Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35) *ISSN 2522-9818 (print)*
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35) *ISSN 2524-2296 (online)*

UDC 004.75:37.018.43 DOI: https://doi.org/10.30837/2522-9818.2026.1.127

Oleksii Shapyro, Ihor Sotnyk

A METHOD OF SEMANTIC SEARCH FOR EDUCATIONAL CONTENT BASED ON MULTI-AGENT TECHNOLOGIES

The digital transformation of industry is accompanied by the active adoption of new technologies and the rapid evolution of production processes. A significant portion of educational materials is distributed across various information sources, including internal corporate systems, open educational platforms, and specialized web resources. Such resources often contain duplicates, redundant information, and heterogeneous metadata, which complicates the timely retrieval of relevant learning materials. The **subject** of the study is a method of semantic search for educational content in a distributed information environment using ontology-based knowledge models. The **goal** of the work is to investigate a method of semantic search for educational content in a distributed information environment based on a multi-agent organization of information resource processing and the use of ontology-based knowledge models. The **objectives** of the study are: to investigate the architectural model of a multi-agent search system; to develop a semantic selection algorithm based on the comparator identification method and an ontology-based model; to formalize a relevance evaluation predicate considering weighted metadata coefficients; to develop a multi-agent software platform; and to experimentally evaluate performance and resource consumption under different agent operating modes. Research **methods** include: the method of multi-agent organization of information resource processing with non-blocking message exchange; three-level URL deduplication; ontology-based term matching and a formalized relevance evaluation predicate; and experimental measurement of processing time, the number of processed links, and system resource consumption. **Results**: a model of a multi-agent system with four types of agents and a semantic search algorithm eliminating loops and duplicate links has been proposed; a software platform based on Kotlin using coroutines and asynchronous interaction between agents has been implemented; experimental results demonstrate that the proposed organization of processing provides higher performance compared to the sequential mode. **Conclusions**: the integration of semantic search and a multi-agent architecture enables efficient organization of the process of discovering and processing educational content in a distributed environment. The proposed method ensures coordinated operation of agents, eliminates link duplication, and provides a rational balance between search completeness and the use of computational resources.

Keywords: semantic search; educational content; software agents; multi-agent systems; ontologies; scalability; performance; asynchrony; Kotlin.

Introduction

The current stage of information technology development in industry and related manufacturing sectors is characterized by a growing volume of educational content and increasing demands on the effectiveness of e-learning systems. The digital transformation of enterprises, rapid updates to production equipment, hardware and software systems, and technological regulations create a constant need for the continuous improvement of staff qualifications. Under these conditions, the availability of relevant educational content directly impacts the quality of technological operations, the level of workplace safety, compliance with standards, and the reduction of the risk of errors and downtime. The rapid growth in the volume of open educational content is accompanied by a sharp diversity of sources, formats, and the quality of descriptions. A significant portion of the materials is published outside of specialized platforms, with minimal or inconsistent metadata, which complicates the automatic discovery, classification, and subsequent reuse of resources in learning environments. For industrial scenarios, this is further complicated by the fact that training materials are often scattered across internal enterprise knowledge bases, corporate LMSs, and external resources (suppliers, technical documentation, courses, standards), creating knowledge fragmentation and additional barriers to rapid learning.

Traditional keyword-based search approaches yield inconsistent results precisely in educational scenarios, where semantic alignment with the subject domain, learning objectives, and the expected level of proficiency is crucial. For example, in industry, the same terminology can have different meanings depending on the type of equipment or standard, and content requirements are determined not only by the general topic but also by the employee's role, access level, operational context, and other factors that must also be taken into account. Additional factors contributing to quality degradation

ISSN 2522-9818 (print) Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2026. No. 1 (35)

include duplicates and semi-duplicates, different language versions of the same material, time-based revisions without proper versioning, as well as noise in the form of advertising and navigation elements. In the absence of structured descriptions, even routine tasks require consideration of the context of the subject ontology, not just term matches.

At the same time, the requirements for the scalability and resilience of such systems are growing: data collection and preprocessing must handle arrays of tens of thousands of links, withstanding peak loads on memory, the processor, and the network, while maintaining controllable latency and fault tolerance. In practical terms, this is compounded by external constraints, which complicate the maintenance of consistent throughput and reproducibility of results within corporate infrastructure.

All these discrepancies between the needs of real-world educational applications for industrial workforce upskilling and the capabilities of existing search approaches highlight a problem requiring further research: search quality directly impacts the speed of learning, safety, and the effectiveness of production processes.

Literature Review and Problem Definition

The IEEE LTSC (Learning Technology Standards Committee) [1] was established to develop specifications and recommendations for the integration of modern information technologies in education. This committee develops architectures and technologies for educational systems in accordance with international standards. Separately, the IMS Global Learning Consortium is engaged in developing standards to ensure the interoperability of educational platforms [2]. According to the IEEE LTSA (Learning Technology System Architecture), the universality of this architecture ensures its use as a reference model for the development of educational technology systems of any type. It has been shown that these standards form the basis for the interoperability of various educational platforms and ensure a unified approach to organizing the educational process. However, issues remain unresolved regarding the adaptation of these general architectural recommendations to the specific needs of intelligent semantic search for educational content and the lack of concrete algorithmic solutions for the automated analysis of educational resources. This may be due to objective difficulties related to the excessive abstractness of international standards, which focus on general principles rather than specific technical implementations, making their application challenging for practical semantic search tasks.

In [3], key trends in adaptive e-learning systems were investigated based on an analysis of 57 studies from recent years. It was shown that building effective adaptive systems requires learner, domain, and adaptation models that account for individual learning characteristics and dynamically adjust instructional materials. However, issues remain unresolved regarding the fact that most existing systems do not provide sufficient accuracy in predicting learner needs in real time and do not adapt to the context of learning activities. This is due to objective difficulties related to the complexity of collecting and processing large volumes of data on student behavior, as well as the inherent uncertainty of individual cognitive processes, which makes universal solutions economically impractical.

In [4], an ontology-driven framework for semantic search of educational content is proposed, utilizing natural language processing and machine learning methods for text classification. It is shown that the ontological approach allows for the identification of semantic relationships between concepts and the discovery of the most relevant materials compared to traditional keyword search. However, issues related to the automation of ontology creation and updating for various subject areas remain unresolved. The reason for this is the cost involved in engaging experts to develop high-quality ontologies and the fundamental complexity of automating the processes of identifying semantic relationships in dynamically evolving knowledge.

In [5], the application of the latest approaches to semantic search using ontological models and document relevance analysis methods was investigated. The potential of these methods for improving the accuracy of identifying relevant educational materials was demonstrated. However, issues remain unresolved regarding maintaining a balance between search accuracy and system scalability when processing large repositories of educational objects, as well as the integration of such systems with existing educational content metadata standards. The reason for this is the difficulty associated with the fact that complex machine learning models often require significant computational resources, making their application impractical in distributed environments with limited resources.

Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35) *ISSN 2522-9818 (print)*
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35) *ISSN 2524-2296 (online)*

In [6], decentralized multi-agent systems for solving planning and optimization problems in manufacturing environments were investigated. It was shown that the multi-agent approach ensures scalability and resilience to failures of individual components. However, issues remain unresolved regarding ensuring effective communication between agents when there are a large number of participants and adapting agent-based systems to the specifics of educational applications with semantically rich data. The reason for this is the exponential growth in the complexity of communication protocols and the fundamental difficulty of predicting the behavior of a large number of interacting agents.

Additionally, many studies leave unresolved issues related to practical implementation on modern programming platforms and their adaptation to the requirements of high-performance real-time data processing. This is due to difficulties stemming from the fact that many existing implementations are based on outdated technologies and do not support modern asynchronous programming mechanisms. One way to overcome these difficulties is to use modern programming languages with built-in support for asynchronous programming and lightweight parallelism mechanisms. For example, some case studies describe the use of coroutines and asynchronous operations to organize effective communication in distributed systems. It has been shown that this approach allows for increasing system throughput and reducing resource management costs.

Research Objectives and Tasks

Systematizing the local problems identified in the analyzed sources allows us to formulate a general unresolved problem: there is currently no approach to searching for educational content that is simultaneously effective in terms of result relevance, performance, scalability, and resource efficiency. Its relevance stems from the need to quickly find relevant educational materials from a vast number of sources, taking into account subject-area ontologies, content duplication and noise, as well as computational resource constraints. Several factors are the main reasons this problem remains unsolved. **First**, the lack of solutions that combine multi-agent architecture with semantic technologies. **Second**, the inefficiency of traditional algorithms for processing heterogeneous metadata of educational resources. **Third**, the difficulty of scaling distributed systems without compromising service quality. **Fourth**, many existing implementations are based on outdated technologies that do not support modern asynchronous programming mechanisms. All of this supports the need for relevant research to address this issue.

Accordingly, the objective of this work was defined as the need to develop an approach to educational content search that is effective in terms of both the relevance of results and performance, scalability, and resource efficiency.

To achieve this objective, the following tasks were set:

– to develop an architectural model of a multi-agent search system;

– create a semantic search algorithm using the comparator identification method and ontological models of subject domains;

– implement a software platform using modern technologies with built-in support for asynchronous programming and lightweight parallelism mechanisms;

– experimentally evaluate the system’s performance and resource consumption in various agent operating modes.

Materials and Methods

In the context of semantic search for educational content, it is necessary to process large volumes of information resources located in a distributed web environment and characterized by high dynamism and structural heterogeneity. Traditional centralized information retrieval systems may face limitations in scalability, performance, and flexibility of resource processing control under such conditions. One effective approach to organizing search systems in a distributed environment is the use of multi-agent technologies, which allow the system’s functionality to be decomposed into autonomous software components that perform specialized tasks and interact with one another to achieve a common goal.

A software agent is an autonomous software component capable of functioning in an information environment, perceiving its state, making decisions according to specified rules, and interacting with other agents to achieve a common goal.

This paper proposes an architectural framework for an ontology-driven semantic search of educational resources using a multi-agent system. The operation of agents is ensured and regulated by a naming service and an access control module. The agent registry is

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)
Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35)
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35)

designed to provide agents with information about the available interaction environment, specifically regarding the composition of the system's agents, their roles, and functional purposes. An ontological model is used to ensure a unified formalized description of the domain, common to all agents. The information resources supporting the system's operation are stored in a database and a knowledge base.

The multi-agent search system is implemented using four functional types of agents:

– the coordination agent manages the search process, generates initial query descriptors, distributes tasks among other system agents, controls the sequence of resource processing, and coordinates message exchange between agents;

– the resource collection agent follows the URLs specified by the coordination agent, retrieves metadata, and transmits it to the semantic evaluation agent for further processing;

– the semantic evaluation agent uses a relevance assessment mechanism based on the search descriptor model, which allows determining the feasibility of further expanding the search from the current web resource, taking into account its thematic relevance;

– the indexing agent ensures the automated generation of a structured description of the found document in the form of metadata in accordance with the selected representation model.

The architecture of the multi-agent system for semantic search of educational content is shown in Fig. 1. The system operates in a distributed information environment that includes a set of software agents, a metadata database, a knowledge base containing an ontological model of the subject domain, as well as an external web environment that serves as a source of information resources.

The interaction of the system components is organized such that the coordination agent initiates the search process, forms initial query descriptors, and distributes tasks among the resource collection agents. Resource collection agents crawl web resources and transmit the obtained metadata to semantic evaluation agents. After semantic analysis is performed, the results are transmitted to the indexing agent, which generates a structured description of the documents and stores it in the database. The ontological model of the subject domain is used by semantic evaluation agents to determine the thematic relevance of resources to the query. This architecture ensures distributed processing of information resources, coordination of agent interactions, and the ability to scale the system through the parallel execution of search procedures.

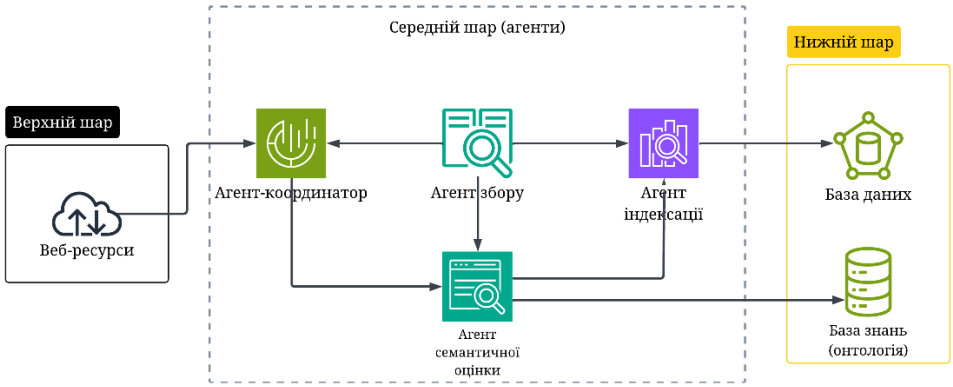

Fig. 1. Architecture of a multi-agent system for semantic search of educational content

The initial stage of the search process involves identifying web resources containing educational materials, which is carried out by a resource collection agent using automated hypertext space crawling technologies. Automated crawling refers to the process of retrieving web resources from the Internet for the purpose of their subsequent indexing and use in a search engine [7]. The main goal of this stage is the rapid formation of a set of relevant web resources along with hyperlinks reflecting their interrelationships. When a URL is obtained that defines an entry point into the search space, the agent downloads the corresponding web resource, extracts the outgoing hyperlinks from it, and adds them to the queue for further processing [8]. The process continues until the queue of resources has been fully processed.

The key factor in the efficiency of a resource-gathering agent is the strategy for processing the

Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35) *ISSN 2522-9818 (print)*
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35) *ISSN 2524-2296 (online)*

hyperlinks accumulated in the queue. In practice, two basic approaches are used: sequential traversal without prioritization and guided selection based on an evaluation function. In the case of sequential traversal, the next resource is selected without considering its content characteristics. In contrast, the guided approach involves the use of evaluation criteria that allow determining the appropriateness of moving to a specific resource. Such algorithms can be based on both additional domain-specific knowledge and formal characteristics of the link structure.

The algorithm is based on a multi-agent organization of the process and ensures distributed processing of the hypertext space, taking into account thematic focus. The search for data sources is carried out by a set of agents distributed throughout the environment, which function as a coordinated system thanks to mechanisms of inter-agent interaction and coordination [9]. To ensure controlled scalability, three complementary mechanisms have been implemented. First, a multi-level duplicate removal system has been introduced, which performs filtering at the resource level, at the level of URLs stored in memory, and between different agents. Second, the procedure for verifying and adding URLs to the shared storage structure has been synchronized, which prevents the duplicate inclusion of identical links. Third, a limit has been imposed on the number of simultaneously active agents, which helps stabilize RAM consumption. The algorithm structure is shown in Fig. 2.

Fig. 2. Algorithm for creating agents

As a result of the resource collection agent's work, a set of URLs is generated in memory; some of these URLs are thematically relevant to the query, but not all resources directly contain educational materials. Therefore, the next stage involves semantic filtering of the collected resources to identify those that actually contain educational objects in the relevant subject area.

The document representation model defines the metadata structure, the query formation mechanism, and the relevance evaluation procedure.

The selected resources are transferred to the indexing agent, where their structured description is generated. Indexing involves constructing the corresponding metadata and storing the obtained information about educational objects for further use in the search process.

A metadata description unit is considered to be a pair (n_i, v_i), where n_i – the name of the i-th element of the learning object's content description, and v_i – the corresponding value of this element. The document's metadata description is then presented as a formula (1)

$$m(d) = \{((n_i, v_i), k_i) | i = 1 \ldots l\}, \quad (1)$$

where k_i – weight coefficient of the i-th element of the metadata description, l – number of elements.

The process of determining semantic similarity between a document and a query is interpreted as establishing a mapping between the structured description of the document's metadata, as expressed by formula (1), and the result of evaluating its thematic relevance. In this case, the input data is a set of elements $m(d)$, and the output is a relevance score determined by comparing it with the search descriptor. This approach allows the use of the comparator identification method [10] to formalize the rule for transforming a set of document features into a semantic evaluation result.

Before initializing the search process, a domain ontology is formed that reflects the key concepts of the

ISSN 2522-9818 (print) Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2026. No. 1 (35)

relevant domain and their semantic relationships. Based on this, a set of terms relevant to the query topic is determined, which may be represented in the structural elements of a web resource, specifically in titles, keywords, and hyperlinks. The selection of these specific elements is due to the fact that in most electronic resources they have a formalized representation via corresponding markup tags, which simplifies their automated extraction. To determine the thematic affiliation of a web resource, the agent performs a comparison of terms from the title h, set of keywords k and the hyperlink text r with the concepts represented in the domain ontology [11]. The resulting thematic description of the resource is then compared with the query topic in accordance with the mappings (2–4).

$$P_k(k,q) = P_k(k,q_1) \vee P_k(k,q_2) \vee \ldots \vee P_k(k,q_l), \quad (2)$$

$$P_h(h,q) = P_h(h,q_1) \vee P_h(h,q_2) \vee \ldots \vee P_h(h,q_l), \quad (3)$$

$$P_r(r,q) = P_r(r,q_1) \vee P_r(r,q_2) \vee \ldots \vee P_r(r,q_l), \quad (4)$$

where q – subject of the inquiry; h – heading terms; k – set of keywords; r – hyperlink text.

The relevance of a page to a search query can be expressed by a predicate (5)

$$S = P(k,h,r,q) = P_k(k,q_1) \vee P_h(h,q_1) \vee P_r(r,q_1), \quad (5)$$

where S – A page score that takes the values 0 or 1.

In the simplest case, a web resource is considered thematically relevant if at least one term from the title, set of keywords, or hyperlink text belongs to the ontologically defined topic of the query. This model corresponds to a binary membership predicate. At the same time, the structure of the predicate can be extended by introducing weighting coefficients for various structural elements of the document, allowing a transition from a strict logical condition to a weighted relevance assessment.

The application of the comparator identification method formalizes the thematic matching procedure within a multi-agent system. Combined with algebraic-logical models [12], this creates a foundation for accounting for the semantic structure of educational objects during their automated selection and indexing.

The Kotlin programming language was chosen for implementing the multi-agent platform due to its support for asynchronous programming and full compatibility with the JVM ecosystem [13]. This choice ensures the solution's scalability and the ability to utilize existing library infrastructure for building distributed systems.

A key factor is the use of coroutines [14], which implement a model of structured concurrency and allow for non-blocking interaction between agents. Unlike the classical thread model, coroutines have lower overhead for execution management and ensure efficient coordination of a large number of simultaneously active agents. This is particularly important for a multi-agent system, where parallel processing and asynchronous message exchange directly impact performance and scalability.

The inter-agent interaction mechanism is implemented using the Kotlin SharedFlow tool [15], which provides a non-blocking model for broadcast event distribution. This approach ensures asynchronous event delivery to dynamically created agents and supports their independent lifecycle. This allows the system to be scaled by adding or removing agents without modifying the communication mechanism. The chosen messaging model minimizes the overhead of synchronization and contributes to the system's fault tolerance in a distributed environment.

The "Factory Method" pattern is used for unified agent creation. Its use allows encapsulating the agent initialization process and ensuring system extensibility without modifying the platform's base code. Adding a new agent type boils down to implementing its behavioral logic and defining the conditions under which it can accept tasks for execution. This approach ensures the scalability of the architecture and simplifies system configuration when expanding functionality.

A mechanism for configurational parameterization of agents has been implemented, allowing their characteristics to be dynamically defined, including the identifier, number of instances, behavioral logic, and additional execution parameters. Agent configuration is performed by loading the corresponding behavioral classes during program execution using the JVM reflection mechanism, which enables the extension of system functionality without recompiling the code and facilitates the flexible configuration of the multi-agent platform according to the requirements of a specific application scenario.

To implement the mechanism for verifying an agent's ability to execute a task, the "Chain of Responsibility" pattern [16] is used. The verification consists of a sequence of handlers, each of which determines whether the agent meets the specified conditions and, if necessary, passes the request to the next element in the chain. This organization allows for

Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35) *ISSN 2522-9818 (print)*
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35) *ISSN 2524-2296 (online)*

the modular formation of a set of task acceptance criteria without complicating the agent's basic logic.

To organize data access, the "Repository" pattern [17] is used, which abstracts the logic of data storage and retrieval from the agents' business logic.

The system's information environment includes:

– a metadata database;

– a knowledge base containing the domain ontology;

– a set of software agents;

– a coordination module;

– an external web environment containing sources of educational resources.

The implemented software platform functions as a multi-agent runtime environment with a console-based control interface and ensures the initialization and interaction of agents within the experimental implementation of the proposed architecture.

Research Results

The following conditions were adopted within the scope of the study:

– educational resources have formalized metadata structured in accordance with IEEE LOM [18] or SCORM [19] standards;

– semantic analysis is performed within a single linguistic interpretation of terms without accounting for cross-cultural differences;

– agent interaction occurs in an environment with hardware-homogeneous infrastructure and a stable network with sufficient bandwidth;

– aspects of information security, authentication, and access control are outside the scope of this study and are treated as external to the proposed architecture.

The research was conducted on a computing platform with an 8-core x86-64 architecture processor operating at a clock speed of 2.5–4.8 GHz and 32 GB of RAM, running the Windows 11 operating system in a JVM environment.

Three agent operating modes were tested. The first mode involved the sequential operation of agents A0, A1, A2, and A3, where each agent completed its task before handing control over to the next. The second mode involved the coordinated sequential operation of 50 agents of type A1 under the control of a coordinator. The third mode implemented the coordinated parallel operation of a set of agents: 50 instances of agent A0, and one instance each of agents A1, A2, and A3.

Performance evaluation was conducted by recording the duration of request processing, the number of processed links, and system resource usage metrics (CPU load and RAM) within specified time intervals (15, 30, 45, and 60 minutes). The results are presented in Table 1, which lists the main system performance metrics.

Table 1. *Testing results*

Characteristics	**Consistent work of agents A0, A1, A2, A3**				**Coordinated consistent work of 50 agents A1**				**Coordinated parallel work of a set of agents A0(50), A1, A2, A3**			
Time (min)	**Database entry**								**Cache and database entry**			
Number of links loaded	15	30	45	60	15	30	45	60	15	30	45	60
Number of links checked	2377	3576	4914	6342	9423	13276	25487	33278	14238	23487	32398	59213
Load	1132	1819	2453	2927	2691	3601	4657	6243	9231	17512	26831	39321
CPU	27	29	32	35	40	43	46	49	49	51	55	57
RAM load (MB)	25	27	28	30	101	103	110	110	439	601	963	965

The results show significant differences in system performance depending on the mode of agent operation. Figure 3 illustrates how the number of links found and verified varies with execution time in sequential mode. Over the course of 60 minutes, the system detected 6,342 links, of which 2,927 were verified, confirming the effectiveness of filtering duplicates and irrelevant resources even during sequential processing. In the second mode, with the coordinated operation of 50 A1-type agents, search efficiency increased significantly. As shown in Fig. 4, in 60 minutes the system found 33,278 links and verified 6,243. Thus, the number of resources found increased approximately 5.24-fold compared to sequential mode, indicating an increase in search coverage and the completeness of educational resource discovery.

The third mode, featuring coordinated parallel operation of a set of agents A0(50), A1, A2, and A3, demonstrated the highest search performance over the same time interval. Figure 5 shows that, with the same execution time (60 minutes), the system found 59,213 links and verified 39,321 links.

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)
Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35)
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35)

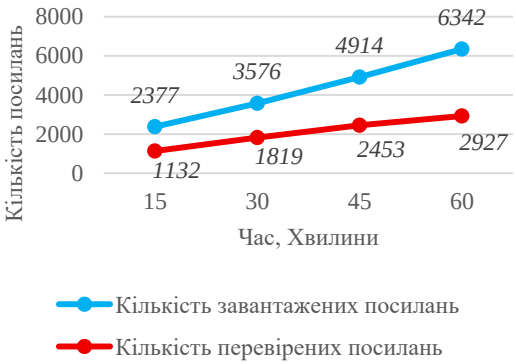

Fig. 3. The relationship between the number of links found (blue curve) and the number of links verified (red curve) as a function of execution time (minutes) in sequential agent mode A0, A1, A2, A3

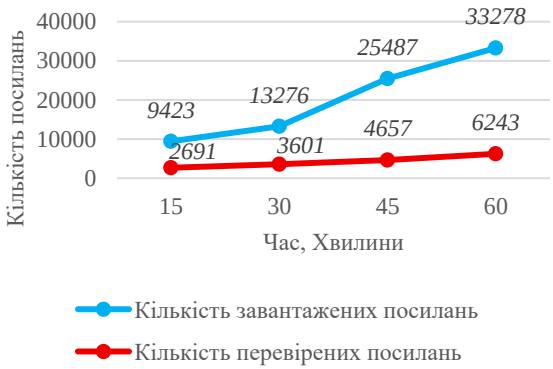

Fig. 4. The relationship between the number of links found (blue curve) and the number of links verified (red curve) as a function of execution time (minutes) in the coordinated sequential operation mode of 50 agents of type A1

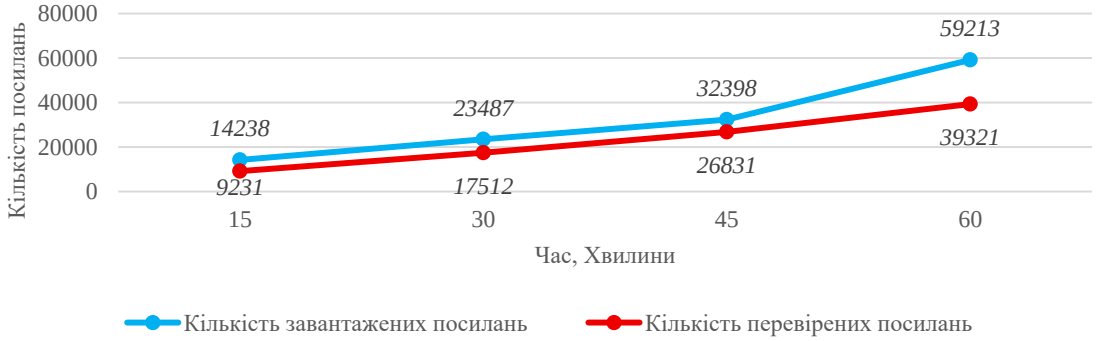

Fig. 5. The relationship between the number of links found (blue curve) and the number of links verified (red curve) as a function of execution time (minutes) in the coordinated parallel operation mode of the set of agents A0(50), A1, A2, A3

A comparison with the first mode reveals a 9.3-fold increase in the number of links found and a 13.4-fold increase in the number of links verified. This indicates an exponential impact of the parallel organization of agent operations on the efficiency of semantic search within the same time constraint.

An analysis of system resource usage metrics revealed significant RAM requirements depending on the operating mode. In serial mode, RAM usage was 25–30 MB with a CPU load of 27–35%. In the second mode, memory usage increased to 101–110 MB with 40–49% CPU utilization due to the simultaneous operation of 50 agents. In the third mode of maximum parallelism, memory usage reached 439–965 MB with 49–57% CPU utilization. It was experimentally established that the limiting factor for performance is the amount of RAM, not the CPU power. The research results demonstrate a significant improvement in the accuracy and completeness of identifying relevant educational resources when using the parallel coordinated operation of a set of agents, with a trade-off in the form of a linear increase in RAM consumption.

Discussion of Results

The obtained results can be explained by the effective use of Kotlin coroutines and optimized algorithms for inter-agent interaction. The experimental data in Table 1 demonstrate a significant difference in performance depending on the mode of agent operation. In the sequential mode, the system found 6,342 links in 60 minutes; in the second mode with 50 agents of type A1, 33,278 links were found; and in the third mode of coordinated parallel operation of a set of agents A0(50), A1, A2, A3, 59,213 links were found. This represents a 9.3-fold increase in efficiency compared to the sequential approach. This increase is attributed to the ability of multiple agents to operate simultaneously without blocking the main execution thread, thanks to the asynchronous nature of Kotlin coroutines and lightweight context-switching mechanisms.

The effectiveness of the semantic search described by equations (2–5) can be explained by the use of the comparator identification method and ontological models of subject domains. The mathematical model for

Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35) *ISSN 2522-9818 (print)*
Innovative technologies and scientific solutions for industries. 2026. No. 1 (35) *ISSN 2524-2296 (online)*

describing metadata according to formula (1) allows for the consideration of weight coefficients ki for each element of the metadata description, which improves the accuracy of identifying relevant educational resources. The relevance evaluation predicate P(k, h, r, q), defined by formula (5), provides a comprehensive evaluation of a document based on three criteria: the relevance of keywords, the title, and the links to the search topic. Such a multi-criteria evaluation significantly exceeds the accuracy of traditional one-dimensional methods based on simple word matching.

The solutions developed address the critical issue of the lack of an effective multi-agent platform that combines semantic search for educational content with modern asynchronous programming technologies. The problem of inefficient document ranking algorithms has been solved by introducing a relevance assessment predicate (5) that accounts for semantic relationships between the three components of a document: titles, keywords, and links. The problem of the complexity of creating universal algorithms to account for the diversity of educational standards (LOM, SCORM) is solved through the use of ontological models that provide a flexible representation of knowledge.

Limitations of the study include dependence on a stable network environment with sufficient bandwidth for message exchange between agents. Additionally, educational resource metadata must comply with LOM or SCORM standards, and assumptions about the uniformity of client device technical specifications are required. Experimental conditions involved testing on sets of up to 60 resources; scaling to larger volumes requires further research.

Conclusions and Prospects for Further Development

In the course of this study, its objective was achieved by completing all set tasks, namely:

1. Develop an architectural model of a multi-agent search system. An architectural model of a multi-agent system for semantic search of educational content, oriented toward four types of agents (scanner, evaluator, indexer, coordinator), was developed. The proposed structure ensures the platform's modularity, scalability, and extensibility.

2. Create a semantic search algorithm using the comparator identification method and ontological models of subject domains. A semantic search algorithm has been implemented, based on the use of a formal relevance evaluation predicate and ontological models of the subject domain. The difference from standard metadata lies in the introduction of weighting coefficients and multi-level comparison (keywords, title, links). Analytical justification is provided by the use of a Boolean predicate function with three independent evaluation components. Experimental studies have confirmed a ninefold increase in semantic search performance compared to standard approaches. The relevance predicate has the structure of a logical conjunction of three independent conditions: the correspondence of the document's keywords to the user's query, the semantic proximity of the title to the query, and the relevance of the outgoing hyperlinks to the query's subject matter. This is explained by the flexibility of the ontological description and the comparator's ability to identify the subject matter of educational resources even with incomplete or diverse descriptors.

3. Implement a software platform using modern technologies with built-in support for asynchronous programming and lightweight parallelism mechanisms. The developed software platform is built using Kotlin coroutines, architectural patterns, and inter-agent communication via SharedFlow. This ensured efficient operation in both sequential and parallel scenarios with hundreds of active agents. The load on system resources remained moderate. RAM consumption ranged from thirty-four megabytes for sequential mode to nine hundred twenty megabytes for parallel execution with fifty agents.

4. Experimentally evaluate the system's performance and resource consumption in different agent operating modes. A series of experimental tests was conducted, confirming the advantages of parallel and coordinated agent operation. In 60 minutes of operation in parallel mode, 9.3 times more links were processed. In the first mode, the system processed 6,196 links, and in the coordinated parallel mode, 57,535 links. This increase is attributed to the resolution of the looping and URL duplication problem through multi-level filtering and flexible synchronization of agent queues. The achieved balance between search completeness and resource consumption is considered optimal for modern medium-scale educational information systems.

Further development of this research could proceed in several directions. **First**, ontological models could be expanded to account for cultural and linguistic differences, which would improve search accuracy in global educational systems. **Second**, it is necessary to

integrate encryption and access control mechanisms into the system architecture, which will ensure its use in critical educational content management systems. **Third**, it is advisable to develop a deterministic mode of system operation with guarantees of sequence and consistency, using distributed systems theory and formal verification models. **Fourth**, the system's adaptation to heterogeneous network conditions should be investigated, taking into account communication delays and failures, which is particularly relevant for educational systems in regions with limited infrastructure.

Conflict of interest

The authors declare that they have no conflicts of interest, including financial, personal, authorial, or any other nature, that could influence the research or the results published in this article.

Funding

The study was conducted without financial support.

Data availability

The manuscript has no associated data.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technology in the creation of this paper.

References

1. IEEE LTSC (2003), IEEE Standard for Learning Technology – Learning Technology Systems Architecture (LTSA), IEEE Std 1484.1-2003, available at: https://standards.ieee.org/ieee/1484.1/3397/ (last accessed 28.02.2026)
2. IMS Global Learning Consortium (2003), IMS Learning Resource Meta-data Specification v1.3, Final Specification, available at: https://www.1edtech.org/standards/learning-resource-metadata (last accessed 28.02.2026)
3. Wang, X., Li, R., Liu, Y. (2024), "Development and techniques in learner model in adaptive e-learning system: A systematic review", *Computers & Education*, 221, 105114 p. DOI: https://doi.org/10.1016/j.compedu.2024.105184
4. Ghembaza, M. I. E.-K., Moussaoui, A., Mansour, H. (2024), "Ontology-driven and semantic web-enabled framework for intelligent e-learning system", *International Journal of Advanced Trends in Computer Science and Engineering*, 13 (6), pp. 259-265, DOI: https://doi.org/10.30534/ijatcse/2024/071362024
5. Huettemann, S., Harpprecht, T., Scherer, R., et al. (2025), "Designing ontology-based search systems for research information", *International Journal of Information Management*, Vol. 83, 102901 p. DOI: https://doi.org/10.1016/j.ijinfomgt.2025.102901
6. Didden, J. B. H. C., Dang, Q.-V., Adan, I. J. B. F. (2023), "Decentralized learning multi-agent system for online machine shop scheduling problem", *Journal of Manufacturing Systems*, Vol. 67, pp. 338-360. DOI: https://doi.org/10.1016/j.jmsy.2023.02.004
7. Cho, J., Garcia-Molina, H. (2002), "Parallel crawlers", *Proceedings of the 11th International Conference on World Wide Web (WWW'02)*, New York, NY, USA, pp. 124-135. DOI: https://doi.org/10.1145/511446.511464
8. Boldi, P., Codenotti, B., Santini, M., Vigna, S. (2004), "UbiCrawler: A scalable fully distributed web crawler", *Software: Practice and Experience*, 34 (8), pp. 711–726. DOI: https://doi.org/10.1002/spe.587
9. Li, Z. *et al.* (2025) "Embodied Multi-Agent Systems: A Review", *IEEE/CAA Journal of Automatica Sinica*, 12(6), pp. 1095-1116. DOI: https://doi.org/10.1109/jas.2025.125552
10. Karataiev, O., Shubin, I. (2023), "Formal model of multi-agent architecture of a software system based on knowledge interpretation", *Radioelectronic and Computer Systems*, Vol. 4, pp. 53-64, DOI: https://doi.org/10.32620/reks.2023.4.05
11. Carbonaro, A. (2008), "Ontology-based video retrieval in a semantic-based learning environment", *Journal of e-Learning and Knowledge Society*, Vol. 4 (3), pp. 203–212, DOI: https://doi.org/10.20368/1971-8829/290
12. Shubin, I. (2021), "Development of conjunctive decomposition tools", CEUR Workshop Proceedings, 2870, pp. 890-900, available at: https://ceur-ws.org/Vol-2870/paper67.pdf (last accessed 28.02.2026)
13. Talib, M. S., Al-Najar, A. A.-M., Hassan, A. H., Talib, Z. S. (2024), "Kotlin Programming Language: A Comprehensive Overview of Definition, Applications, Advantages, and Limitations", *International Journal of Computer Applications Technology and Research*, Vol. 13 (7). DOI: https://doi.org/10.7753/IJCATR1303.1002

14. Gupta, N. et al. (2025), "Comparative Analysis of Kotlin, Kotlin Multiplatform, Swift, Flutter, and React Native in Cross-Platform App Development", *International Journal of Innovations in Engineering and Science*, Vol. 10(6), pp. 23-27. DOI: https://doi.org/10.46335/ijies.2025.10.6.5
15. Kim, J. (2023), "Derivation of Kotlin Flow API from Observer Pattern", *Journal of KIISE*, 50(9), pp. 821-826. DOI: https://doi.org/10.5626/jok.2023.50.9.821
16. Goel, R. (2025), "Design Patterns For Enterprise Application", *International Journal For Multidisciplinary Research*, 7(2). DOI: https://doi.org/10.36948/ijfmr.2025.v07i02.41268
17. Arun, N. (2025), "Creational Design Patterns – A Comprehensive Review And Their Role In Object-Oriented Design", *International Journal of Core Engineering & Management*, 8(2), pp. 15–34. DOI: https://doi.org/10.5281/zenodo.15486190
18. IEEE LTSC (2020), *IEEE Standard for Learning Object Metadata*, IEEE 1484.12.1-2020, DOI: https://doi.org/10.1109/IEEESTD.2020.9262118
19. Martin, H. and Dick, L. (2025) "Distributed Learning for Lasting Impact", *The Clinical Teacher*, 22(6). DOI: https://doi.org/10.1111/tct.70225

Received (Надійшла) 02.12.2025
Accepted for publication (Прийнята до друку) 14.03.2026
Publication date (Дата публікації) 30.03.2026

Відомості про авторів / About the Authors

Шапиро Олексій Костянтинович – аспірант, Харківський національний університет радіоелектроніки, кафедра програмної інженерії, Харків, Україна;
Oleksii Shapyro – PhD Student, Kharkiv National University of Radio Electronics, Department of Software Engineering, Kharkiv, Ukraine;
e-mail: oleksii.shapyro@nure.ua
ORCID ID: https://orcid.org/0009-0005-3539-266X

Сотник Ігор Сергійович – аспірант, Харківський національний університет радіоелектроніки, кафедра програмної інженерії, Харків, Україна;
Ihor Sotnyk – PhD Student, Kharkiv National University of Radio Electronics, Department of Software Engineering, Kharkiv, Ukraine;
e-mail: ihor.sotnyk@nure.ua
ORCID ID: https://orcid.org/0009-0000-5638-6975

МЕТОД СЕМАНТИЧНОГО ПОШУКУ НАВЧАЛЬНОГО КОНТЕНТУ НА ОСНОВІ МУЛЬТИАГЕНТНИХ ТЕХНОЛОГІЙ

Цифрова трансформація промисловості супроводжується активним впровадженням нових технологій та швидким оновленням виробничих процесів. Значна частина навчальних матеріалів розміщується у різних інформаційних джерелах, включаючи внутрішні корпоративні системи, відкриті освітні платформи та спеціалізовані веб-ресурси. При цьому такі ресурси часто містять дублікати, надлишкову інформацію та різнорідні метадані, що ускладнює оперативний пошук релевантних навчальних матеріалів. **Предметом** дослідження є метод семантичного пошуку навчального контенту у розподіленому інформаційному середовищі з використанням онтологічних моделей знань. **Мета** роботи – дослідження методу семантичного пошуку навчального контенту у розподіленому інформаційному середовищі на основі мультиагентної організації обробки інформаційних ресурсів та використання онтологічних моделей знань. **Завдання** дослідження: дослідити архітектурну модель мультиагентної системи пошуку; розробити алгоритм семантичного відбору на основі методу компараторної ідентифікації та онтологічної моделі; формалізувати предикат оцінки релевантності з урахуванням вагових коефіцієнтів метаданих; розробити програмну мультиагентну платформу; експериментально оцінити продуктивність і ресурсоємність у різних режимах роботи агентів. **Методи** дослідження: метод мультиагентної організації обробки інформаційних ресурсів з неблокуючим обміном повідомленнями; трирівневу дедуплікацію URL-адрес; онтологічне зіставлення термінів та формалізований предикат оцінки релевантності; експериментальне вимірювання тривалості обробки,

ISSN 2522-9818 (print) *Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 1 (35)*
ISSN 2524-2296 (online) *Innovative technologies and scientific solutions for industries. 2026. No. 1 (35)*

кількості опрацьованих посилань та споживання системних ресурсів. **Результати**: запропоновано модель мультиагентної системи з чотирма типами агентів та алгоритм семантичного пошуку з усуненням зациклення і дублювання посилань; реалізовано програмну платформу на базі Kotlin із використанням корутин та асинхронної взаємодії агентів; експериментально встановлено, що запропонована організація обробки забезпечує більшу продуктивності порівняно з послідовним режимом. **Висновки:** поєднання семантичного пошуку та мультиагентної архітектури дозволяє ефективно організувати процес виявлення та обробки навчального контенту у розподіленому середовищі. Запропонований метод забезпечує узгоджену роботу агентів, усунення дублювання посилань та раціональний баланс між повнотою результатів пошуку і використанням обчислювальних ресурсів.

Ключові слова: семантичний пошук; навчальний контент; програмні агенти; мультиагентні системи; онтології; масштабованість; продуктивність; асинхронність; Kotlin.

Бібліографічні описи / Bibliographic descriptions

Шапиро О.К., Сотник І.С. Метод семантичного пошуку навчального контенту на основі мультиагентних технологій. *Сучасний стан наукових досліджень та технологій в промисловості*. 2026. № 1 (35). С. 127–138. DOI: https://doi.org/10.30837/2522-9818.2026.1.127

Shapyro, O., Sotnyk, I. (2026), "A method of semantic search for educational content based on multi-agent technologies", *Innovative Technologies and Scientific Solutions for Industries*, No. 1 (35), P. 127–138. DOI: https://doi.org/10.30837/2522-9818.2026.1.127