by the Coronavirus Disease (COVID-19) pandemic, but other searches clearly do not support this assumption.
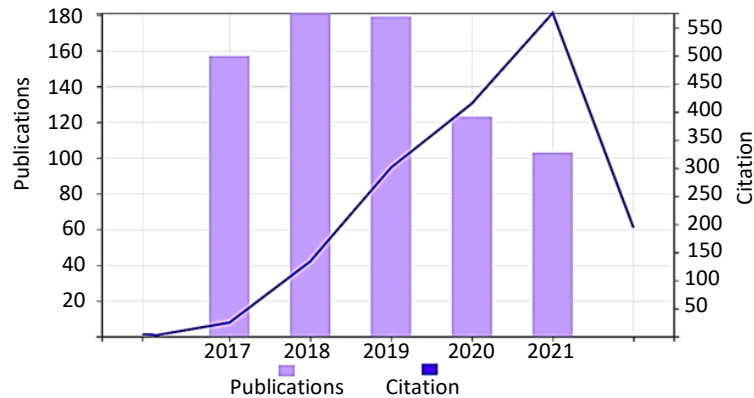

Figure 3: Chart of the number of publications and citations for the teaching AND "software engineering" query

The decline of researchers' interest in the topic of software engineering education is rather related to "reaching a plateau" - the saturation of publication materials for more than forty years of research in this field and a certain stabilization of the development of international recommendations on computer education.

According to Elsevier's criteria, the work that best meets the search conditions is [14]. This work examines Flipped learning of software engineering and proposes a method of its reformation based on the project approach. The work that received the largest number of citations – 60 – is the above-mentioned work [1].

In the Table 3 – 3 provide data on the top 10 by the number of publications, respectively, by institutions, with which the publications of the sample are affiliated, countries, and authors.

Table 1

Top 10 institutions

Institution	Number of publications	Percentage of 746
Universidad politecnica de Madrid	11	1.48
Universidade Federal do Para	11	1.48
University of North Carolina	11	1.48
Babes Bolyai university from Cluj	10	1.34
Universidade Federal de Santa Catarina	10	1.34
Technical university of Munich.	9	1.21
Dhirubhai Ambani institute of information and communication technology	8	1.08
KU LEUVEN	8	1.08
California State University system	7	0.94
Carnegie Mellon University	7	0.94

Table 2

Top 10 countries

Country	Number of publications	Percentage of 746
USA	104	13.94
Brazil	99	13.46
Germany	78	10.46
Spain	60	8.04
China	58	7.77
England	28	3.75
Mexico	26	3.46
Canada	24	3.22
Finland	20	2.68
Australia	19	2.55

Table 3

Top 10 authors

Author	Number of publications	Percentage of 746
Mottok J.	11	1.48
Reuter R.	9	1.21
Tiwari S.	8	1.07
Conte T.	7	0.94
Oliveira SRB.	7	0.94
Singh P.	7	0.94
Brockmann P.	6	0.80
Gold-Veerkamp C.	6	0.80
Steinmacher I.	5	0.67
Vescan A.	5	0.67

4.2. The selection publications results

Based on results of publications selection, a diagram was constructed (Figure 4), which includes 15 trends from those shown in Figure 1, which have the largest number of publications. The given data shows a percentage of the sample size (out of 746).
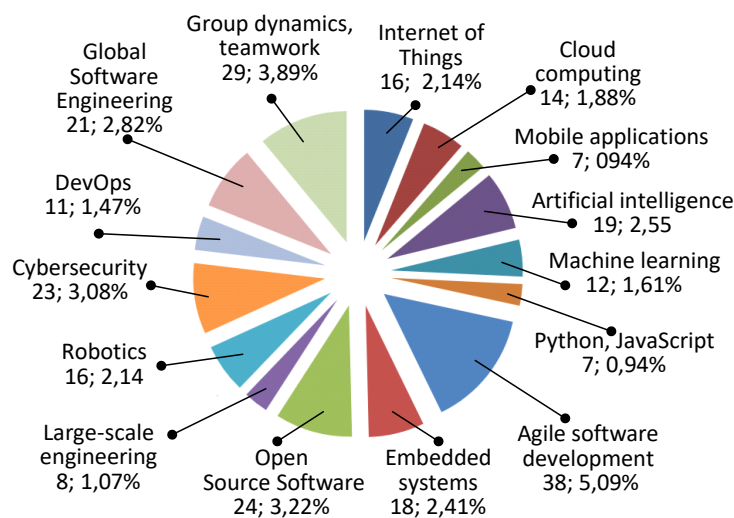

Figure 4: Chart of the number of publications considering current technological trends

According to the diagram in Figure 4, without taking into account the possibility of using several terms in one work, almost a third of the works - 32.6% consider modern technological trends. The chart does not include such trends as Industry 4.0 (1 publication), Digital transformation, Digitalization (4), System of system (1), Quantum computing (1), Lean Startup (1), COTS (1), Real Time (7), Global connection (0), Merger of technologies (0), Enterprise Agile (0), Cognitive technologies (0), Data science (4), Blockchain (0), Ambient computing (0), Biocomputer (0), High-performance computing (1). In fact, the general term Industry is mentioned in 133 works (17.7%), which indicates a significant attention of researchers to industrial challenges.

Similarly, the diagram of Figure 5 was constructed, which includes 15 means of information technologies supporting the educational process from those shown in fig. 2, which received the largest number of publications. Not included in this chart Reusability (1), Continuous Software Engineering (0), Microservice architecture (1), Low-Code / No-Code (0), Rust (1).

According to the diagram in Figue 5, without taking into account the possibility of using several terms in the work, already more than half of the works - 53.4% are devoted to the consideration of issues of technological means of education.
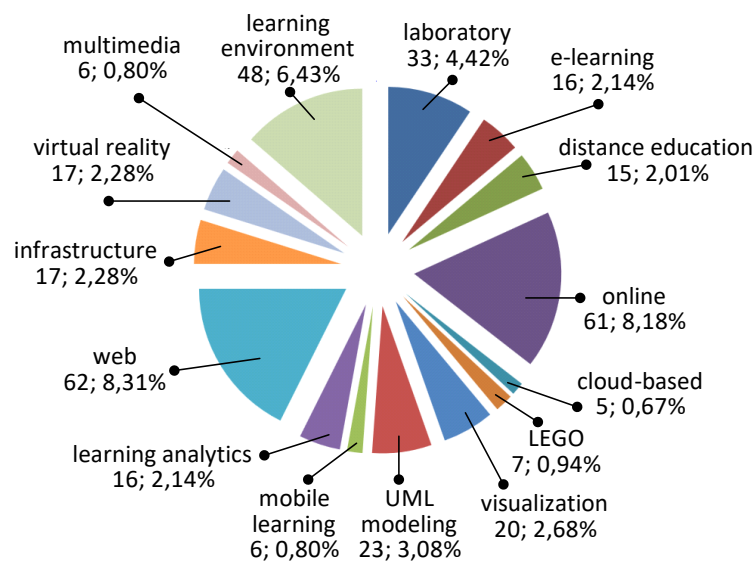

Figure 5: Chart of the number of publications dealing with technological learning tools

In this way, two groups of publications with a total volume of approximately three quarters of the corpus were formed, the analysis of which mainly provides answers to the review questions RQ1 and RQ2. But what do the authors of another quarter of the works write about? As a result of a more careful examination of this ensemble of publications, several significant collections of works were found that considered certain technologies and teaching methodologies (Figure 6).

A group of other fairly diverse publications was also identified. In relation to technological trends, traditional methodologies are mentioned: Project Management (44), Business Information Technology (1), Model-Driven Engineering, or Model-Based Software Engineering (5), Software Language Engineering (1), Empirical software engineering (5), Requirement engineering (4), Human computer-interaction HCI (9), Programming paradigms (4), Software quality (1), COCOMO (1), as well as new trends such as Green and Sustainable Software Engineering (2), Search-based Software Engineering (1), Cyber-physical systems (5), Multi-Agent Systems (2), Big Data (5), and some others.

In a group of other articles devoted to educational technologies, such concepts are used as Smart education, or Smart pedagogy (5), Case-Based Learning (2), Problem-based learning (3), Teaching modelling (5), Computational thinking and constructionism(6), Learning tools (6), Classroom design (1), Virtual Campus (1), Learning ontology (3), Android-based teaching (2), STEM (4), Collaborative Learning (4), Cooperative Learning Method (3), Eye Tracking (4), MOOCs - Massive Open Online Courses (4), Conceptual data modelling (2), Software Testing Educating (23), Teach Requirements

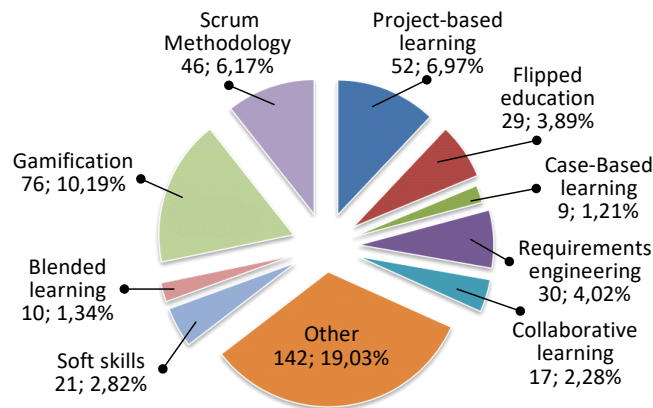Engineering (6), Startup Software Development Education (5), Micro-Learning Application (1), and some others.

Figure 6: Chart of the number of publications considering other technological means and methodologies

4.3. Strategies reflected in publications

In order to find an answer to RQ3, naturally, it is necessary to conduct an analysis of publications that simultaneously used terminology from two or even three groups of publications identified as a result of the review. Consequently, 215 such works were found (28.82%).

Therefore, for example, in work [15] four terms from different groups are used - Large-scale engineering, Learn environment, Gamification, Flipped. The purpose of this study is to apply an innovative pedagogy called the flipped classroom to create a student-centered learning environment for software development, particularly in the direction of industrial scale projects. In work [16] 8 terms are used – IoT, Global Software Engineering, e-learning, Smart education, Collaborative learning, Cloud-based learning, Web learning, Project-based learning. The paper presents a series of distributed virtual intellectual e-learning courses conducted by universities in Japan, Germany and Sri Lanka based on contextual and project-based learning, collaborative learning and teaching using video conferencing and cloud platforms. These innovations are offered in response to industrial challenges in the ability to manage distributed projects and cross-cultural communication skills to collaborate effectively in international teams. In [17], which also practices several terms from different groups, the use of special software for real-time collaboration of distributed companies as a learning tool in the field of SEE is proposed. This paper concludes that students actively participate in such challenges and are more motivated to practice their skills during online exercises.

A significant contribution to improving the industrial relevance of higher education in SE is teaching more efficient ways of working as a software developer. One of them is the automation of repetitive parts of work to help focus on tasks that directly create value for customers and the business. For this, the achievements of artificial intelligence and robotics should be involved. The document [18] notes that robots are a popular platform for introducing novice programmers to computing and artificial intelligence. The paper presents a platform for students to safely and easily program high-performance robots using JavaScript. At the same time, in the article [19] it is generally proposed to combine the teaching of artificial intelligence and software engineering, since both are engaged in modeling real cases, having common elements. In addition, it emphasizes the importance of students following the best practices of industrial teams.

Finally, it would be interesting to know about the research interests of the most active authors identified as a result of this review (see Table 3). Take for example the work of the first author Mottok J., most of which, by the way, was written in co-authorship with the second-ranked Reuter R. An analysis of their works shows that they use such terms as Active video watching, Soft skills, Eye movements, UML Modeling, Competence Assessment, Reflection, Competence Development, Secure Software Engineering, IT Security, SEMAT (Software Engineering Method and Theory); Digital

Game-Based Learning (DGBL), Open Source Software, Start-up education; Project-based learning; Internet of Things, Inductive teaching, Heuristic, Augmented Reality, Flipped classroom; Programming robotics; Virtual Classroom, Distance learning; CDIO (Conceive-Design-Implement-Operate), SAFe (Scaled Agile Framework). The list shows that the strategies reflected in publications of these authors are mainly aimed at overcoming the problems of improving teaching and learning, and to a lesser extent at analyzing the impact of industry trends. This actually agrees with authors strategies other publications that were considered.

On completion, it is necessary to pay attention to the significant variety of terms and definitions used in the publications revealed as a result of the conducted review. Related to this is the serious problem of different understandings and/or interpretations of terminology inherent in the educational environment (the same, of course, happens in other areas). Appropriate information models, in particular based on computer ontologies, which have recently gained considerable popularity in various fields, can help alleviate such communication problems [20]. In [21], one of the first issues raised was the use of ontologies in software research and software projects. Mainly, this work highlights two applications of ontologies in software engineering and software technology: sharing knowledge about a problem domain and using a common terminology among all stakeholders. In turn, the goal of the article [22] is an attempt to propose an ontology for informational decision support in Software Engineering Management tasks.

Having ontological approach, it is possible to display the relationship of the terms used in the publications of the sample (Figure 7). A fragment of such an ontology provided to simplify visualization. This example is actually a meta-ontology of the domain to which this study is devoted.
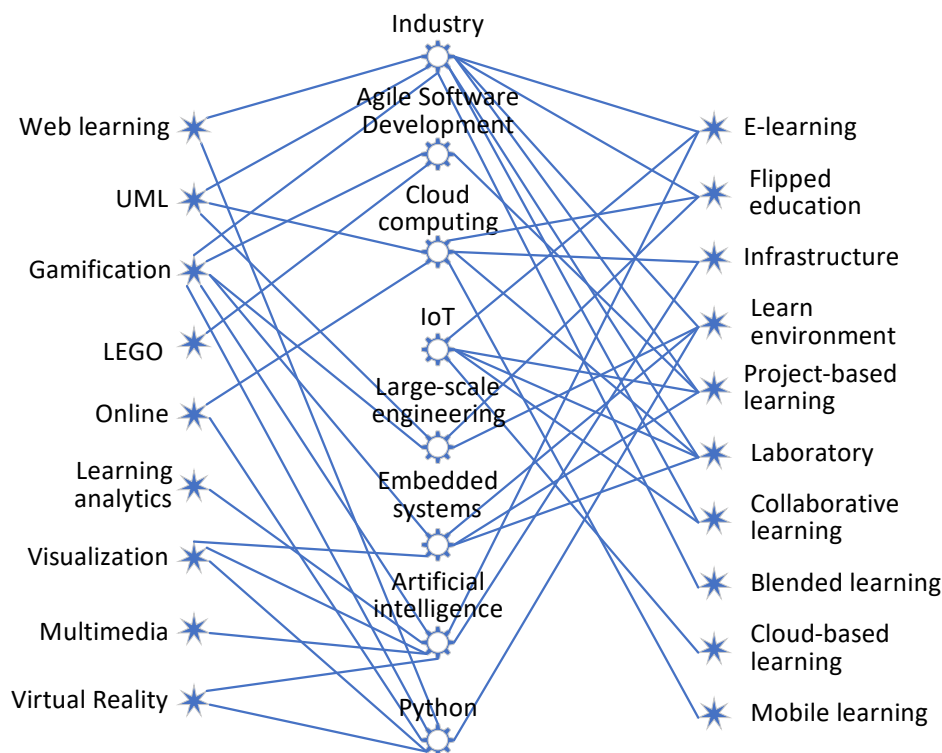

Figure 7: Meta-ontology of terms used in publications

Within the framework of the general problem of setting up the environment of educational activity based on information systems, the task of developing reasonable methods of processing informational educational resources is relevant. Ontological modeling is a good basis for the processes of intellectualization of such systems and the construction of appropriate technological software tools on their basis.

5. Conclusions and prospects for further research

The global industrial demand for highly qualified software engineers in the context of digital transformation is continuously growing. Many countries are constantly experiencing a shortage of developers. On the other hand, the analysis shows that there are gaps between the industry, technological trends and education. Many modern trends are reflected in publications relatively minor.

The industry's need for software engineers will grow faster than educational institutions can train new personnel. It is believed that the main reasons for this deficit lie in the slowness of the adaptation of the education system to the current needs of the market and in the difficulty of students acquiring relevant practical skills. Thus, the key solution should be to increase the level of industrial and market relevance of education.

In today's conditions, educational activities in general and SE in particular, are characterized by constantly growing unstructured textual and multimedia content, which is demonstrated by the conducted research. In practice, processing this content is associated with tangible problems. This controversy gives rise to the restraint of innovative development of the field of education with SE, which negatively affects the pace of its actualization. To some extent, it is proposed to fill the above-mentioned gaps on the basis of information modeling of the educational environment and the introduction of appropriate progressive information technologies to support the educational process.

Systematic reviews of the literature, similar to the one conducted, can be the basis for the formation of information resources maintained by information systems supporting educational activities. These resources can be used not only by responsible persons and by university teachers for the formation and development of relevant educational and methodical materials, but also directly by students in the educational process.

6. Acknowledgements

7. References

[1]   K. T. Al-Sarayreh, K. Meridji, A. Abran, Software engineering principles: A systematic mapping study and a quantitative literature review, Engineering Science and Technology 24(3) (2021) 768–781. doi:10.1016/j.jestch.2020.11.005

[2] B. Boehm, A view of 20th and 21st Century Software Engineering, in: Proceedings of the 28th International Conference on Software Engineering, ICSE'06, 2006, pp. 12–29. doi:10.1145/1134285.1134288

[3] O. Cico, L. Jaccheri, A. Nguyen-Duc, H. Zhang, Exploring the intersection between software industry and Software Engineering education – A systematic mapping of Software Engineering Trends, Journal of Systems and Software 172 (2021) 110736. doi: 10.1016/j.jss.2020.110736

[4] V. Yu. Bykov, M. P. Shyshkina, The conceptual basis of the university cloud-based learning and research environment formation and development in view of the open science priorities, Information Technologies and Learning Tools 68(6) (2018) 1–19. doi:10.33407/itlt.v68i6.2609

[5]   M. Rostoka, A. Guraliuk, G. Cherevychnyi, O. Vyhovska, I. Poprotskyi, N. Terentieva, Philosophy of a Transdisciplinary Approach in Designing an Open Information and Educational Environment of Institutions of Higher Education, Revista Romaneasca pentru Educatie Multidimensionala 13(3) (2021) 548-567. doi: 10.18662/rrem/13.3/466

[6]   Y. G. Sahin, U. Celikkan, Information technology asymmetry and gaps between higher education institutions and industry, Journal of information technology education-research 19 (2020) 339–365. doi:10.28945/4553

[7]   H. Cornide-Reyes, S. Campillay, A. Alfaro, R. Villarroel, Identification of skills for the formation of agile high performance teams: a systematic mapping, in: Proceedings of the 6th

Conference on Information and Communication Technologies of Ecuador, TIC-EC, 2019, pp. 141–152. doi: 10.1007/978-3-030-02828-2_11

[8]   N. D. Acosta, M. B. Garcia, J. B. Romero, A. L. Sevillano, N. F. Paez, O. Roa, Educational trends in software engineering: a systematic review study, in: Proceedings of the International Congress of Innovation and Trends in Engineering, CONIITI, 2019. doi:10.1109/ CONIITI48476.2019.8960866

[9]   S. Ouhbi, N. Pombo, Software Engineering Education: Challenges and Perspectives, in: Proceedings of the IEEE Global Engineering Education Conference, IEEE EDUCON, 2020, pp. 202–209. doi:10.1109/EDUCON45650.2020.9125353

[10] M. M. Alhammad, A. M. Moreno, Gamification in software engineering education: A systematic mapping, Journal OF Systems And Software 141 (2018) 31–150. doi:10.1016/j.jss.2018.03.065

[11] M. Pereira, V. Mendoza, D. Assis, A. Farias, R. Fagundes, M. Lencastre, A systematic review of the literature on the teaching and learning of requirements engineering, in: Proceedings of the 9th International Conference on Education and New Learning Technologies, EDULEARN, 2017, pp. 1340–1349. doi: 10.21125/edulearn.2017.1283

[12] D. Budgen, P. Brereton, N. Williams, S. Drummond, The contribution that empirical studies performed in industry make to the findings of systematic reviews: A tertiary study, Information and software technology 94 (2018) 234–244. doi: 10.1016/j.infsof.2017.10.012

[13] B. Kitchenham, D. Bugden, O.P. Brereton, The value of mapping studies – A participant-observer case study, in: Proceedings of the 14th International Conference on Evaluation and Assessment in Software Engineering, 2010, pp. 1–9.

[14] X. M. Bai, Q. Wang, Research on Teaching Reform of Software Engineering Course based on Flipped Classroom, in: Proceedings of the 3rd Conference on Education and Teaching in Colleges and Universities, CETCU, 2017, pp. 301–304. doi: 10.2991/cetcu-17.2017.77

[15] Y. T. Lin, Impacts of a flipped classroom with a smart learning diagnosis system on students' learning performance, perception, and problem solving ability in a software engineering course, Computers in human behavior 95 (2019) 187–196. doi: 10.1016/j.chb.2018.11.036

[16] D. M. Marutschke, V. Kryssanov, H. T. Chaminda, P. Brockmann, Smart Education in an Interconnected World: Virtual, Collaborative, Project-Based Courses to Teach Global Software Engineering, in: Proceedings of the 6th KES Annual International Conference on Smart Education and e-Learning, KES SEEL / 2nd KES International Symposium on Smart Transportation Systems, KES-STS, 2019, 144, pp. 39–49. doi: 10.1007/ 978-981-13-8260-4_4

[17] G. Seuchter, E. Teiniker, Real-time collaboration tools for distance learning, in: Proceedings of the 10TH International conference on education and new learning, EDULEARN, 2018, pp. 4106–4117. doi: 10.21125/edulearn.2018.1040

[18] J. Spitzer, J. Biswas, A. Guha, Making High-Performance Robots Safe and Easy to Use For an Introduction to Computing, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence / 32nd Innovative Applications of Artificial Intelligence Conference / 10th AAAI Symposium on Educational Advances in Artificial Intelligence, 2020, 34(09), pp. 13412–13419. doi:10.1609/aaai.v34i09.7065

[19] L. Diosan, S. Motogna, Artificial Intelligence Meets Software Engineering in the Classroom, in: Proceedings of the 1st ACM SIGSOFT International Workshop on Education through Advanced Software Engineering and Artificial Intelligence, EASEAI, 2019, pp.35–38. doi:10.1145/3340435.3342718

[20] O. Nesterenko, O. Trofymchuk, Patterns in forming the ontology-based environment of information-analytical activity in administrative management, Eastern-European Journal of Enterprise Technologies 101 5/2 (2019) 33–42. doi:10.15587/1729-4061.2019.180107

[21] M. Piattini, C. Calero, F. Ruiz, "Ontologies for Software Engineering and Software Technology", Springer, 2006. doi10.1007/3-540-34518-3

[22] O. Nesterenko, Yu. Selin, The Teams Information Model for Software Engineering Management, in: Proceedings of the IEEE 16th International Conference on Computer Sciences and Information Technologies, CSIT, 2021, 1, pp. 341–344. doi:.10.1109/ CSIT52700.2021.9648737