ACCSCIENCE PUBLISHING

Artificial Intelligence in Health

ORIGINAL RESEARCH ARTICLE

Explainable solutions from artificial intelligence for health-care support systems

Valeriya Gribova[1,2] **and Elena Shalfeeva**[1,2*]

[1]Laboratory of Intelligent Systems, Institute of Automation and Control Processes, Far Eastern Branch of RAS, Vladivostok, Primorsky Krai, Russia

[2]Department of Software Engineering and Artificial Intelligence, Far Eastern Federal University, Vladivostok, Primorsky Krai, Russia

Abstract

For decades, efforts to standardize medical care have struggled to fundamentally reduce errors and unjustified variations in medical practice, largely due to the influence of the human factor. The formalization of clinical guidelines and computer-assisted interpretation makes it possible to provide decision-support tools to improve health-care quality. They can better influence clinician behavior than narrative guidelines. Medical ontologies and algorithms based on such ontologies allow the interpretation of formalized clinical documents (guidelines). To support health professionals as consultants, systems must provide reliable knowledge and rely on approaches explicitly explaining their recommendations. Integrating software engineering, knowledge engineering, and artificial intelligence advancements can provide health-care professionals with computer-interpretable clinical guidelines. These should be decision-support complexes combined under a common terminological framework capable of understanding patient health documents. The research focuses on an emerging concept of manufacturing systems working with digital clinical guidelines. The paper presents an architectural principle, a new technology for creating viable clinical decision support systems. It presents a development environment for constructing and controlling the system's improvements. The main contributions of the study include the automation of multiple physician tasks by filling a single structured "medical history," integration of formalized knowledge from clinical guidelines and other reliable sources to satisfy both the relevance of the methods used and personalization to patient, transparency of all applicable knowledge, explainability of advice based on the essence of the knowledge and linked to the source, and the integrability of decision-support complexes with neural network services, capable of inputting data from a structured medical history.

Keywords: Explanatory decision support system; Knowledge-enabled system; Interpretable clinical guideline; Knowledge ontology; Viable system

***Corresponding author:**
Elena Shalfeeva
(shalf@iacp.dvo.ru)

Citation: Gribova V, Shalfeeva E. Explainable solutions from artificial intelligence for health-care support systems. *Artif Intell Health*. 2025;2(3):138-153.
doi: 10.36922/aih.5736

Received: October 31, 2024

Revised: March 1, 2025

Accepted: April 25, 2025

Published online: May 8, 2025

Copyright: © 2025 Author(s). This is an Open-Access article distributed under the terms of the Creative Commons Attribution License, permitting distribution, and reproduction in any medium, provided the original work is properly cited.

Publisher's Note: AccScience Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

One way to improve health-care quality, reduce unwarranted variation in practice, and lower health-care costs is through e-consultants. These tools are intelligent services and computer-interpreted clinical recommendations integrated into the workflow of medical care.[1,2] To support health professionals as consultants, such tools (e-consultants) must be based on solid, substantial knowledge. Machine learning algorithms, including deep learning techniques, provide consultations by being trained on massive datasets. However, the specificity of medical information means no datasets are adequate for solving the problem. An exception exists in simplified prediction or risk assessment tasks, where datasets consist of thousands of uniform tuples of predictors. For pre-diagnostic tasks, a large number of symptom checkers are provided.

This study showed that for 10 randomly selected clinical cases with a reliable diagnosis (five from PubMed publications and five from the city hospital) and the five most promising symptom checkers, the correct diagnosis was in the top five in 40% of cases, and became the most likely in only two cases. This confirmed that artificial intelligence (AI) services trained on data from one institution work well on simple cases or data from the same place. Fortunately, methods based on explicitly presented and manageable information are being developed.

In medicine, collections of text-based clinical guidelines for physicians, doctors, and clinicians are constantly being expanded and modified. Formalized clinical guidelines, which a program can interpret, make it possible to provide decision support tools with better chances of influencing clinician behavior than narrative guidelines. The methodology of creating interpretable clinical guidelines is rightly associated with knowledge-based systems,[3] a software system with a particularly important information component, the knowledge base. This additional architectural component is created with the participation of specialists and experts. The methods of automated recognition of such medical texts are still being developed and improved. Experts input these texts into electronic databases in a conscious and meaningful way. However, this process requires appropriate formalization tools.

The integration of explicitly written knowledge with machine learning allows the vast amount of human knowledge and the capabilities of machine learning to be used to achieve previously unattainable performance. Integration can increase the reliability and robustness of machine learning, facilitate interaction between humans and machine learning systems, and make system decisions understandable to humans.[4] Knowledge-enabled systems (KES) are applicable to mediate between machine learning algorithms and human users.[5]

To support health professionals as consultants, not only is high-quality knowledge important in such systems, but these systems must also rely on approaches that explicitly explain their recommendations. However, machine learning algorithms have not been able to generate an explanation of their decisions.

Time-consuming and complex systems are designed for long-term operation and use. However, the terms of use change over time, as do the user's requirements for functionality, interface, and even expertise. In medicine, clinical guidelines, which reflect clinical knowledge and practice, are subject to change. We refer to the software system for physicians that can update its knowledge as medicine evolves as a "viable" system.

Integrating advances in AI, software engineering, and knowledge engineering can offer clinicians comprehensive, medically intelligent systems that can provide credible answers, understand patients' histories and medical records, and evolve with the knowledge of specialists.

This paper aims to present an architectural principle and a new technology for creating explanatory clinical decision support systems (ExClDSS) and a tooling environment to ensure their evolution.

The ontology-oriented production method is proposed, the contribution of which can be expressed in five aspects:
(i) Independent (but coordinated) creation and support of each type of architectural component: knowledge bases, clinical dictionaries, software reasoners with explanations (ontological interpreters), and user interface. A single ontology ensures their consistency
(ii) Cognitive scientists are engaged in the ontology of knowledge, laying the foundation for forming various types of cause-and-effect, spatial, and temporal relationships of medical concepts. Therefore, the knowledge base is close to the real knowledge that doctors are taught at universities
(iii) Explanation is formed based on the applied knowledge base (in the format of the "ontology of explanation" approved by doctors)
(iv) Evaluation of the created knowledge base with various methods and tools. We evaluate the formal completeness and integrity of the knowledge base and conduct a monitoring assessment of the current state and correctness of the replenished collection of cases (precedents), ensuring a monotonous improvement of the knowledge base
(v) Software reasoners (together with a graphical user interface [GUI]) do not depend on knowledge and reference books. Therefore, they are repeatedly used to lay out many systems with knowledge bases.

The research's novelty is as follows. Unlike most existing methods of knowledge-based product constructing – that assume either the presence of a sufficient set of data to extract the necessary knowledge from them or the presence of an expert who can formulate an adequate set of knowledge – our research attempts to address the problem across the compatibility and complementarity of these paths, rather than interchangeability.

Thus, the research scope is the manufacturing technology of continuously developing trusted systems to support difficult clinical decisions.

2. Materials and methods

2.1. The concepts of explanation in a decision support system

Medical knowledge results from a person's theoretical and practical activities, reflecting the accumulation of previous experience. Knowledge is the regularities of the domain (relationships and rules) necessary to solve problems based on new data (facts). Clinicians expect recommendations or hypotheses from e-consultants with convincing explanations and systems with solid, reliable knowledge. The production of detailed explanations is an important element of decision support systems in general and computer-interpreted clinical guidelines in particular.

The generation of interpretable medical knowledge requires additional specialized mechanisms. The universal representation of knowledge in the form of rules processed by a single "inference engine" has a limitation in medicine: since the number of rules is measured in thousands, it becomes virtually impossible for experts to review, verify, refine, and correct them.

Semantic models of the medical knowledge domain are required to formulate clinical recommendations as components for ClDSS. These models must reflect the logic that physicians use when appealing to their sentences and fragments. The description of a set of concepts, relationships, and constraints used by specialists when solving problems and transferring knowledge between specialists is what we refer to as a knowledge ontology.

The hierarchies of classes of entities and their binary relations are already a step toward the declarative representation of a part of knowledge, but they cannot cover the most important clinical connections (and we do not consider them to be a full ontological language). Knowledge ontology is a part of the semantic description of the medical domain, and the other part is the semantic representation of data and documents. In domain ontology, all the concepts of specialists, relations of concepts, and restrictions on interpreting their meanings are defined. Together with them, the required types of statements are specified about factors such as necessary conditions, grouping, and cause-effect relationships. This ontological approach makes it possible to develop systems that offer detailed explanations of their recommendations for knowledge bases. Various intelligent components of the system work with a single semantic description of the source data, reading the elements they require.

Each intelligent component (algorithm) usually interprets its specific blocks of knowledge in clinical guidelines: diagnostic rules or treatment recommendations. The engineering of such interpretable knowledge as independent architectural components (knowledge about diagnosis, risks, and treatment) is done based on their ontologies.[3] Existing methods and technologies[6,7] make creating, testing, and deploying knowledge bases and knowledge-based systems, or KES, possible.

The most detailed part of the explanation is formed based on ontological knowledge (selected fragments in the analysis) and, in some cases, partially copies its cause-effect and structural connections. Forming the explanation of the results in terms that are understandable to the doctor and that correspond to his logic allows the application of a full-fledged medical ontology.

The explicit representation of all concepts and their relations makes it possible to input, shape, store data (and knowledge), and demonstrate results that can be generated automatically. Such ontological algorithms could be used in conjunction with machine learning approaches, either as a source of ground truth or as a thematic layer that could be used to promote interaction or improve explainability. When other AI modules are connected to generate advice, recommendations, or solutions, they should work with a single semantic description of the source data to ensure compatibility in explaining all results.

Using medical ontology allows for creating a GUI in terms that are understandable to the doctor and also, as a rule, forms a dialog script that corresponds to the logic of explaining the results. To support the doctor at all stages, it is important to combine AI modules as components of medical knowledge, obtained in different ways, into a united ExClDSS, all based on a single terminological framework.

Existing explainable AI methodologies use large language models, requiring large capacities and rechecking to mitigate potential hallucinations.[8,9] The path used in this study is an improvement (based on ontology) of classical explanation generation: during the decision process, the reasoner records its arguments in the knowledge base.

2.2. Decision support system maintenance and viability concepts

The developer of a clinical system has the task of designing and implementing the mechanisms that will provide further maintenance to meet the changing knowledge and service conditions in the created software product. The same mechanisms help solve problems in implementing "continuous delivery," a process in which software is always kept relevant.

The application of typical architectural solutions, declarative representation of components, and separation of competencies between developers of components of different types are all used to create maintainable decision support systems. Information technology managers struggle to scale AI projects because they lack the tools to create and manage a "production-grade AI pipeline."[10]

With the advent of complex software systems, the problem of their long-term maintenance has become more and more critical. Maintenance is the possibility of adaptation (to hardware and system software, to new types of human-machine interfaces and users) and extensibility (at the request of users). Modification of software systems is due to changes in operating conditions, user requirements, and subject area. In the operation of applied systems to support professional activity, there is a need to add new user-defined functions (and adapt to new devices or user interface changes). The average maintenance cost of the software system life cycle is about 50%,[11] but according to some reports, it can reach 80 – 90%.[12]

In addition to maintenance, viability has become a modern, useful property of software systems. It is defined as sustainability in a changing environment (maintaining usefulness and operability),[13] and the ability to evolve, as the ability to adapt with the least possible cost to requirements' variability, maintaining architectural integrity.[14] We will specify "the viability" as software system resilience to some functioning environment changes (the maintenance of working capacity) and the ability to develop over the "life" (evolvability).

In the case of applied decision support systems in intelligent tasks such as diagnosis, planning, and forecasting, the situation is different. Here, knowledge variability and the emergence of new solutions, such as creating new diagnostic methods and identifying new influencing factors, are expected, rather than just the extension of user functions. Therefore, the approach to maintaining ClDSS should not be similar to maintaining application software systems.

Many well-known tasks in diagnosis, treatment, and prognosis in general and medicine, in particular, are quite stable. Algorithms for solving them are described and can be qualitatively programmed once for long-term use. However, this is not the case for medical knowledge and clinical guidelines. They cannot be "sewn" into programs because they are regularly updated in this dynamic field.

Clinical decision support systems should have one part (knowledge) that is constantly evolving and another part that can read and understand it, i.e., be an interpreter. Medical knowledge, such as clinical guidelines, is an evolving part of ExClDSS. ExClDSS are expected to remain useful and effective in an environment of changing knowledge. Under conditions of variability in clinical knowledge, the viability of the medical system is manifested in its ability to adapt and update in response to new information and evolving practices.

In medical knowledge, the influence of factors and events on the patient's state, their change over time, individual characteristics, and some of their processes on others is important. The development (evolution) of such complex knowledge bases is the main "challenge" of modern "conditions" with (Ex)ClDSSs. "The ability to adapt under a change in the set of facts and knowledge" is one of the aspects of intelligence.[15(p.5)] For medicine related to solving intellectual problems, this implies the evolution of knowledge. The ontological approach to knowledge and programming for working with them was sufficient.

3. Clinical decision support systems as software systems that apply understandable knowledge

To ensure that doctors trust ClDSSs, their developers need to demonstrate correctness (sometimes accuracy) on subsets of precedents (cases from practice), implement the ability to explain the proposed solution or hypothesis (the explanation must be understandable, consistent with formalized knowledge), have a mechanism for permanent improvement of the knowledge base that does not worsen its correctness, apply procedures for regular evaluation of stored clinical knowledge, and provide the opportunity for specialists to read and evaluate the included knowledge.

Knowledge must be formed considering standardized clinical guidelines and under domain experts' control. One method is to use trained text recognizers. Knowledge can sometimes be created by experts themselves (possibly with knowledge engineers and cognitive scientists). In this case, experts fully participate in the development and maintenance process with programmers and designers. This requires knowledge bases to be presented in a form understandable to medical domain experts. When knowledge is isolated and framed in independent architectural components and knowledge bases, the system using them becomes a KES.

Several AI, mathematical modeling, and machine learning methods for solving practical problems provide medical services based on hidden knowledge.[16] In medicine, these are most often the tasks of risk assessment (for example, risk of hypokalemia in patients with arterial hypertension) and predicting complications.[17,18] Almost all services based on machine learning provide versions of a preliminary diagnosis without considering the dynamics of the patient's observations.[19,20] For such tasks (preliminary diagnostics, risk assessment, or forecasting), as a rule, an intelligent service becomes inaccurate if it was "trained" on the data of one institution and it tries to operate in other circumstances.

There are tasks for which no one has yet accumulated adequate training material. In medicine, these are corrections to disease treatment and differential diagnosis. For this, intelligent services consultants trained on text corpora (and GPT helpers) within the idea of hybrid services to support the doctor's work may be used.

3.1. The influence of the ontological model on the properties of system components

For ClDSS to correctly formulate advice or results for solving medical problems (risk, diagnostics, treatment, and prognosis), it needs to operate with concepts that specialists use. For example, for knowledge of the task of monitoring the recovery process, one of the most common types of sentences (statements) is:

<process type$_k$, set-of {(period$_{ik}$ + interval$_{ki}$), set-of {characteristic$_j$, characteristic values range$_{ijk}$}}>.

For the diagnosis of diseases, statements about the relationships are needed:

<diagnosis$_k$, process' existence necessary condition$_k$, set-of {factor$_{km}$}>;

<diagnosis$_j$, set-of {symptom complex$_{jk}$ | variant$_{jk}$ of disease course}, [necessary condition$_j$]>;

<symptom complex$_k$, set-of {feature$_j$, range$_{kj}$ of values of feature}>;

<symptom complex$_k$, set-of {sign$_{jk}$, {period$_{ik}$, duration of period$_i$, range$_{ijk}$ of values of sign$_j$ in period$_i$}}>;

<variant$_n$ of disease course, set-of {(period$_{in}$ + interval$_{in}$), set-of {observation$_k$, set-of {observation element$_{jk}$, range of values$_{ijkn}$>}}>;

<necessary condition$_k$, set-of {factor$_{km}$}>.

All such concepts and relationships are explicitly written and "available" to algorithms when they are developed based on ontologies and access to explicit knowledge resources. The medical ontology was created by experts and knowledge engineers. The factors without which a disease does not begin can be the events or properties of an organism; they can also determine one of the options for the development of the disease. For the treatment of diseases, the statements about the method to eliminate the cause of the disease:

<diagnosis$_k$, event$_u$, set-of {(observation$_{jk}$, new-range$_{iu}$ of observation values)}, delay$_{ku}$>;

The statements about the impact on an organism for recovery start:

<process variant$_{kn}$, (period$_{in}$, interval$_{in}$), treatment event$_u$, set-of {observation$_{jkn}$, value range$_{jkn}$ (period$_{(i+1)n}$, interval$_{(i+1)n}$)>}}>;

The statements about acting on a symptom to alleviate it:

<process variant$_{kn}$, (period$_{in}$, interval$_{in}$), observation$_{jk}$, values$_{jkn}$ range, (event$_u$, delay$_{kj}$), values new range$_{iu}$>.

The domain ontology represents all types of statements as a structural language (template) for introducing or describing knowledge. The knowledge base explicitly contains sets of statements of the corresponding type sufficient for this profession.

The traditional architecture of KES is the knowledge base + fact base + intellectual problem solver + intellectual GUI.[15Finn2004] Knowledge bases are generated manually or inductively, including training samples from archives and databases; this process involves inductive generalizations in machine learning.[16] Bayesian classifiers, clustering algorithms, and reinforcement learning[21,22] are sometimes involved in this process.

The approach to creating systems with transparent knowledge bases is based on an architecture expanded by a new component: ontology + ontological knowledge base + ontological fact base + ontological interpretator (problem solver) + intelligent graphical interface. It can include databases with reference information, operational information, and work with files.

Often in one domain, a set of interrelated tasks is solved; examples of related tasks are diagnosis, treatment, and prognosis. To solve all tasks, one formal domain ontology can be created, but it is more convenient to map a separate ontological resource to separate tasks solved in the domain. A set of formal ontologies for related tasks may be required when designing applied systems with knowledge bases, which is followed by the creation of a set of ontological knowledge bases (for each task).

3.2. The influence of ontology on the integrability of various components within a single architecture

To combine various achievements of knowledge engineering and AI in complex medical intelligent trustworthy systems, it is reasonable to choose the document "patient's medical record" as an integration point. The document "medical history of any patient" contains a structured set of facts observed or objectively measured in the considered situation (medical case) regarding which the problem is solved.

All results and decisions are recorded in the same document (medical history), regardless of the method in which they were collected. The place in the document structure must be strictly connected with the essence of the result (diagnosis in one place and prognosis in another). Such a document structure is part of the domain ontology. Thus, it can be asserted that ontology ensures the integrability of various components.

Often, it is necessary to add a pre-existing software service with hidden knowledge (trained model) to the system to solve a specific problem. Typically, this task falls into one of three categories: risk assessment, prediction, or recognition of a class of pathologies. A structured description of the service is sought, which includes the following elements: (i) name and author of the method, (ii) essence of the result, (iii) vector of initial data, (iv) conditions of applicability (entering values in limited ranges), (v) manner of launching the service, and (vi) if the expected response of the service is numeric, then the description should also include the interpretation of the result.

For the mutual exchange of data and results with software services (with hidden knowledge), a single semantic template is used: <name of method, author, essence of the result, vector of initial data, [conditions of applicability], description of result interpretation, launch method or full address of microservice>. For example, for a software service for assessing the risk of developing a disease, the description of the interpretation of the result = a set of pairs <threat level value, range of calculated values>. Adding such a semantic template (with a description of the interpretation of the result) to the medicine ontology ensures the explainability of connected, intelligent services. The vector of initial data (from the semantic template) should be formed only with the help of the terms of the "medicine" thesaurus. The "medicine" thesaurus (a dictionary of terms for observing a patient and studying the patient's body) is traditionally considered part of the ontology of this domain area. Hence, the ontology is a structural basis of both tools for experts and users (editing tools) and for software components of KESs.

4. The viability model of clinical knowledge-enabled decision support systems

An ontology, as a structural basis for viewing and editing knowledge bases, provides a basis with a declarative property. However, for the knowledge base to be adaptable, the user interface of the knowledge editing tool must meet the requirements and expectations of domain experts.

The main challenge in medical systems manufacturing is ensuring that the knowledge reflects current knowledge (e.g., clinical guidelines) and continuous improvement. Continuous improvement of the knowledge base allows it to become a reliable source (repository) of expert knowledge, hoping to create a "reference" knowledge base. Its quality will determine the success of the use of this knowledge.

The relevance of a knowledge base is achieved through three main ways:[23-25] interactive change of the knowledge base, usage of machine learning methods (tools of inductively generating knowledge from selected precedents and tools of knowledge discovery from "big data"), or a combination thereof. The "success" of adaptability depends on several conditions and principles.

4.1. Architectural properties of clinical systems enabled with declarative knowledge

This intelligent software system class, which explains decisions, requires specialized development and maintenance tools. The key principle is the special role of the knowledge ontology (as a model of professional concept relations). Its formalized representation, separated from the professional knowledge itself, allows for the independent development of each ontological component, relying on its integrability. The medical ontology makes it possible to create a GUI that is understandable to the doctor and, as a rule, to create a dialogue script corresponding to the logic of explaining the results.

The interpretation of knowledge consists of choosing each hypothesis and transitioning from it (along the chain of connections) to the expected values areas of observations for subsequent comparison with facts, as well as constructing an explanation with the collected arguments. The "structural" complexity of the ontological interpretation algorithm is determined by the number and the length of the chains of cause-and-effect relationships in the statements, the degree of fuzziness prescribed in cause-and-effect relationships and statements, and the structure of observation, description, or conditions for a decision.

To develop interpreters (task solvers), coding tools for new software units or their new versions, tools for cataloging units for reuse, and tools for integrating reusable units and new ones into new solvers or their new versions are needed. Solvers built according to a given ontology (for problems of diagnosis, prognosis, etc.) must be reusable reasoning engines of medical services. A version of clinical knowledge is their input parameter. Therefore, regular updating of the knowledge base does not require changes in other components of the KES.

A connector module is used to connect an external microservice to the medical service, which solves an additional task for the same data (documents about the patient). It consists of standard tasks such as reading the list of names of the required data in the declaration (specification) of this external service, finding the values of this data in the input document, composing a "PUT" request with the specified uniform resource locator, and sending it. If the microservice is not interactive, it is necessary to wait for a response, select fragments (specified in the declaration) for explanation, and add them to the provided substructure of the final explanation.

4.2. Ontological approach to support and develop KESs

The dependency of all KES components on a domain ontology supports the viability properties, which include the replaceability of components for their improved versions, admissibility of the improvement of the decision method, permissibility of changing or adding functions (for example, the formation of additional results), adaptability of the user interface due to changes in the input data, and permissibility of expanding the ontology (adding concepts and relationships).

As a result, the structure of the KES and its components does not require changes due to current maintenance and sustainable development.[26] As mentioned above, this class of software systems (ClDSSs) requires specialized maintenance tools because clinical guidelines (and other medical knowledge) are constantly evolving. Due to the importance of evolving knowledge bases, only ClDSSs integrated with a knowledge base management system should be considered.

A toolkit for building application systems with declarative (interpretable) knowledge is based on a domain and problem ontology. If a separate formal ontology is created for each task (diagnosis, treatment, prognosis, etc.), it is easier to develop tools to ensure the quality of homogeneous, localized knowledge. The tools for developing and verifying knowledge bases are desirable to be integrated into the architecture of the decision support system (to be a part of the integrated architecture of the decision support system). Thus, a maintenance environment has to provide knowledge base editors, tools for assessing knowledge bases by archives of etalons (solved problems), tools for checking and evaluating the quality of knowledge bases, and tools for the inductive formation of knowledge base fragments.

If the pre-existing solver is in accordance with the problem statement and "building up" additional functionality is not supposed to be used, then coding tools for program units may be unnecessary.

4.3. Testing the quality of the system with a knowledge base

When testing the quality of ExClDSS work, "control sets of clinical cases" should be used. This is a carefully selected set of documented medical histories containing the correct solution and facts sufficient to develop the correct solution. Based on the "control set of clinical cases," a "control set of test cases" (CSTS) should be prepared by clearing out information that is not important for the target task. This, in particular, depends on the task being tested (treatment, diagnostics, prognosis, or prevention).

We believe it is important to use metrics to assess the quality of the ExClDSS components (such as sensitivity, specificity, positive predictive value, and precision) and metrics to evaluate the quality of ExClDSS performance in supporting the solution of specific clinical problems. These metrics are defined to the CSTS. For example, for the task of forming hypotheses about a diagnosis, we use: CorrectnessEstimation (number of tests-with-a-finding/card of CSTS) and AccuracyEstimation (number of tests-with-a-hit/card of CSTS), where a test-with-a-finding is a test clinical case of a certain disease, for which the ExClDSS generated many hypotheses during testing, among which was this disease. The test-with-a-hit is a test clinical case for which the ExClDSS generated its disease as the only hypothesis. The card of CSTS is the power of the set of all prepared test clinical cases.

These quantitative metrics are associated with clinician satisfaction. For almost every clinical task (except for differential diagnostics), the metrics for assessing the accuracy and correctness should be determined separately for each of the diseases, knowledge of which is "embedded" in the ExClDSS. Each of these tasks has its own requirement for the set of patient data in the test clinical cases used, and they are not the same for acute, slowly progressive, chronic, and hereditary diseases.

5. Manufacturing environment for viable clinical decision support systems

As noted above, this class of clinical software systems in the construction of <ontology + set of ontological knowledge bases, set of ontological interpreters, set of user interface components, sets of facts> requires specialized development and maintenance tools where coding tools for new software units or their new versions may turn out to be unnecessary. The authors are aware of ontological portals but not of development environments focused on quality assurance and the development of ontological components for such systems.

5.1. Implementing KESs in manufacturing environments

An example of an environment for producing clinical KESs is thematic medical ontological portals, and an example of a toolkit for modern KES production and maintenance environments is the Intelligent Adaptive Clinical Platform as a Service (IACPaaS) cloud platform. The creation of an ontological portal starts with the creation of an ontology as a semantically structured basis for the creation and processing of ontological information resources (Figure 1).

An ontology (as a template or "meta-information") defines a semantic model, structure, rules for the formation of information resources, limitations of its interpretation, or processing rules.[27] Usually, cognitive scientists (users of the IACPaaS platform) form such semantic structures and rules for the community of experts and specialists (users of the IACPaaS portals) (Figure 1). The tool for ontology formation is the IACPaaS meta-information-editor "ontology editor."

Creating an ontology is a creative process requiring extensive analytical work and a systematic domain analysis to identify common patterns in forming knowledge, structure, and integrity constraints. The ontology is generally not changeable throughout the life cycle of the KES. The separation of an ontology from a knowledge base (and a set of facts) leads to the ability to interpret them with a specialized ontology-based algorithm. The algorithm (ontological reasoner) searches for or refutes hypotheses by traversing the (declarative) knowledge base. It "sorts" the knowledge base statements of each type related to the hypothesis, comparing these sentences of input information (patient's document).

The medicine ontology was carried out by cognitive scientists (knowledge engineers) together with experts. It "covers" several classes of tasks (for example, diagnosis of diseases regardless of their etiology[28]). The knowledge base is an ever-changing component and should be formed based on the created ontology. KES is a special case of applied software services on the IACPaaS platform (IACPaaS services). They need the information resources (of the portal), such as knowledge bases. The set of IACPaaS-portal tools for the formation of KES's information components are the IACPaaS-editor of knowledge base, generated in terms of ontology with a self-adaptive user interface (Figure 2), and the IACPaaS data editor (with self-adaptive user interface) (Figure 3).

The regular generator of information editors makes knowledge base editors available and constantly operating in the production environment (Figure 2).

In addition, the IACPaaS toolkit to form software solvers (in addition to coding tools for software units) contains (Figure 4) (i) an IACPaaS meta-information editor to explain the resulting structure, (ii) a "master" of the formation of declarative parts of software units and their blocks-reasoners, which conduct reasoning on ontological information, (iii) a generator of code blanks (according to declarative parts) for new IACPaaS software units, (iv) a solver constructor from GUI (or "root"-unit or IACPaaS-agent), a software unit (being represented by its declarative parts), including connector modules, and (v) tools for testing IACPaaS agents and preparing them for reuse.

5.2. Creating clinical KESs using Intelligent Adaptive Clinical Platform as a Service environment tools

The KES design technology in the proposed environment provides for a sequence of activities.

(i) Find pre-existing knowledge ontologies whose concepts and relationships are sufficient for the tasks and data ontology. If the IACPaaS platform does not already have an ontology for the problem under

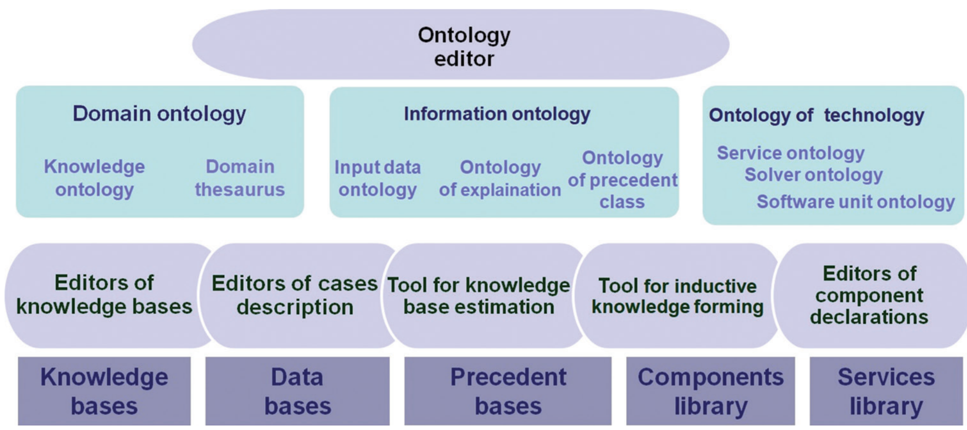

Figure 1. Basic components of the environment for developing basic components of a knowledge-enabled system

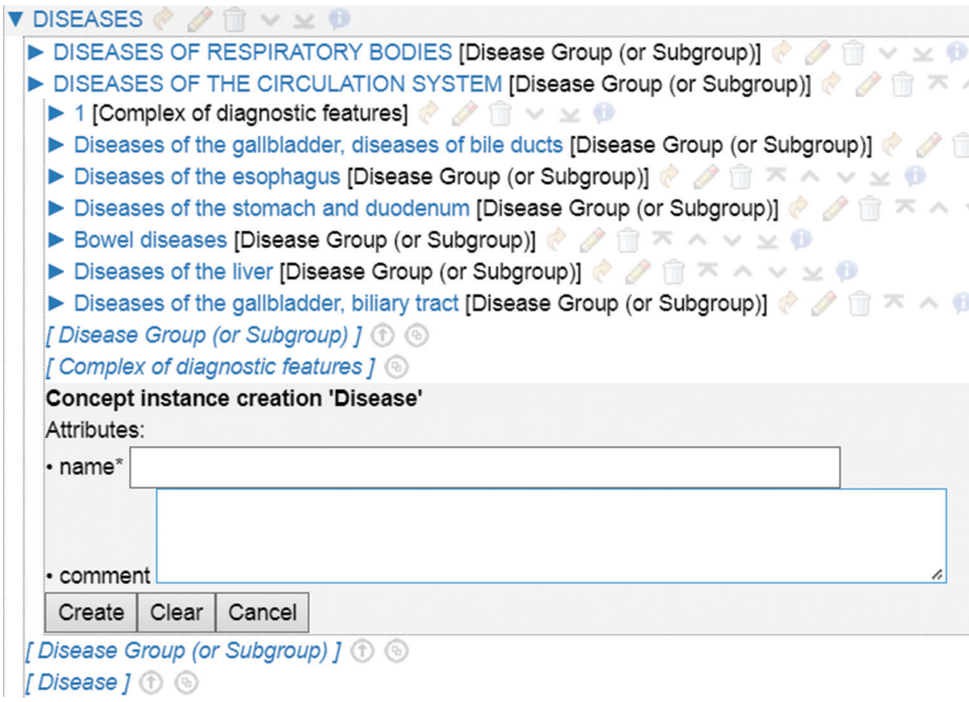

Figure 2. Development process by the Intelligent Adaptive Clinical Platform as a Service editor of the knowledge base

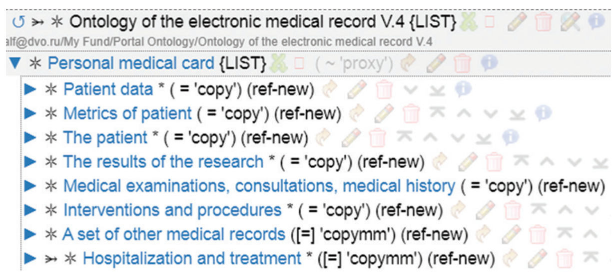

Figure 3. The example of the Intelligent Adaptive Clinical Platform as a Service medical document ontology

consideration, e.g., for risk assessment or prevention, the knowledge engineer uses the ontology editor to describe a set of relevant concepts, relationships, and constraints (creating a knowledge ontology for the problem or task). Often, the engineer will work with the expert to establish a set of decision rules (ontological agreements). A new task may require an extension of the thesaurus

(ii) Formation of the knowledge base by medical experts. The IACPaaS knowledge base editor (for example, based on the extended knowledge ontology) may be required. It is possible to formulate clinical guidelines without obstacles. Using the tools for translating text documents into a given structure is advisable. As a rule, creating the knowledge base is a collective process

(iii) If the creation of a solver is required, then the designer uses the ontology editor to create an ontology to explain the results. Using software engineering techniques, designers and programmers then declare and implement solver components (agents). Notably, creating a knowledge base and creating a problem solver (particularly assembling program blocks from the library) are parallel processes. The ontological structure ensures the compatibility of the components within the portals created.

5.3. Ensuring clinical KESs' viability

The properties of KESs related to viability are provided by the development environments implemented on the IACPaaS platform. Consider the example of the knowledge updatability property and support for knowledge updating. When an update to the knowledge base is required in connection with the acquisition of new knowledge (statements), it is possible to modify it manually. This development (production) environment has knowledge base editors.

Then, it is important to evaluate the consistency with the available facts. It must check the non-decrease of a set of correctly solved tasks when replacing the version of the knowledge base (according to the importance of monotonous improvement of knowledge bases[29]). The procedure for checking the non-deterioration of knowledge is as follows: from each reference task (precedent), enter input conditions into the solver integrated with the new version of the knowledge base, and obtain the explanation

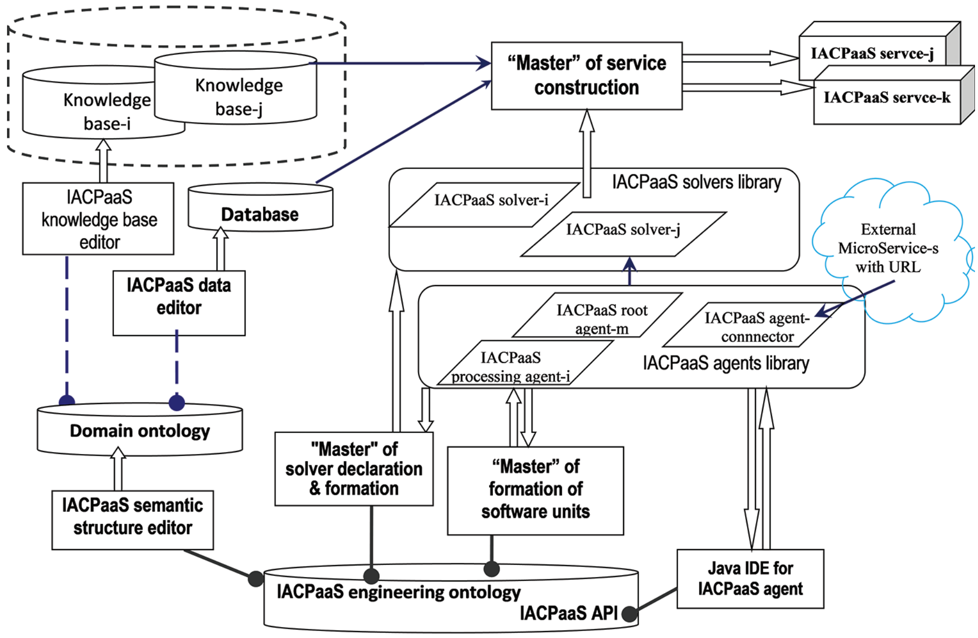

Figure 4. The model of the KESs manufacturing environment
Abbreviations: API: Application programming interface; IACPaaS: Intelligent Adaptive Clinical Platform as a Service; URL: Uniform resource locator.

to compare it with the result, fixed in this precedent. It is advisable to develop a tool for checking the quality of the knowledge base together with a solver since they have many common software blocks.

If it is necessary to update the knowledge base in connection with obtaining precedents that do not correspond to the knowledge (if the precedent contains the correct result of solving the problem, which does not correspond to the result obtained with the help of KES), then it is effective to form a new version of the knowledge base automatically, based on the methods of inductive formation of fragments of the knowledge base.[29] An example of such an update of knowledge "from practice" occurs when, after a certain period, the correct result of diagnosis or treatment becomes available from a medical institution. This outcome is then compared with the result from KES: are they contradictory? In such cases, evaluating the consistency of the updated KES's work result with existing precedents is preferable.

To implement the process of monotonous improvement of knowledge bases, the following means are required: (i) inductive knowledge creation tools for each intellectual problem solved (diagnosis, planning, forecasting, etc.), (ii) tools to support the choice of precedents (correctly solved problems in a statement), and (iii) tools for verifying the correctness (quality) of the new version of the knowledge base (the same as described above). The "knowledge updatability" property depends on the availability and performance of the above tools; other tools are added to the development environment (as a framework) one after another.

We compared the process of building a complex of interconnected evolving knowledge bases to make medical systems using the Protégé[6] and IACPaaS[30] tools. We created classes of diseases, symptoms, and drugs on the web, in Protégé. Next, we had to associate diseases with symptoms using the object property mechanism (some acute diseases required a dynamic description of the clinical picture). However, these mechanisms did not describe knowledge in a way that doctors would need and understand, and explaining the dynamics of disease development was particularly difficult. It should be noted that the Protege tools are incomprehensible and difficult for doctors; they are intended for knowledge engineers (although describing the dynamics proved difficult for them). Similar difficulties were encountered when trying to describe treatment protocols, taking into account the specifics of drug use and patient characteristics. Protégé's mechanisms did not allow the patient's history to be formed as a single document. In contrast, the advantages of IACPaaS in addressing this limitation have been demonstrated.

5.4. Implementing a clinical knowledge-based decision support system

The formation of a Medical Portal, Med-IACPaaS (https://iacpaas.dvo.ru/), began with the development of medical ontologies and editing tools. Previously formed by experts and knowledge engineers, the medical ontology has been formalized as a hierarchical semantic network. It includes a glossary of terms (more than 25,000) for describing the patient's anamnesis, current state, complaints, objective results, laboratory, and instrumental research. The glossary contains commonly used and specific terms, such as the basis of symptoms of cardiological pathologies and neurological terminology. Additionally, the ontology also includes an ontology of medical diagnostics (about 70 concepts and 100 relations between concepts) as one of the knowledge ontologies, the ontology of knowledge about the nomenclature and effects of medicines on the human body with various impaired functioning (about 80 concepts), the ontology of treatment regimens of diseases (about 80 concepts), and a knowledge base (for several nosology groups) formed based on the ontology, the ontology of medical case records history (more than 120 concepts) nodes of the intended for describing information about characteristics of an organism, facts, events, and observations in patients, and the formats of structured reports with analysis and explaining hypotheses on decision on the base of knowledge.

In terms of such networks, experts then began to create knowledge bases without the participation of knowledge engineers. To date, experts and doctors have created and maintained clinical information guides as knowledge bases on a wide range of nosologies (Figure 5). The diagnosis knowledge base, formed in terms of the ontology of medical diagnostics, currently includes more than 250,000 concepts describing the diagnosis of 35 diseases from 10 groups. The knowledge base formed in terms of the treatment ontology currently contains more than 90,000 concepts.

The IACPaaS clinical services were created on various nosologies: viral, diseases of the oral cavity, salivary glands, and jaws, diseases of the gallbladder, biliary tract, and pancreas, bowel diseases, hemorrhagic fevers, coronary

Figure 5. The fragment of the description of some diseases in the diagnosis knowledge base of the Medicine Intelligent Adaptive Clinical Platform

heart diseases, diseases characterized by increased blood pressure, and chronic rheumatic heart disease. Usually, experts form a clinical picture of diseases with dozens of dynamic symptoms. The knowledge about the diagnostic signs of biliary tract disease was built inductively, based on the data set of their surgical department. In addition, from the knowledge built by the linguistic processor Ontosminer (http://ontosminer.opkrt.ru/) based on the analysis of millions of documents from the free resource PubMed, some clinical guidelines were selected for which it was possible to carry out validation based on real case histories. Specialists in mucopolysaccharidoses manually created them, tested them on real examples of patients from different countries, and refined the knowledge.

Developing knowledge bases and intelligent software components for medical services was carried out in parallel. Such services are intended to help medical teams and institutions support the solution of the problems of their intellectual activity, providing a "third opinion" through cloud means. Their hypothesis explanations rely on formalized knowledge (Figure 6).

The labor costs are fully justified because each solver-interpreter works for more than one profile, and each nosological base is used to solve several intellectual problems (diagnosis, treatment, and prognosis). The "cloud" implementation of ClDSS allows the monitoring of relevance and evolving a single clinical guidelines base for several profiles and classes of tasks. The general base accumulates the experience of several professional communities and teams in addition to universal knowledge.

6. Comparing different approaches to the automation of medical activities

Today, specialists from different medical teams are using medical IACPaaS services to test the capabilities of AI in solving various problems for formalized health case histories. For general practitioners, differential diagnosis services are being created, and for gastroenterologists, a complex for diagnosis, treatment, and prognosis of recovery. For cardiologists, the complexes are made so that the risk assessment is carried out in two ways: based on ML and formalized knowledge. In comparison, advice on diagnostics is given in two other ways: based on knowledge and formalized precedents.

Medical software assistants (solving different tasks) are built using IACPaaS tools and components of the IACPaaS medical portal. All assembled systems work based on a single terminological base of symptoms and facts (more than 20 thousand) with synonyms.

The overall knowledge base is large and maintained by multiple experts. Updates are carried out according to procedures that keep the cloud services running. For example, knowledge on some digestive system diseases has been expanded, information on the regional manifestations of fever caused by rodents has been clarified, and new, unique knowledge on metabolic disorders of glycosaminoglycans, mucolipids, and gangliosides has been added.

Knowledge base editing and verification tools are used to update current diagnostic and therapeutic knowledge.

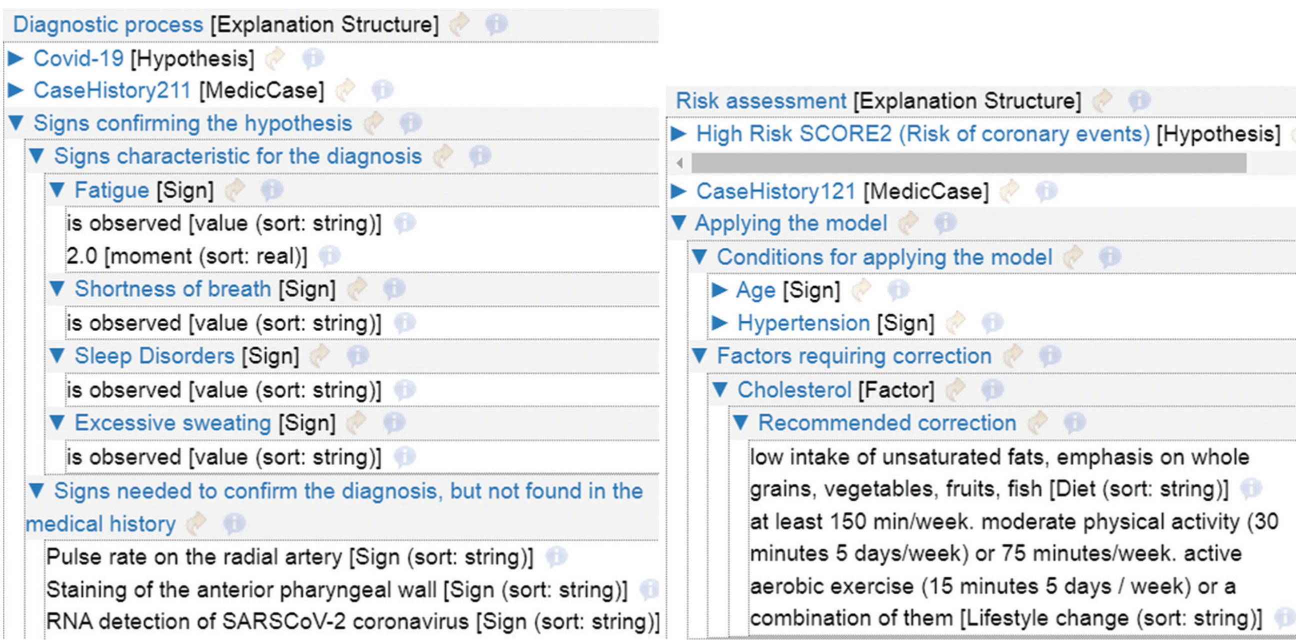

Figure 6. The fragment of hypotheses explanatory in diagnosing the Intelligent Adaptive Clinical Platform service

Inductive knowledge generation tools based on tabular data sets are used for two types of tasks. To test the knowledge, tools are used to compare solutions in the reference case history sets with the generated results of each AI component integrated into the medical IACPaaS services. Two comparative analyses were carried out: (i) opportunities for generating advice: IACPaaS-ExClDSS (including integrated ones) versus other available services, and (ii) labor costs (as the number of employees) to maintain the relevance of the knowledge base. In the cloud implementation of knowledge-enabled services, there is a division of competencies versus traditional KESs development.

The comparative analysis involved five medical internet services (https://symptoms.webmd.com, https://www.everydayhealth.com/symptom-checker, https://www.mayoclinic.org/symptom-checker, https://kiberis.ru, and https://symptomchecker.isabelhealthcare.com) and five of the most recent IACPaaS "assemblies."

Five clinical cases were randomly selected from the accumulated set of clinical cases from practice, and five clinical cases with the same diagnoses were taken from publications on PubMed. As for internet services, for 10 reference cases, the true diagnosis was rarely in the top three (Table 1): 2 – 4 times (out of 10 stars), and more frequently in the top 10: 6 – 8 times. In contrast, with IACPaaS, the true diagnosis appeared in the top three 7 – 8 times, and always within the top 10.

The results of treatment and prognosis are difficult to compare, because only one internet service was ready to issue a cure, and two of our selected IACPaaS assemblies had a treatment module (prescription was issued for eight + six cases, 6 times it coincided with prescription from the reference case).

The prognosis is always implemented separately from other tasks. It is aimed at a specific disease, while IACPaaS services can predict disease course at the initial stages of diagnosis and predict recovery course (with the prescribed treatment).

Before creating a cloud platform and developing an ontological approach, we developed several diagnostic consultants with a knowledge base based on the rules. The development of the first version of the diagnostic service for a group of diseases (therapy, ophthalmology, and gastroenterology), including about 15 diseases, took an average of 20 months: for testing with debugging (8 months), to expand with another disease (a month), and subsequent testing with debugging (2 months). With cloud implementation of KES (with separation of competencies) for a similar group of diseases, it takes on average 10 months (due to a more understandable form of writing): Ontological Diagnostic Reasoner (5 months), testing with debugging (3 months), to extend with another disease (1 month), and new testing with debugging is a month.

Table 1. Comparative analysis of diagnostic hypotheses of two types of Internet services

Type of service	The reference diagnosis was in the top three	The reference diagnosis was in the top 10
Internet	2 – 4	6 – 9
Intelligent adaptive clinical platform	6 – 8	10

Internet services provide savings by scaling their use. Theoretically, to offer many specialists and teams of the same profile, additional efforts are required only to integrate the presentations of patient data. However, in practice, services have a limited set of concepts that do not allow them to accept all the information about the patient. In the (new) cloud technology KES, with a separate declarative knowledge base and a powerful glossary of terms, there are several savings due to a single center of knowledge update and knowledge control (by accumulated case histories). Similarly, the knowledge base about treatment is improving (but there is no way to compare because the team did not have much experience in the past).

The situation to test the system's viability was requested in early 2020 to expand the service for the diagnosis, differentiation, and treatment of COVID-19. In comparison with other service providers who presented updated versions a few months after the appearance of diagnostic guidelines (Infermedica, klinica.com.ua, and medicase.pro), in our technology, the addition of an existing knowledge base to describe several known variants of manifestation, course, and diagnosing methods of the new disease took several days.

For this extension of the KES, medical experts used two knowledge base editors, adding several dozen statements of diagnosis and treatment. The accuracy of the updated knowledge base was evaluated using the first 15 case histories available. A week later, a new cloud service was launched to search for hypotheses about a patient's possible viral disease and differential diagnosis. This cloud service is an example of explanatory AI (Figure 6). It provides a rationale for the proposed solutions and recommendations (unlike the services of klinica.com.ua or medicase.pro). The service indicates which signs of the disease are/are not included in the clinical picture of the disease and whether additional information is needed to confirm or refute it. At the same time, the service asks which values of which signs need to be obtained additionally.

The new cloud service and a declarative method for accessing it (based on the existing solver) demonstrate the feasibility of the technique and approach for evolving KESs and the adequacy of the proposed infrastructure for the development and ongoing evolution of KESs.

7. Conclusion

The application of the proposed approach ensured the construction of scalable medical software services to support specialists of different profiles at different stages of work. It has been demonstrated that the proposed method for producing medical software assistants brings them to the level of explainable AI, which is the consequence of the interpretability of clinical guidelines and knowledge about the course of diseases and their management.

The proposed methodology and production environment for viable systems proved easy to learn and convenient for teamwork. For a medical diagnostic system, each significant knowledge extension (more than 20 such acts were performed in total) required from 5 h to 2 working days for an expert, 5 – 8 h for quality control, 20 min for an architect, and without a programmer, which would be unattainable in another production environment. After each update, the product characteristics analysis showed that the results were consistent with the case samples received from real practice.

Further research should focus on integrating the developed tools with textual facts, knowledge parsers, and third-party diagnostic and predictive tools. A detailed study is required to demonstrate whether the components working with structured information, verbal text, images, and digital arrays can be combined into a single complex. This approach would save valuable time for users in critical areas of activity.

Work is currently underway to expand the capabilities of the approach further. Today, the bottleneck for us is an adaptable user interface. The technology allows you to generate three user interfaces based on the explanation ontology, but these features are insufficient. We are currently working on creating tools for automatically generating an interface based on the user model, considering the usability requirements.

The main contributions of the study are: (i) the automation of multiple physician tasks by filling a single structured "medical history" (integrated with full electronic medical record), (ii) the integration of formalized knowledge from clinical guidelines and other reliable sources to satisfy both the relevance of the methods used and the personalization to the patient, (iii) the transparency of all applicable knowledge, (iv) the explainability of advice based on the essence of the knowledge and with a link to the source, and (v) the integrability of ExClDSS with neural network services, capable of inputting data from a structured document, such as the medical history. Our participation in piloting the (Ex)ClDSS in some medical clinical institutions aligns with the rhetoric of conferences emphasizing the importance of AI for healthcare.

Some of the limitations of the study include the lack of pre-existing converters of formalized knowledge (e.g., in the Protégé paradigm) into our development and support environment (this would provide an opportunity for both the integration of high-quality knowledge and the quality control of the accumulated archives of precedents), a high "entry threshold" to the IACPaaS platform for Python-savvy programmers, and insufficient attention to colorful visualization tools and flexibility of data input.

Today, we are helping to bridge the gap between AI innovation and real-world applications. The experience of moving to trial operations in 2024 has shown that doctors welcome such important general characteristics of these systems. These include the ability to explain hypotheses (results), a mechanism for adding specific knowledge (e.g., new in the clinical information guidelines), and specific properties of specific software systems (for risk assessment, diagnosis, and prognosis). Doctors particularly appreciate systems that facilitate a dialogue to increase the result's accuracy. For treatment-related software systems, the ability to apply knowledge from modern, regularly updated clinical research is also crucial. Developers of ExClDSS, using our technology, emphasize the importance of features such as procedures for regular evaluation of the knowledge base by subsets of precedents (from archived sets and cases from the practice of specialists), as well as reading and directly evaluating the knowledge contained in a specific system.

As technology developers, we consider it important to have a procedure to verify the accuracy or correctness of hypotheses based on any subsets of precedents provided by clients or potential users. Therefore, we believe there is potential for this ontological technology to bridge the gap between AI innovations and their real-world applications.

Acknowledgments

None.

Funding

The work was supported by the Ministry of Education and Science of Russia (No.: FWFW-2021-0004).

Conflict of interest

The authors declare they have no competing interests.

Author contributions

Conceptualization: Valeriya Gribova
Investigation: All authors
Methodology: All authors
Software: Elena Shalfeeva
Writing – original draft: All authors
Writing – review & editing: Elena Shalfeeva

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

Some software and information components for the study can be obtained by contacting the authors at shalf@iacp.dvo.ru.

Further disclosure

Some of the findings have been presented in the preprint (https://doi.org/10.21203/rs.3.rs-814383/v1) deposited in the preprint server "Research Square."

References

1. Peleg M. Computer-interpretable clinical guidelines: A methodological review. *J Biomed Inform*. 2013;46(4):744-763.

   doi: 10.1016/j.jbi.2013.06.009

2. Young O, Shahar Y, Liel Y, *et al*. Runtime application of Hybrid-Asbru clinical guidelines. *J Biomed Inform*. 2007;40(5):507-526.

   doi: 10.1016/j.jbi.2006.12.004

3. Novais P, Oliveira T, Satoh K, Neves J. The role of ontologies and decision frameworks in computer-interpretable guideline execution. In: Nalepa G, Baumeister J, editors. *Synergies between Knowledge Engineering and Software Engineering. Advances in Intelligent Systems and Computing*. Vol. 626. Cham: Springer; 2018. p. 197-216.

4. Deng C, Ji X, Rainey C, Zhang J, Lu W. Integrating machine learning with human knowledge. *iScience*. 2020;23(11):101656.

   doi: 10.1016/j.isci.2020.101656

5. Müller L, Gangadharaiah R, Klein SC, *et al*. An open access medical knowledge base for community driven diagnostic decision support system development. *BMC Med Inform Decis Mak*. 2019;19:93.

   doi: 10.1186/s12911-019-0804-1

6. Musen M. The protégé project: A look back and a look forward. *AI Matters*. 2015;1(4):4-12.

   doi: 10.1145/2757001.2757003

7. Mortensen J, Minty E, Januszyk M, *et al*. Using the wisdom of the crowds to find critical errors in biomedical ontologies: A study of SNOMED CT. *J Am Med Inform Assoc*. 2015;22(3):640-648.

   doi: 10.1136/amiajnl-2014-002901

8. Loh HW, Ooi CP, Seoni S, Barua PD, Molinari F, Acharya UR. Application of explainable artificial intelligence for healthcare: A systematic review of the last decade (2011-2022). *Comput Methods Programs Biomed*. 2022;226:107161.

   doi: 10.1016/j.cmpb.2022.107161

9. Yang G, Ye Q, Xia J. Unbox the black-box for the medical explainable AI via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. *Information Fusion*. 2022;77:29-52.

   doi: 10.1016/j.inffus.2021.07.016

10. *Gartner Identifies the Top Strategic Technology Trends for 2021*. STAMFORD, Conn; 2020. Available from: https://www.gartner.com/en/newsroom/press-releases/2020-10-19-gartner-identifies-the-top-strategic-technology-trends-for-2021

11. Pressman RS. *Software Engineering: A Practitioner's Approach*. 7th ed. New York, NY, USA: McGraw-Hill; 2010. p. 242-275.

12. Islam M, Katiyar V. Development of a software maintenance cost estimation model: 4th GL perspective. *Int J Techn Res Appl*. 2014;2(6):65-68.

13. Izurieta C, Bieman JM. A multiple case study of design pattern decay, grime, and rot in evolving software systems. *Software Qual J*. 2013;21(2):289-323.

    doi: 10.1007/s11219-012-9175-x

14. Breivold HP, Crnkovic I, Eriksson PJ. Analyzing Software Evolvability. *In: 2008 32nd Annual IEEE International Computer Software and Applications Conference*. IEEE; 2008:327-330.

    doi: 10.1109/compsac.2008.50

15. Finn VK. About Data Mining. *Artificial Intelligence News*. 2004; 3-18. Available from: https://masters.donntu.ru/2006/kita/balabanov/library/articles/art010.pdf (In Russian)

16. Weissler EH, Naumann T, Andersson T, *et al*. The role of machine learning in clinical research: Transforming the future of evidence generation. *Trials*. 2021;22:537.

    doi: 10.1186/s13063-021-05489-x

17. Lin Z, Cheng YT, Cheung BMY. Machine learning algorithms identify hypokalaemia risk in people with

hypertension in the United States National Health and Nutrition Examination Survey 1999-2018. *Ann Med*. 2023;55(1):2209336.

doi: 10.1080/07853890.2023.2209336

18. Kang SH, Baek H, Cho J, *et al*. Management of cardiovascular disease using an mHealth tool: A randomized clinical trial. *NPJ Digit Med*. 2021;4:165.

doi: 10.1038/s41746-021-00535-z

19. Zhang Y, Hu N, Li Z, Ji X, *et al*. *Lumbar Spine Localisation Method Based on Feature Fusion. In: CAAI Transactions on Intelligence Technology*; 2022.

doi: 10.1049/cit2.12137

20. Wang Y, Li S, Zhao B, *et al*. A ResNet-based approach for accurate radiographic diagnosis of knee osteoarthritis. *CAAI Trans Intell Technol*. 2022;7(3):512-521.

doi: 10.1049/cit2.12079

21. Golenkov VV, Gulyakina NA, Grakova NV, Nikulenka VY, Eremeev AP, Tarasov VB. From training intelligent systems to training their development means. *Open Semantic Technol Intell Syst*. 2018;2:81-98.

22. Nikolenko SI, Tulupiev AL. *Self-learning Systems*. Moscow, Russia: MCNMO; 2009. p. 288. (In Russian)

23. Dodonov AG, Landje DV. *Viability of Information Systems*. Kiev, Ukraine: Naukova Dumka; 2011. p. 256. (In Russian)

24. Kryazhych OO. Ensuring the viability of the information in time under its processing in the DSS. *Math Mach Syst*. 2015;2:170-176. (In Russian)

25. Gribova VV, Kleshchev AS. Paradigm for controlled intelligent systems. *Manag Syst Inform Technol*. 2016;3(65):32-38. (In Russian)

26. Chhabra JK. Improving modular structure of software system using structural and lexical dependency. *Inform Softw Technol*. 2017;82:96-120.

doi: 10.1016/j.infsof.2016.09.011

27. Gribova VV, Kleschev AS, Moskalenko FM, Timchenko VA, Shalfeeva EA. Extensible toolkit for the development of viable systems with knowledge bases. *Programmnaya Ingeneria*. 2018;9(10):339-348.

doi: 10.17587/prin.9.339-348

28. Gribova V, Shalfeeva E, Petryaeva M. Cloud Infrastructure for Creation of Interpretable Diagnostic Knowledge Bases of Diseases Regardless their Etiology. *In: Proceedings of the 21st International Workshop on Computer Science and Information Technologies (CSIT 2019)*. Atlantis Press; 2019.

doi: 10.2991/csit-19.2019.13

29. Kleschev AS, Chernyakhovskaya MY, Shalfeeva EA. The paradigm of an intellectual professional activity automation. Part 1. The features of an intellectual professional activity. *Ontol Design*. 2013;3(9):53-69. (In Russian)

30. Gribova V, Moskalenko P, Petryaeva M, Okun D. Cloud environment for development and use of software systems for clinical medicine and education. *In: Proceedings of the 7th Scientific Conference on Information Technologies for Intelligent Decision Making Support (ITIDS 2019)*. Atlantis Press; 2019.

doi: 10.2991/itids-19.2019.40