Multilevel Ontologies for Big Data Analysis and Processing

Maryna Popova[1], Larysa Globa[2] and Rina Novogrudska[1]
[1]*National Center "Junior Academy of Sciences of Ukraine", Dehtyarivska Str. 38-44, 04119 Kyiv, Ukraine*
[2]*Igor Sikorsky Kyiv Polytechnic Institute, Peremohy avenue 37, 03056 Kyiv, Ukraine*
pm@man.gov.ua, lgloba@its.kpi.ua, rinan@ukr.net

Keywords: Big Data, Multilevel Ontology, Taxonomization, Relations, Knowledge, Concepts.

Abstract: The problem of ever-increasing amounts of unstructured information in various fields of human activity is known as the problem of Big Data. Providing support for analytical activities requires determining the main factors that affect certain states of objects and processes in domains, as well as the degree of their influence, this significantly complicates the decision-making process, especially if data are represented heterogeneous information, there is a need to simultaneously take into account the impact of data from several areas dealing with several levels of classification. Given the significant volumes of text documents, it is impossible to solve the problem of structuring linguistic information by computer-aided extraction of the basic concepts that determine the text content (meaning), as well as the problem of constructing a formalized structure for formation the classes of individual objects and relations between them. The paper considers the ontological approach to the analysis and processing of Big Data represented both heterogeneous and linguistic data in the form of a multilevel ontology, implemented by computer-aided extracting of the basic concepts that define the text content (meaning) and determining semantic relations between the distributed information resources. The proposed approach uses the possibility of non-canonical conceptual ontologies to define equivalent concepts and thus to integrate the multiple ontologies that affect the same subject domain. This approach was implemented to create a multilevel ontology in the systemic biomedicine, the application of which in the process of postgraduate doctors and pharmacist's education has significantly reduced the search time of relevant information and errors number due to the lack of unified terminology.

1 INTRODUCTION

Today, in various fields of human activity, such as science, education, economics, health, business and other fields, there is so much data that the need to analyse and process them to improve the management of certain business processes is actual and urgently needed. It has stimulated the development of new intelligent data processing methods focused on practical application. An indisputable difficulty in solving various applied problems in different domains is the analysis and processing of Big Data that describe them and are characterized by diversity, large volumes, unstructured, as well as the inability to determine the degree of their impact on certain business processes, which, in turn, complicates decision-making processes. An even more complex problem is the decision-making process based on the processing of Big Data represented by linguistic information.

However, any human activity deals with domains that contain different components (sections), which are characterized by their own system of concepts, their knowledge and many tasks that require adequate formalized models of their representation. Ensuring the support of analytical activities requires the identification of the main factors that affect certain states of objects and processes in the domains, as well as the degree of their impact, so all available components relevant to specific tasks need to be integrated as some structure, such as a pyramidal network or graph. All these factors are important to represent and integrate at different levels. The need for simultaneous presentation of several domains that deal with several levels of classification, determined the development of the concept of multilevel modelling [1, 2].

In a broad sense, Big Data is a socio-economic phenomenon associated with the emergence of technological capabilities to analyse ultra-large data sets in some problem areas, but the entire world of a significant amount of linguistic (textual) information

requires significant automation of analytical processing [3], which consists in performing certain steps, namely: 1) structuring of linguistic (text) information due to computer-aided extraction (Data Mining, Data Extraction) of basic concepts that determine the content (meaning) of the text; 2) building a formalized structure by classes of concepts and relations between them formation; 3) determining the mechanism for conducting logical output based on the created structure.

The paper considers the ontological approach to the analysis and processing of Big Data represented by linguistic information in the form of a multilevel ontology, implemented by structuring texts and establishing semantic connections between distributed information resources.

The paper is structured as follows: Section 2 is dedicated to the analysis of the main data sources about knowledge representation in multilevel ontologies. Section 3 describes existed approaches to the implementation of ontologies, including multilevel ones. Section 4 provides an example of multilevel ontology implementation. The results of the paper are summarized in the Conclusion.

2 STATE OF ART

Today, ontological approach to automating the process of analytical processing of large amounts of textual information stored on the global network is wide spread [4-6].

Conventional ontologies (such as well-formalized OWL ontologies) use descriptive logic (first-order logic) to determine the class affiliation of individual objects and their binary relations (properties).

The concept of multilevel ontology or multidimensional ontology is to define a class of objects and their relations by geometric means in multidimensional space, not by formalisms based on first-order logic.

More formally, a multilevel ontology means a finite set of points that are classes in a multidimensional space. Multilevel ontology objects represented in the form of graph nodes are combined into classes according to certain properties, and binary relations («class-instance») – in the form of directed edges, connecting two objects. This representation of a multilevel ontology is present in almost all modern conceptual and ontological models. However, most of their use is limited due to the division of classes and instances into disparate sets. In other words, instances of classes cannot be other classes, and a multilevel taxonomy based on a class-instance relation cannot exist. Unlike first-order ontologies, binary relations between objects in a multilevel ontology are have no names. Similarly, they do not have semantically significant names and objects; instead, each is assigned a unique identifier (such as a URI in OWL ontologies).

Formally, any multilevel ontology can be reduced to an OWL ontology, but such a reduction will destroy the conceptual basis of the original multilevel ontology [7].

While creating intelligent systems based on knowledge, it is advisable to create a structure that would save the original form of the ontology, ensuring the integration of knowledge and ontologies of different domain sections. As a means of such integration, the authors suggest the use of meta-ontology, which defines the system of concepts described while creation of a domain sections ontology. Such a meta-ontology is an ontology of a more abstract level in relation to the domain sections ontology [8] (Figure 1).

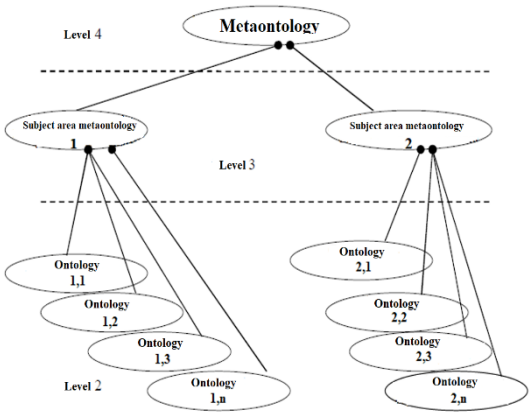

Figure 1: The multilevel ontology concept.

Metaontology, used to describe other ontologies, defines the structure of the internal organization of large ontology sections, indicating the general properties of the ontology matching types used in the domain section ontology. There are several levels of metaontology if this is necessary.

Multilevel integrated ontology (MIO) can be considered as an ontology, the concepts and roles of which are represented by dimensions, categories, measurements and facts. This ontology should also include all the axioms and statements needed to validate the intended model of multidimensional data. As a result, MIO can be used both to determine the directions of analysis and to test the resulting model for the presence of some new properties [9].

For knowledge management, multilevel ontologies and schemas for their representation are

considered mainly in the context of the data warehouses development and integration.

The authors [10] present a multilevel model with an OWL ontology model based on the descriptive logic of the Stanford Center for Biomedical Informatics Research, and define rules for transforming from a multidimensional level to an OWL ontology.

In [11] RDF-model OLAP-cube with an emphasis on the relation between the attributes of measures and measurements and its effect on the ability to summarize is described. The authors define the concept of measure- measurements consistency and demonstrate how to make logical output from OLAP ontology. The OLAP ontology is built using semantic web technologies and is mainly used to help users create OLAP cubes and queries to it.

Researchers from the Jaume I University (Spain) [9] propose to use ontology and semantically annotated data resources as a basis for designing semantic data repositories and an ontology-based environment for designing multidimensional analysis models.

In work [12] a new structure for the conceptual multilevel models' development, starting with a set of applied biomedical ontologies, is proposed. The methodology underlying the multilevel model is very simple, it is necessary to determine only the facts, indicators, measurements, categories and relations. This allowed to implement the model in almost any existing multidimensional database, performing the appropriate transformations. Regarding the scalability of the approach, the proposed solution allows you to manage large ontologies by selecting fragments that represent semantically complete modules of knowledge.

In the research [13] database transformation rules are used to generate the OWL ontology. Ontology-based technology provides semantic explanation and personalization capabilities based on the relation between concepts in the ontology. Multilevel ontology is designed to most fully reflect the terminology of a complex structured domain, identify common and partial in the content of such a complex structured area and provide the ability to reuse the description of concepts and relations in knowledge engineering and intelligent systems development.

The potential benefits of using multilevel ontologies are:

- obtaining additional levels of information presentation, which will be reused to create new levels;
- obtaining a more compact representation of the ontology text by introducing abstract concepts-relations between entities and their use in defining other terms.

Thus, the possibilities of multilevel ontology are used to solve the problem of semantic integration of reusable ontologies. In addition to the approach to finding relevant elements using metalevel specifications, the possibility of joining reusable ontologies as higher-level specifications can be considered. Acting as meta-information, such an ontology can remain independent and embedded without much efforts.

The multilevel ontology model should provide:

Facilitate the interpretation of concepts within the community. Today, there are many information systems characterized by the use of different conceptualizations, which complicates the interaction between them. Using a multilevel ontology will help solve this problem.

Reduction of errors. In many areas of human activity, such as medicine, there is no unified terminology, and the number of different ontologies is constantly increasing, which leads to semantic heterogeneity and, consequently, to the problem of semantic interoperability. However, the creation of a theoretical bridge in the form of a multilevel ontology will help to resolve ambiguities in ontological terms and concepts, thereby facilitating interpretation and reducing errors.

Data integration. Domain ontology is the only tool that allows you to reconcile at the semantic level of the model of heterogeneous data sources. Integration often occurs automatically because the ontologies used in the process capture and identify concepts in a formal and unique way.

Exchange of meaningful information. A coherent domain conceptualization can be easily used as a data exchange format. Unlike the usual exchange format, which defines the complete structure of the exchanged data and where the value of each data element is determined by its place in the global structure, ontology-based exchange is very flexible, which allows reasonable interpretation of completely different exchange structures by the same receiving system.

Extended support for semantic interoperability. Multilevel ontologies offer broader support for semantic interoperability, due to the fact that they reconcile the ontologies inconsistencies in different information systems.

Reuse of information. The ontology provides access to the data referenced by the concepts it defines. Ontologies are also used to query databases.

Existing approaches do not solve, first of all, the problem of structuring textual information by computer-aided extracting the basic concepts that determine the content (meaning) of the text, and the problem of building a formalized structure for forming classes of individual objects and relations between them. To solve these problems, it is proposed to automate the process of multilevel ontological model implementation by using software to solve analytical problems based on it.

3 MULTILEVEL ONTOLOGIES DESIGN

Ontology design is often not the ultimate goal in itself, usually ontologies are further used by intelligent systems to solve practical problems. In work [14] a 7-stage approach to ontology design is proposed.

1) Definition the scope and purpose of ontology.
2) Considering reusing existing available ontologies developed by someone else.
3) Listing of important ontology concepts, not taking into account possible coincidences of concepts that can be identified.
4) Definition of classes and their hierarchy.
5) Definition of properties related to classes.
6) Definition of constraints (number of elements, range of domain constraints), which relate to the properties.
7) Creating instances of classes in the hierarchy.

This approach uses the ability of non-canonical conceptual ontologies to define equivalent concepts and thus to integrate several ontologies describing the same domain.

An alternative approach to the ontology development starting with the canonical conceptual ontology is proposed in [15].

1) The first step in ontology development should be to agree in the community of its application. To reach an understanding, you should:
   - clearly define what is the domain described by the ontology;
   - choose a powerful model to accurately identify primitive concepts existing in the domain;
   - develop a common understanding of the canonical set of concepts that describe the field of knowledge.

2) Based on the certain canonical conceptual ontology, a non-canonical ontology can be created for practical use by a group of end users, to create their own idea of the domain or to formally model all concepts existing in the target domain related to ordinary linguistic notation (word or sequence of words). Thus, the possibility of exchanging information, expressed in concepts of the canonical conceptual ontology, is preserved.

3) To ensure that the ontology is used for linguistic output and/or to provide an end-user-friendly multi-lingual interface, it is necessary to define a list of concepts for a specific language and link them to each ontology concept. The multilevel "onion" model built based on this alternative approach [15] and obtained as a result of domain formalization, includes:
   - canonical conceptual ontology, which provides a formal basis for modelling (canonical and accurate descriptions of each concept) and effective exchange of knowledge in the domain between different sources;
   - non-canonical ontology, which provides mechanisms for linking various conceptualizations developed in this domain, which are used to interact with other software components or sources that already have their own special ontologies;
   - linguistic ontology, which represents the concept in natural language (in different languages) and sets the linguistic transformations over primitive and definite concepts.

Basic rules of ontology development according to [14] are formulated as follows:

1) There is no single right way to model the domain – there are always viable alternatives.
2) Ontology development is necessarily an iterative process – a repeated passage through the ontology in order to clarify it.
3) Ontology elements should be close to the objects (physical or logical) and relations in a particular domain.

Therefore, regardless of the choice of approach to multilevel ontology design, it is necessary to meet the basic requirements for its formation and development:

- flexibility – the ability to quickly and easily update any of ontology fragments, the ability to organize a decentralized "multi-agent" creation and editing of ontologies;
- openness – to add both individual concepts of any content and any conceptual subsystems, openness to the vocabulary of natural languages and additional options for

conceptual interpretation of words already contained in the lexicon of ontology;

- meaningful scalability – the ability to quickly select (expand/cut) certain fragments in accordance with the task, area of interest and point of view of individual professional groups;
- model scalability – the ability to present conceptual systems at different levels of detail to describe and formalize the relevant fragments of reality (for example, in the following sequence: simple semantic categorization of vocabulary – taxonomy – complete terminological model – production system – logical theory);
- versatility for the user – suitability for use in various software components and on different platforms.

4 MULTILEVEL ONTOLOGY IMPLEMENTATION

In this research the concept of ontology "level" is considered somewhat more widely.

First, the "internal" ontology level characterized by the depth of the binary relations of the taxonomy that underlie it is determined. The depth of binary relations means the depth of nesting of concepts categories, in graph terms it means that there is a certain distance between the terminal and root nodes, which exceeds 1 step: the greater the number of steps from root to terminal node, the higher the level of ontology.

Secondly, the concept of "external" ontology level characterized by the number of search iterations at the user request from the taxonomy concept context in the information sources integrated into the ontology is considered.

Multilevel ontology means a logical-linguistic model, the first level of which is represented by concepts in the form of logical formulas, reflecting the patterns inherent in the classes of objects and logical relations, the second – the corresponding concepts of consistent ontologies, the third and subsequent – semantically related information units contained in heterogeneous distributed sources of knowledge created by different standards and technologies and described in natural language (databases and knowledge bases, information banks, electronic archives, collections of electronic documents, etc.) (Figure 2).

4.1 Structuring Texts by Basic Concepts Computer-Aided Extraction

At the stage of first level of multilevel ontology formation the natural language texts taxonomization and contexts transdisciplinary categorization are carried out, that includes:

Concepts extraction – search in natural language documents terms that reflect the names, characteristics and relations between these terms.

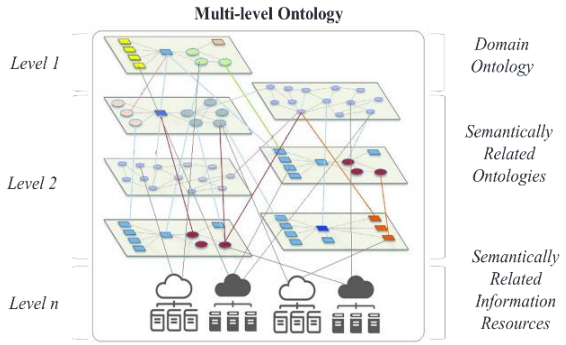

Figure 2: Multilevel ontology designing stages.

Usually, knowledge from any field of research is presented in text documents that contain poorly structured or even unstructured information. Processing such documents manually can be an extremely time-consuming process, and processing large arrays of such documents is almost impossible.

Before working with poorly structured or unstructured documents, it is necessary to structure them. During this process, the data is presented in a form convenient for computer-aided processing, which is easily read by standard means of ontologies designing, and is displayed in a user-friendly form.

The most difficult is to perform the structuring of natural language texts, as this process requires a sufficiently complete formal description of the language subset to which they belong. Each of the texts describes a specific area of research or part of it. The text uses the concepts that form its terminology. The text structuring consists of the concepts field extraction, in particular, the relevant field of study concepts (terms) identification, as well as their attributes and relations. The formed concepts field, in turn, can be represented as an ontology.

The task of text documents arrays structuring is to natural language process, that includes natural language semantic-linguistic analysis – the natural text documents processing, including formalization of the syntactic-semantic structure of sentences, computer-aided selection of multiword terms and

contexts in which they used, and given semantic relations based on templates of their descriptions.

The main result of semantic-linguistic analysis is the construction of a domain glossary – a list of objects that exist within it, that can act as either terms or names of certain entities. If the terms are read from the text, then in the future the text highlights the contexts in which these terms are used, and if possible – their definitions too. If named entities are read, then the process of determining the attributes of these entities is carried out in the future. Thus, the structuring of a certain natural language text T can be represented as a certain transformation (structuring transformation):

$$F_{str} : T \to O, \quad (1)$$

In fact, the text structuring transformation is a multi-stage process, each stage of which requires the use of specialized models and procedures. The process of text structuring can be divided into two main stages: lexical analysis $T \to T_{pr}$, which forms the primary structure of the text T_{pr}, and the ontology formation $T_{pr} \to O$, which allows you to select the necessary information from the primary structure and present it in the form of an ontology. To perform the second stage, a recursive reduction method is proposed [5, 16, 17], which provides a sequential primary structure transformation using a set of dynamically specified by the ontology designer rules.

The procedure of concepts extraction from natural language texts is implemented as one of the cognitive information technology "POLYHEDRON" [18] modules "KONSPEKT" [19], which functions include:

- text linguistic analysis to the level of superficial syntactic and semantic analysis;
- extraction of domain concepts from relevant texts;
- extraction and contextual description of the natural language texts concepts related to a given topic, which is given by a keyword or phrase;
- generation based on the results of semantic analysis of a given number of secondary keys, the use of which in a cyclic mode allows to deepen the disclosure of the topic in the formed contextual concepts descriptions;
- use contextual concepts descriptions to select from a set of text documents those that are most relevant to a given topic.

Terms from natural-language texts are distinguished using procedures and software tools for linguistic and semantic analysis of texts.

«KONSPEKT» [19] provides computer-aided contexts extraction that use the corresponding terms, and presentation of them in the form of a specialized XML structure. It allows computer-aided fill in the graph nodes with contexts based on the coincidence of nodes names and concepts names based on the results of semantic and linguistic texts analysis.

The results of semantic and linguistic analysis are used for natural-language texts taxonomization – a cognitive procedure for structuring text arrays based on the systemological representation of their terminological system in a hierarchical form. Because of natural-language text taxonomization, its structure can be represented as a graph, each node of which contains the corresponding contexts or attributes. Contexts content includes, respectively, semantic descriptions and characteristics of the corresponding concepts and phrases or characteristics of named entities.

Taxonomization provides extraction of classification units from text array that characterize its semantics and purpose. The text taxonomy reflects the order of interaction between terminological constructs or named entities.

Establishing relations between concepts. Relations indicate interactions between concepts. They are defined by properties and attributes that characterize domain classes.

Due to the established relations, ontology is not just a structure of concepts, but also reflects complex relations between them and comprehensively represents the domain. There are three main types of relations between concepts:

- R_t – taxonomic relations – express the relation «is-a» or the relation «general/partial»;
- R_c – compositional relations – express the relation «part of»;
- R_{top} – topological relations – reflect how different components of a terminological system are connected to each other through certain connections, or show the «paths» of physical interactions between components, as well as provide information about the spatial location of these components.

The definition of multiple relations of binary order over thematic concepts allows to achieve a high level of correctness in the formation of taxonomic categories and thematic classifiers. This ensures multiple interactions between taxonomic structures.

The result of applying the text taxonomization procedure is the definition of semantically significant relations between various objects, which can include

both certain relations between terms belonging to the domain (synonymy, class-subclass, etc.), and specific relations between named entities for this area of knowledge.

Representation of the primary text structure as a taxonomy. The taxonomy organizes concepts in a controlled dictionary into a hierarchy. The main purpose of the taxonomy is to create an ontology structure for human understanding and integration of other sources. In taxonomy, binary relations between different concepts of a domain are determined based on their definitions.

The primary text structure T contains a structured representation of lexemes (words or symbols), as well as syntactic relations between them. This structure, in fact, is an oriented graph, and lexemes are nodes of this graph.

Any natural-language text T is represented by a set of lexemes L, on which the precedence relations are defined $\Leftarrow$. This relation converts L to a linearly ordered set. The text T can also be represented as a sequence of sentences S that also define the precedence relation:

$$T = \left\{ S_1 \Leftarrow S_2 \Leftarrow ... \Leftarrow S_{n_t} \right\}, \quad (2)$$

where n_t is the total number of sentences in the text.

Each sentence S_i is represented by some subset of lexemes:

$$L_{S_i} = \left\{ l_{ij}, j = \overline{1, n_i} \right\}, \quad (3)$$

where n_i – the number of lexemes in *i* sentence.

Obviously, the condition is met:

$$\forall l_1 \in S_1, \forall l_2 \in S_2, S_1 \Leftarrow S_2 \Rightarrow l_1 \Leftarrow l_2, \quad (4)$$

where S_1, S_2 are arbitrary text sentences;

l_1, l_2 – lexemes.

Each lexeme has a number of attributes:

$$l_{ij} = \left\langle l_{ij}^T, P_{ij} \right\rangle, \quad (5)$$

where l_{ij}^T is the text representation of the lexeme l_{ij};

P_{ij} – lexeme attributes.

A lexeme can be related to other lexemes using syntactic relations $r_S \in R_S$:

$$r_S = \left\langle l^1, l^2, r_t \right\rangle, \quad (6)$$

where l^1, l^2 – lexemes that have a relation between them;

r_t – relation type.

Thus, oriented graph representing the primary structure of a natural-language text has the form:

$$T = \left\langle L, R_S \right\rangle. \quad (7)$$

The main problem is the inefficiency of working with the text representation of the lexeme, which is redundant and requires the construction of specialized functions defined on the set of words representations in text. Such functions are cumbersome and inefficient, and in software implementation, they often depend on the specifics of implementing text variable processing in a given programming language.

Since the set of text representations of lexemes is incalculable, we can construct a transformation of the form:

$$V: L^T \to \mathbb{N}, \quad (8)$$

where L^T is the set of text representations of lexemes.

Let the text be written in a specific alphabet A, the number of characters in which $n_\mathrm{A} = card(\mathrm{A})$. This alphabet can be considered as a notation with a base n_A. Accordingly, each letter $\alpha \in \mathrm{A}$ can be matched with a certain number $i_\alpha \in N$, which is the index of this letter in the alphabet. Any word in the input text is a sequence:

$$l^T = \left\{ \alpha_1, \alpha_2, ..., \alpha_{n_l} \right\}, \quad (9)$$

where n_l – word length $n_l > 0$;

α_i – letters of the alphabet A.

If we consider letters A as digits of a number in the corresponding notation, then such a number can be converted to decimal using the formula:

$$V(l^T) = i_{\alpha_1} \times (n_\mathrm{A})^{n_l} + i_{\alpha_2} \times (n_\mathrm{A})^{n_l - 1} + ... \\ ... + i_{\alpha_{nl}} \times (n_\mathrm{A})^0, \quad (10)$$

where i_{α_j} – letter index α_j in alphabet A;

n_A – number of characters in the alphabet A.

Using the V function, you can replace all l^T with their corresponding $l^V = V(l^T)$. As a result of this operation you can get a more efficient representation of the lexemes set:

$$\left\langle l^V, P \right\rangle \in L^V, \quad (11)$$

where l^V – code representation of a lexeme;

P – grammatical characteristics of the lexeme;

L^V – multiple code representations of lexemes.

In the future L^V, it can be considered as a set of lexemes L.

The taxonomy formation algorithm is based on the induction of utterances based on the selection of pairs (class name – name concept). If the statement is true, then a bipartite graph is constructed (a

unidirectional oriented graph with several edges entering and exiting one node). If the statement is false, then the graph is not constructed. The truth of the statement is established based on identifying the existence of a unifying property that is common to both concepts. The set of all bipartite graphs that are built on the set of true statements is a growing pyramidal network that is the basis of the taxonomy. The nodes contain class names and concept names.

Formally the technological basis for taxonomy formation is determined by a loaded bipartite graph:

$$G = (\hbar_1 \bigcup \hbar_2, E), \quad (12)$$

where $\hbar_1 \bigcup \hbar_2 = \varnothing$, nodes with $\hbar_1$ marked predicate names, and nodes with $\hbar_2$ marked argument names;

E – set of edges. Graph edges connect nodes marked with predicate names to vertices marked with argument names.

Vertices from the set $\hbar_1$ are called predicate nodes, vertices from the set $\hbar_2$ are called concept nodes, and predicates themselves are called conceptual predicates.

The statement is formed based on the composition of nodes incident to a single edge.

The ontological graph acts not only as a means of organizing information, but also as a environment for active user interaction with distributed information resources displayed in the form of a spatially ordered set of statements.

The effectiveness of using taxonomies in the process of integration and aggregation of information resources significantly depends on the quality of a domain structuring. Therefore, questions related to the ordering a set of taxonomic concepts determine the constructiveness of the knowledge system.

Axiomatization. Axioms provide the correct way to add Boolean expressions to ontology. Such logical expressions can be used to clarify concepts and relations in ontology. Axioms are used to develop an explicit way of expressing what is always true. Axioms can be used to determine the meaning of several components of ontology, identify complex relations, and verify the correctness of information or obtain new information.

Thus, the cognitive procedure for multi-stage sequential transformation of the primary text structure into an ontological form based on the selection of primary patterns – recursive reduction of natural language contexts that provides computer-aided transformation of text arrays into a taxonomy, thesaurus and ontology. The result of applying the procedure is the identification of lexemes (words or symbols, such as punctuation marks) that make up the attributes of domain objects (in particular, their names), the identification of primary intercontextual relations, and the taxonomic representation of text semantics.

The reduction process consists of sequentially extracting objects from the input text (a glossary of the domain is formed), relations between objects (domain taxonomy is formed) and attributes of objects (which are later considered as functions of interpretation (axioms), which allows us to consider the result of reduction as an ontology). This process can be represented by the following formula:

$$T \to O^1 \to O^2 \to O. \quad (13)$$

The recursive reduction method [16, 20] consists in recursively performing the process of reducing the input natural-language text, which, in turn, is carried out by applying a specialized operator to it:

$$F_{rd} : T \to O. \quad (14)$$

The reduction operator is a combination of four operators:

$$F_{rd} = F_{l*} \circ F_x \circ F_r \circ F_{ct}, \quad (15)$$

where F_{l*} is the aggregation operator that performs the auxiliary function of extracting phrases from the text that can represent a specific object; F_x is the operator for identifying ontology objects X. This operator applies a condition to the extracted phrases that determines whether to interpret a particular phrase as the name of an object; F_r – operator for identifying ontological relations R divided into relations between objects and auxiliary relations between the object and its contexts; F_{ct} – context identification operator that extracts its attributes from the context of a particular object (defined using the auxiliary relations extracted at the previous stage).

In general, each of the four transformation execution operators F is defined by the database of rules $DB_{\mathbb{R}}$ for performing this transformation. The rule RDBR has a unified structure for all stages:

$$\mathbb{R} = \langle f_{ap}^{\mathbb{R}}, f_{tr}^{\mathbb{R}} \rangle, \quad (16)$$

where $f_{ap}^{\mathbb{R}}$ – applicability function, which determines whether the rule can be applied to a specific set of input information;

$f_{tr}^{\mathbb{R}}$ – transformation function, which defines the transformation of input information.

The transformation $F_{\mathbb{R}} : X \to Y$ defined by the rule $\mathbb{R}$ has the form:

$$F_{\mathbb{R}}(x) = \begin{cases} f_{tr}^{\mathbb{R}}(x), f_{ap}^{\mathbb{R}}(x) \\ x, \neg f_{ap}^{\mathbb{R}}(x) \end{cases} . \quad (17)$$

So, knowledge structuring by taxonomizing natural-language texts describing this knowledge to reflect the semantics of integrated and aggregated information resources in the form of hierarchical structures, over which a certain extensible axiomatic is defined and between which sets of relations are defined, allows us to solve the problem of their correct interpretation in the process of using ontology.

An important property of ontologies is the ability to structure information simultaneously with its perception. In this case, the formation of the memory structure occurs due to the interaction of perceived information and information that is already stored in the network graph. Because of the implementation of information structuring processes, the semantic and syntactic proximity of information is established. The found associative relations are fixed in the structural components of memory.

4.2 Ontological Interface Design

According to the ontological graph (taxonomy) model by means of computer-aided code generation by comparing the taxonomy objects with the set of source codes in the programming language was designed the ontological interface – a means of user-friendly interaction with the ontology [21, 22].

Changing the taxonomy (structure of the ontology) does not require making changes to the interface code, which provides dynamic extensibility, because it describes the correspondence between the ontology components and the target programming language instructions. Thus, the interface code generator is controlled by an ontology model, which is implemented as a wide set of software components and consists of static and dynamic parts. The static part contains file templates that implement fixed algorithms for controlling the code generation process, and the dynamic part contains algorithms for mapping descriptions of interface model components to program code (programming language instructions).

Ontological interface elements are the information content of a multilevel ontology. The visual representation of an ontology object is an image (drawing, picture, icon, photo, etc.), the source of which is specified in the corresponding node of the ontograph (taxonomy). The order of object display (in the form of an image gallery) of taxonomy concepts on the screen depends on the internal organization of nodes in the ontograph. The text description of the ontology object and links to sources of distributed information resources are displayed next to the image and have a common style for all objects (colour, size and font style, position in relation to the image, corresponding icons for links to information resources of various formats, etc.).

The ontological interface has tools for both horizontal and vertical navigation with elements of the slide menu and hamburger menu, which automatically adapts to different screen widths and mobile platforms. The “Prism” view mode uses full-screen navigation tools with the location of text and graphic elements of the ontology on 100% of the screen space. Therefore, ontological interface tools take advantage of the most common types of network resource navigation to reduce cognitive load and increase the efficiency of working with the ontology.

Based on the diversity of ontologies, establishing semantic agreement between them to ensure interoperability is a necessary condition for the formation of a second level multilevel ontology.

Ontology matching is the process of establishing a connection (conjunction) between different ontologies without changing the original ontology, so that both parties can get a common understanding of the same object [23]. It can also be defined as the process of finding a suitable object with the same or closest predictable value between two or more ontologies [24]. Otology matching takes two ontologies as input data and creates a semantic correspondence between entities in the two input ontologies. The authors [25] define ontological matching as follows: “Given two ontologies O_1 and O_2, matching one ontology with another means that for each entity (concept C, relation R, or instance I) in the O_1 ontology, we are trying to find a corresponding entity that has the same valid value in the O_2 ontology”.

Ontology matching can also be defined as a process in which two ontologies with overlapping content are linked at the conceptual level, and instances of the original ontology are computer-aided converted to instances of the target ontology according to existing relations [26].

Ontology matching is established after analysing the similarity of certain metrics of entities in comparable ontologies. The result of the ontology matching process is called alignment. Alignment is defined as a set of correspondences that represent relations between different entities. A match can be described by a tuple:

$$\langle id, s, e, r, v \rangle, \quad (18)$$

where id – unique match ID,

s – source ontology object O,

e – the essence of the target ontology O',

r – alignment relations such as equivalence (=), more general, intersection and disjunctionality of two entities and

v – reliability value, such as the similarity value.

Measurement of correspondence is the basis of all comparison algorithms, as it determines the degree of similarity between the ontologies to be matched. The formal designation of the similarity degree is given below [27]:

$$sim : E \times E \to R. \quad (19)$$

Similarity function:

$$E = E_1 \bigcup E_2, \quad (20)$$

where E_1 – a set of entities in the O_1 ontology,

E_2 – a set of entities in the O_2 ontology that receives two entities as input and calculates the similarity value.

The authors [28] provide some examples of similarity measurement that can be used when ontology matching. They include:

Terminological method that compares entity/concept labels. It uses purely syntactic approaches, as well as the use of vocabulary such as WordNet. The syntactic approach calculates correspondence using measurements of chain dissimilarity, while the lexical approach calculates correspondence using lexical relations such as synonymy and hyponymy.

Method for comparing internal structures that compares the internal structures of concepts, such as the value interval and attribute power.

Method for comparing external structures that compares relations between entities and other ontology components. It provides methods for comparing entities in an ontology and methods for comparing external structures by taking into account loops.

Semantic method compares interpretations or entities models in ontology.

To form the third and subsequent levels of a multilevel ontology, indexing of a set of information resources (Big Data sources) was performed using lexicographic systems virtualization technology [29] and an agent approach.

Thus, the third and subsequent levels of multilevel ontology are digital collection of documents formed as a result of systematization of network resources, a set of natural-language texts united by one or a set of features (linguistic, conceptual, pragmatic, temporal, stylistic, functional, intentional, etc.). The most popular are collections of texts with the same topic (educational and scientific collections), one author (complete works), a certain historical era, a certain language or created under certain circumstances, in a certain form, for a certain purpose (educational and methodological materials, normative legal acts regulating legal relations in a certain area, etc.), for a category of readers with a certain level of access (public data, data for official use), etc. Modern information and communication technologies allow doing this dynamically, selecting full-text documents relevant to the user's request from a Web supermassive of indexed texts or local databases – specialized electronic libraries.

An important tool for studying text collections is the relation of semantic identity of natural-language texts.

While forming online digital collections of text documents, the following methods are used: comparative analysis (checking texts for semantic identity), system analysis (researching semantic identity as a relation with system-forming properties) and modelling the relation of semantic identity of natural-language texts.

The natural-language text expertise is based on the representation of a natural-language text as an “ordered hierarchy of content objects”.

Theory of lexicographic systems [30] operates with the concept of elementary information units (EIU), which it interprets as a subsystem of relatively stable discrete entities that is induced in the structure of any system and develops as a result of the action of various types of L-effects. Accordingly, all non-elemental objects of the system are considered as certain combinations of EIU.

During processing, a two-level hierarchy of text content objects is set. At the first (“upper”) level fragments of text, “thought blocks” that reveal one topic (micro theme), at the second (“lower”) – their constructive units (components) – thought objects, carriers of subject meanings and relations – words, phrases, combinations of words. Thus, the components of fragments act as elementary information units.

Concepts that denote the same subject are identical. At the same time, if the volume and the same generic feature completely coincide, they have different content and differ in species characteristics.

Creating collections of network texts in a multilevel ontology is provided by using specialized technologies of ontology-driven web or intranet crawling. The crawler subsystem is tightly integrated with the corps system and indexing system and the

multi-language synonym zone. Crawlers, like the corps system, are virtualized lexicographic agents, they are a type of lexicographic systems.

The identity search methodology involves pre-processing texts to improve the efficiency and speed of search – normalization.

Searching for text identities on the Internet using web crawlers is a non-trivial task. The search scenario is as follows:

- the length of queries is determined (8 words in a series are considered optimal);
- queries are generated from eight overlapping words (1 – 8; 2 – 9; 3 – 10 etc.). A single-word overlap gives you the highest search quality, although it takes longer to complete the query, so for large texts, the overlap should be as small as possible. Requests refer to the crawler API.

The array of texts received in response to the query contains a significant amount of search noise – mistakenly defined as text identities, so it is subject to further processing.

In further processing, suffix trees, the thesaurus method with a multi-language synonymous zone, the shingles method, Bag of Words, the N-Gram method, and distributive semantics are used.

Described methodology usage was tested while developing multilevel ontology "Systemic biomedicine". Its structure is shown on Figure 3.

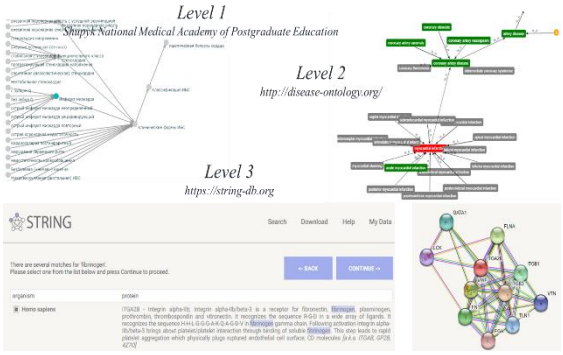

Figure 3: Example of implementing the multilevel ontology "Systemic biomedicine" scheme.

Each of the levels of a multilevel ontology can be flexibly expanded and supplemented with new objects, as well as integrate distributed information systems and sources of information resources.

In practice, multilevel information classification is not introduced in many data models. The main reason lies in the high computational complexity of logical problems associated with its modelling. The description of a multilevel classification cannot be modelled in first-order logic, since the metaclass is modelled by a second-order statement in which the variable bound by the quantifier must take values corresponding to the classes.

Thus, the ontology description language Ontolingua [31] is based on first-order logic, given the use of the KIF language for statements, and therefore does not allow multilevelness from the very beginning. The ontology language in Semantic Web Technologies OWL uses the RDF Schema language as its basis, which allows modelling of metaclasses. The OWL Full dialect uses this feature, but it is not allowed. The OWL DL dialect does not preserve the semantics of RDF Schema classes, but introduces its own, corresponding to descriptive logic, which assumes a subset of first-order constructs in which you can solve feasibility problems and some other logical problems. The same applies to OWL 2 language profiles and their corresponding logics. None of the profiles introduces the possibility of modelling metaclasses, despite the fact that approaches are used to increase the expressive power, in particular, in the RL profile – by introducing conditions for the use of constructs in statements of superclasses and subclasses [32]. Therefore, to create a multilevel ontology, a specific XML format is used that can ensure the interoperability of information at all levels to implement multilevel ontology.

5 CONCLUSIONS

Carrying out research has shown that there are no approaches to solve the problem of structuring text information by computer-aided extraction of the basic concepts that determine the text, content (meaning) as well as the problem of constructing a formalized structure for formation the classes of individual objects and relations between them.

The solution to the problem of significant linguistic information amounts analytical processing available on the Internet is proposed.

It consists of performing such steps of texts processing as structuring texts by computer-aided extraction of basic concepts that determine the content (meaning) of the text; building a formalized structure for the individual objects' classes and relations between them formation; determining the mechanism for conducting logical output based on a multilevel ontology.

The modified model of a multilevel ontology differs from the known ones in that the concept of a level is considered not in three-dimensional space, but in multi-dimensional space due to the semantic connectivity of distributed information resources.

The multilevel ontology model is able to provide facilitating the interpretation of concepts within the community, reducing errors due to the lack of unified terminology, data integration, exchange of meaningful information, extended support for semantic interoperability, information reuse.

In the future, it is planned to use the proposed multilevel ontology model to improve the mechanisms for searching and conducting logical inference.

REFERENCES

[1] B. Neumayr, K. Grn, and M. Schrefl, "Multilevel domain modeling with m-objects and m-relationships," Proc. of 6th Asia-Pacific Conf. on Conc. Model., New Zealand, 2009.

[2] C. Atkinson and T. Khne, "The essence of multilevel modeling," Proc. of 4th Int. Conf. on the Unified Model. Lang., pp. 19-33, Toronto, Canada, 2001.

[3] V. Mayer-Schonberger and K. Cukier, Big Data: A Revolution That Will Transform How We Live, Work, and Think, Canada: Eamon Dolan/Houghton Mifflin Harcourt, 2013, p. 242.

[4] A. Luntovskyy and L. Globa, "Big Data: Sources and Best Practices for Analytics," Proc. of International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo'19), pp. 1-6, 2019.

[5] O. Stryzhak, V. Prychodniuk, and V. Podlipaiev, "Model of Transdisciplinary Representation of GEOspatial Information," in Advances in Information and Communication Technologies. Lecture Notes in Electrical Engineering, vol. 560, Cham: Springer, 2019, pp. 34-75

[6] M. Popova, O. Stryzhak, O. Mintser, and R. Novogrudska, "Medical Transdisciplinary Cluster Development for Multivariable COVID-19 Epidemiological Situation Modeling," Proc. Of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2020), 2020, pp. 1662-1667, doi: 10.1109/BIBM49941.2020.9313204.

[7] G. Barzdins, N. Gruzitis, G. Nešpore-Bērzkalne, B. Saulīte, I. Auziņa, and K. Levāne-Petrova, "Multidimensional Ontologies: Integration of Frame Semantics and Ontological Semantics," Proc. of 13th EURALEX International Congress, pp. 23-28, Barcelona, Spain, 2008

[8] I. L. Artemyeva, "Complexly structured subject areas. Construction of multilevel ontologies," Information Technology, vol. 1, pp. 16-21, 2009.

[9] V. Nebot, R. Berlanga-Llavori, J. Pérez-Martínez, M. Aramburu, and T. Pedersen, "Multidimensional Integrated Ontologies: A Framework for Designing Semantic Data Warehouses," Data Semantics, vol. 13, pp. 1-36, 2009.

[10] N. Prat, J. Akoka, and I. Comyn-Wattiau, "Transforming multidimensional models into OWL-DL ontologies," Proc. of Multidimensional Models Meet the Semantic Web: Defining and Reasoning on OWL-DL Ontologies for OLAP, Hawaii, USA, 2012.

[11] T. Niemi and M. Niinimäki, "Ontologies and summarizability in OLAP," Proc. of the Proceedings of the 2010 ACM Symposium on Applied Computing(SAC'10), pp. 1349-1353, March 2010.

[12] O. P. Mintser and V. M. Zaliskiy, Systemic biomedicine. Kyiv: Interservice, 2019, p. 552 .

[13] L. El Saraj, B. Espinasse, T. Libourel, and S. Rodier, "Towards Ontology-Driven Approach for Data Warehouse Analysis," Proc of 8th Int. Conf. on Software Eng. Advances (ICSEA 2013), pp. 1-6, Venice, Italy, 2013.

[14] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Technical report ksl-01-05 and stanford medical informatics technical report smi-2001-0880, Stanford Knowledge Systems Laboratory, 2001

[15] S. Jean, G. Pierra, and Y. Ait-Ameur, "Domain Ontologies : A Database-Oriented Analysis," Proc. of Web Inf. Sys. and Techn. (WEBIST'2006), pp. 238-254, Set`ubal, Portugal, April 2006.

[16] V. Prychodniuk, "Technological means of transdisciplinary representation of geospatial information," ITGIS, Kyiv, 2017.

[17] V. Prychodniuk, "Taxonomy of natural-language texts," Information Models and Analyses, vol. 5(3), pp. 270-284, 2016.

[18] O. Ye. Stryzhak, L. S. Globa, V. Y. Velichko, M. A. Popova, and others, "Computer program Cognitive IT platform "POLYHEDRON"," Certificate of copyright to the work №96078 dated 17.02.2020, Official bulletin No 57 (31.03.2020), pp. 402-403, 2020.

[19] V. Velychko, M. Popova, V. Prykhodniuk, and O. Stryzhak, "TODOS – IT-platform for the formation of transdisciplinary informational environments," Syst. Arms and Milit. Equip., vol. 1(49), pp. 10-19, 2017.

[20] O. Ye. Stryzhak, V. V. Prykhodniuk, S. I. Haiko, and V. B. Shapovalov, "Display of network information in the form of interactive documents. Transdisciplinary approach," Math. Model. in econ., vol. 5(3), pp. 87-100, 2018.

[21] M. Popova, "Ontology of interaction in the environment of the geographic information system," ITGIS, Kyiv, 2014.

[22] M. Popova, "A model of the ontological interface of aggregation of information resources and means of GIS," Inf. Tech. and Knowl., vol. 7(4), pp. 362-370, 2013.

[23] C. Rung-Ching, L. Bo-Ying, and B. Cho-Tscan, "Using Domain Ontology Mapping for Drugs Recommendation," Department Of Information Management, Chaoyang University Of Technology, Taiwan, 2009

[24] I. Olaronke, A. Soriyan, and I. Gambo, "Ontology Matching: An Ultimate Solution for Semantic Interoperability in Healthcare," Int. J. Comp. App., vol. 51, pp. 7-14, 2012, doi:10.5120/8325-1707.

[25] M. Ehrig and S. Staab, "QOM – Quick Ontology Mapping," Proc of the Int Sem. Web Conf., vol. 3298, pp. 683–697, 2004.

[26] B. Veli, B. L. Gokce, D. Asuman, and K. Yildiray, "Artemis Message Exchange Framework: Semantic Interoperability of Exchanged Messages in the Healthcare Domain," Software Research and

Development Center, Middle East Technical University (METU), Ankara, Turkiye, 2006.

[27] S. Z. Katrin, "Instance-Based Ontology Matching and the Evaluation of Matching Systems," Inaugural-Dissertation. Department of Computer Science, Heinrich Heine University of Dusseldorf, Germany, 2010

[28] E. Rahm and P. Bernstein, "A Survey of Approaches to Automatic Schema Matching," VLDB Journal, pp. 334-350, 2001.

[29] M. V. Nadutenko , "Virtualized lexicographic systems and their application in applied linguistics," ULIF, Kiev, 2016.

[30] V. A. Shirokov, Information theory of lexicographic systems. Kyiv: Dovira, 1998, p. 331.

[31] The ontology description language Ontolingua [Online]. Available: http://www.ksl.stanford.edu/ software/ontolingua, July 2005.

[32] A. E. Vovchenko, V. N. Zakharov, L. A. Kalinichenko, D. Yu. Kovalev, O. V. Ryabukhin, and oth., "Multilevel specifications in conceptual and ontological modeling," Proc. of 13th All-Russian Scientific Conf. (RCDL'2011), pp. 35-43, Voronezh, 2011.