Methods and Models of Intellectual Processing of Texts for Building Ontologies of Software for Medical Terms Identification in Content Classification

Vasyl Lytvyn[0000-0002-9676-0180]1, Yevhen Burov[0000-0001-6124-3995]2,
Petro Kravets[0000-0001-8569-423X]3, Victoria Vysotska[0000-0001-6417-3689]4,
Andriy Demchuk[0000-0001-5710-9347]5, Andrii Berko[0000-0001-6756-5661]6,
Yuriy Ryshkovets[0000-0001-5831-4000]7, Serhii Shcherbak[0000-0003-2914-2101]8,
Oleh Naum[0000-0001-8700-6998]9

[1-6]Lviv Polytechnic National University, Lviv, Ukraine
[7]SoftServe, Lviv, Ukraine
[8]EPAM, Lviv, Ukraine
[9]Ivan Franko Drohobych State Pedagogical University, Drohobych, Ukraine

Vasyl.V.Lytvyn@lpnu.ua[1], Yevhen.V.Burov@lpnu.ua[2],
Petro.O.Kravets@lpnu.ua[3], victoria.a.vysotska@lpnu.ua[4],
Andrii.B.Demchuk@lpnu.ua[5], Andrii.Y.Berko@lpnu.ua[6],
yuriy.v.ryshkovets@lpnu.ua[7], sergey.shcherbak@gmail.com[8],
oleh.naum@gmail.com[9]

Abstract. The article investigates the problem of automated development of basic ontology. A method, algorithm and means for extracting knowledge from natural text are proposed. It is shown that such an algorithm should be multi-stage and include a hierarchical multi-level procedure for recognizing concepts, relationships, predicates and rules, which are introduced as a result of ontology. The analysis of the subject area is the search and analysis of various information systems analogues. The analysis of methods and criteria of information systems is carried out. The analysis of the system and its functionality is presented. This paper examines methods and models of intellectual text processing, the results of which are intended to build software ontologies, and are used during Ontology Learning, when it is necessary to improve, extend, modify an existing ontology model, or build ontology from basic ontology, having only text-collection collections as sources of knowledge. In the latter case, the task is particularly complex and requires the use of the full range of text mining methods (Text Mining as TM). The paper deals with the solution of TM problems at different stages of the PMT processing: obtaining information (identifying entities - concepts and terms, their properties, facts, events, establishing relationships between entities, in particular associative ones), categorization, and clustering, semantic annotation. Below we will consider the tools of automated analysis of natural language and software products implemented on their basis for filling the system.

2019 IDDM Workshops.

Keywords: ontology, ontology training, automated development, knowledge base, text document, content classification.

1 Introduction

This paper examines methods and models of intellectual text processing, the results of which are intended to build software ontologies, and are used during Ontology Learning, when it is necessary to improve, extend, modify an existing ontology model, or build ontology from basic ontology, having only text-collection collections as sources of knowledge. In the latter case, the task is particularly complex and requires the use of the full range of text mining methods (Text Mining as TM) [1-6]. The paper deals with the solution of TM problems at different stages of the PMT processing: obtaining information (identifying entities is concepts and terms, their properties, facts, events, establishing relationships between entities, in particular associative ones), categorization, and clustering, semantic annotation [7-14]. Below we will consider the tools of automated analysis of natural language and software products implemented on their basis for filling the system [15-22].

There are a number of perspective linguistic developments, among which it is expedient to single out the method of Part-of-Speech-tagging, which consists in the automatic recognition of which part of the language belongs to each word in the text [2, 23-28]. Two types of algorithms are used to improve the accuracy of such analysis: probability statistics and algorithms based on production rules that operate on words and codes. For the latter, they may use rules that are automatically collected from a corpus of texts [3, 29-34] or prepared by qualified linguists [4, 35-41].

Unlike lexical-grammatical analysis, the purpose of syntactic parsing (Text Parsing) is the automatic construction of a phrase tree that is, finding interdependencies between different levels of a sentence. There are a number of different approaches to parsing, for example, Ergo Linguistic Technologies Parser, developed by D. Bickerton and F. Braalik of Honolulu University [5-7]. The analyzer uses a notation scheme adopted at Penn Treebank, it is oriented to implementation in question-answer interfaces and is a commercial product. Another successful syntax analyzer is the Functional Dependency Grammar, built by researchers at the University of Helsinki (founders of Lingsoft and Conexor). The basis of the analyzer is the theory of dependencies, which was first proposed by L. Tesnier, and it is implemented within the context-dependent grammar [1-6]. Also, algorithms for the use of name groups, selected using partial parser [1-7], are used in the software TextAnalyst (SIC "Microsystems") and Extractor (Institute of Information Technology of the National Research Council of Canada), in particular, the latter is used in the search engine Journal of Artificial Intelligence Research. Among the systems developed in Ukraine, it is worth mentioning the development of the Department of Mathematical Informatics at Taras Shevchenko National University of Kyiv is a system of text processing in natural language. The system is designed to solve problems such as analysis and synthesis of texts in natural language, automated generation of abstract text, automated indexing (definition of the subject) of the text. The most important technical solution in the system is the ability to "weigh" the vertices of the semantic web of text. The most important network vertices are the vertices that have the highest number of connections with others [42-47]. This procedure can be used to construct an image of the abstract by weighing the vertices and rejecting the lightest - "marginal".

2 Formulation of the problem in general

Obviously, it takes a lot of time and resources to manually build a complete related ontology for a specific subject area. The reason for this cost is that such ontologies must contain tens of thousands of elements in order to be capable of solving the wide range of applications that arise in this software. Therefore, the manual construction of ontology by a human operator is a lengthy routine process that, in addition, requires a thorough knowledge of the subject area and an understanding of the principles of ontology construction.

Therefore, we will build mathematical support for the automation of ontology construction, and more precisely its construction. Because we believe that the basic terms and the relationship between them should be entered manually by an expert person into the ontology. We call this ontology a basic one and denote it $O_{base} = \langle C_b, R_b, F_b \rangle$. That is, the construction of ontology begins from the moment when it already has some data. Therefore, we will call this process the development of a basic ontology and denote: $\gamma : O_{base} \to O$. In order to build ontologies that adequately describe semantic software models, it is necessary, first of all, to solve the problems of obtaining knowledge from different sources in order to identify many concepts and establish a hierarchy on that set. Since much of the information is contained in natural-language texts, it is promising to acquire knowledge of textual information as well as intellectually processing specially selected collections of natural-language texts.

3 Analysis of scientific results

3.1 The structure of the ontology

One of the most effective approaches to completing ontology is its automated teaching of natural texts. Automated filling can be implemented by analyzing text documents using a knowledge processor (Fig. 1).

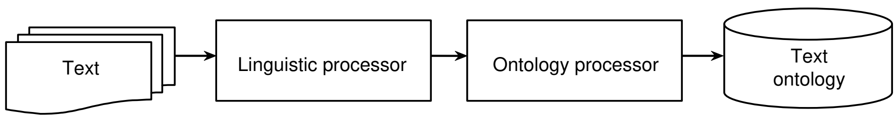

Fig. 1. Structural and functional diagram of the knowledge processor

In the presented scheme, the task of the linguistic processor is to perform its lexical, lexical, grammatical, syntactic and semantic analysis. As a result, the ontology is replenished with: concepts, threesomes (subject - action - object) and cause and effect relationships between SDO threes. The other part of the important relationships between concepts and their properties is established by the ontology processor, which builds the ontological structure for each C concept obtained after analyzing the text. The work of the ontology processor is supported by an appropriate knowledge base, the main components of which are, firstly, a set of rules, and secondly, a universal WordNet-type MMWW database [8]. The knowledge processor is used in a system of automated learning from text documents, which, in turn, is used to solve the problem of semantic search in full-text databases. As noted in [9]: ontology is the language of science. The language of science, as a structured scientific knowledge, is a multi-layered hierarchical formation in which blocks are distinguished [10]: the term system; nomenclature; means and rules of formation of conceptual apparatus and terms.

Therefore, from the point of view of the process of building an ontology, it is necessary to build its term system O_T and nomenclature O_N. In our approach, the basic ontology should accurately integrate part of the term system (Fig. 2), that is O_T and nomenclature O_N In our approach, the basic ontology should accurately integrate part of the term system (Fig. 2), that is

$$O_B \cap O_T \neq \varnothing \tag{1}$$

Encyclopedias, the terminological and explanatory dictionaries on which the software terminology system is built, are usually clearly structured and consist of dictionary entries. Therefore, it is necessary to investigate their possible structures in order to recognize the concepts and relationships between them.

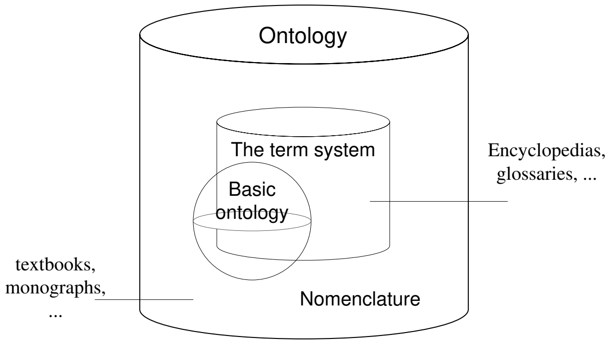

Fig. 2. Architecture of ontology

Building a nomenclature is more complicated. If terms are already highlighted in dictionaries, then in scientific texts (textbooks, monographs, etc.), they should be highlighted, search for properties of concepts and relations between concepts. Methods should be developed for obtaining terms from sources of knowledge, fixing them and dividing them into categories. Such extraction is related to natural-language processing of information. Such methods are constantly being improved and are in constant development, so their submission should be declarative, since such a representation provides the easiest way to improve it. Scientific texts are monologic. So, we conclude that technology is needed, which would allow almost in the automatic mode to create methods of native-language processing of scientific text, which already in the automatic mode would allow to build ontological models. Thus, in [9], genetic algorithms for the generation of decision models and automated programming for the automated generation of software code were used for this purpose. An overview of well-known approaches and projects for automated ontology construction is given in [11].

3.2 Features of automated ontology construction

The starting point when creating any model of knowledge about software is the choice of its categorical apparatus. For any abstract systems that use the same thesaurus or dictionary, there is no guarantee that they will be able to use the same information correctly until a single conceptualization is adopted. The conceptualization is based on the category of abstractions that are associated with the construction of the term that underlies any ontology. We substantiate the construction of the term construction as a sign of the semiotic system. Today, there is no single correct way to model software. However, there are some fundamental rules for the development of ontology:

- Effective resolution always depends on the proposed program and the expected extensions;
- Ontology development is a must-have iterative process;
- Concepts in ontology must be close to objects and relationships in software.

The paper uses a categorical apparatus derived from the work of linguists, logicians, and computer scientists [12, 13]. The definition of the categorical apparatus is connected, on the one hand, with the identification of conceptual objects of objective reality and relations between them, on the other, with their presentation. Indeed, one of the inter predations of the language of scientific texts has to do with understanding it as a sign system: the language of mathematics, chemistry, that is, the artificial symbolic languages produced in different sciences. They have artificial vocabulary and syntax. These languages are included in the scientific text, thus forming part of the language of science and making it a kind of education. First of all, let's describe the basic concepts that will be used in the future. The term is a sign of a special semiotic system that has a nominative and definitive function. Nominative - because the term refers to, denotes a whole complex semantic fragment from the general system of intensities (contents) constructed. Definitive - because it replaces a definition that has an explicit and / or implicit appearance from a range of utterances and understands that definition in its use, being a minor factor in relation to it. The specificity of terminology lies in the awareness of the content of the signs of the language of science, that is, in the ability of the speaker to explicate the definition of the term used. That is, the term is a sign of a special semiotic system, which is the minimum carrier of scientific knowledge, the short name of a concept that has a definition.

Definition is the union of forms of structural and substantive definitions, in which structural information implies the representation of the most probable niche substantive and from the substantive information the most probable structural interactions of elements of the field of the terms system, so that together these two aspects provide a representation of its integrity and functional validity. This means that it is necessary to have as a substantive definition of the term a verbal definition of the term, and as a structural definition a fragment of a network of signs.

The referent is the representation of the denotation of real world entities (object, phenomenon, process), knowledge described in the sign system. A concept is the knowledge that is expressed in this concept in the conceptual modeling of software.

Intensive is the content of a concept that corresponds to the structural definition and is described as an internal form of the concept that combines its lexis and logo and sufficiently to define the extension. Extension is a concept scope.

Conceptual objects are divided as follows:

- Essence is tangible and intangible objects, ways of considering them;
- Property is quantitative, qualitative, relational (ratio);
- Action is operations, processes, states;
- Quantities are time, space…

Conceptual relationships:

- Quantitative (coinciding with the theoretical-multiple relations of identity, inclusion, deletion, intersection, union);
- Qualitative (hierarchical and functional).

In the AI industry, the real world is considered to be objects. Objects can be made up of parts. Objects have properties that matter. Objects can be different in relation to each other. Properties and relationships change over time. At different points in time, events occur that trigger the processes in which objects are involved and change over time. Events can trigger other events, that is, have an effect. The world and its objects can be in different states.

3.3 Generalized scheme of processing monologic texts

Methods of construction of ontologies can be divided into groups. The first group will include traditional methods of natural language processing, and the second will include methods that relate directly to ontology construction. Consider the technology of analysis of naturalistic text and construction on their basis ontology, proposed in [14]. A modified layout of the general scheme is shown in Fig. 3.

The implementation methods of the first four blocks are considered to be the most elaborated. However, it should be noted that the studies are ongoing because a satisfactory result of their work has not yet been obtained. Yes, we suggest using language ontology to perform relevant analyzes. The pre-processing function includes lexical analysis, splitting of complex sentences into simple sentences, division of sections, sub-sections, sentences in the source text, verification of compliance with accepted restrictions. At this point, the research of scientific texts is considered inadmissible as complex sub-sentences that combine recursively-embedded meaning sentences.

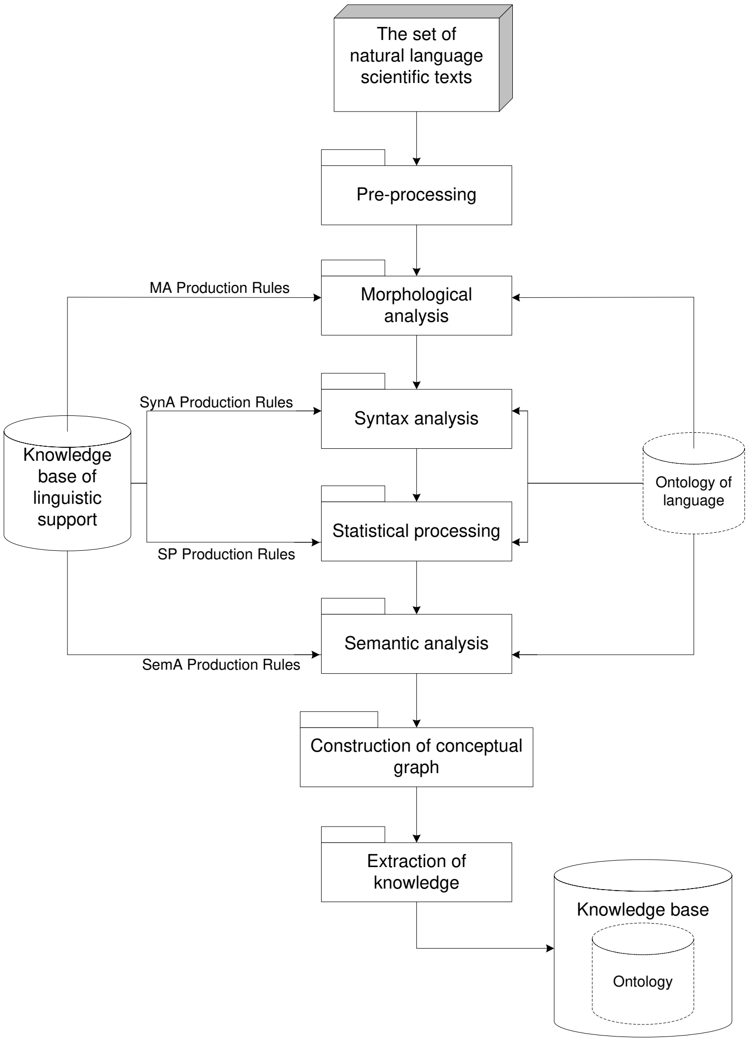

Fig. 3. General scheme of automated ontology construction

The main task of lexical analysis is to split the input text of a document, which is a sequence of single characters, into a sequence of tokens. From this point of view, all characters in the input sequence are divided into characters that belong to any tokens, and characters that separate the light-seven (delimiters). In some cases, the tokens may not be delimited. As a result of lexical analysis, a set of tokens is formed $L = \{l_i \mid i=1, \ldots, k$, k is the number of tokens in the text$\}$. Each token is assigned a vector:

$$\rho_i = \left\langle p_i, n_i^l, n_i^s, n_i^p, n_i^d, n_i^c \right\rangle \quad (2)$$

where p_i is unique token vector number; n_i^l is the sequence number of the tokens in the sentence; n_i^s is the order number of the sentence in the text; n_i^p is paragraph number; n_i^d is section number; n_i^c is chapter number.

The main function of verbal morphological analysis is to determine the language part of the token l_i and assigning it a vector of morphological information ρ_i .The analysis of lexemes uses dictionaries of endings, a dictionary of inflectional classes, a dictionary of ready word forms and a dictionary of basics, tables of compatibility of the basics of a inflected class and vectors of morphological information.

3.4 Syntax analysis

All syntactic units of a naturalistic sentence must be uniquely identified during parsing. Syntax units are constructs in which their elements (components) are joined by syntactic links and relations. Syntax is an expression of the interrelation of elements in a syntactic unit, that is, it shows the syntactic relationships between words, creates a syntactic structure of the sentence and phrase, as well as the conditions for realizing the lexical meaning of the word. There is usually only one kind of syntax involved - subordination. This kind of syntax communicates the relation between the facts of the objective world in the form of a combination of two words, in which one acts as the main and the other as the dependent. Relationships between tokens are represented as lexical-grammatical links between words, which are questions from the main word to the dependent (for example, the operating system). The input to syntax is the results of morphological analysis, presented in the form of multiple pairs $\langle l_i, \rho_i \rangle$, where l_i is the token, ρ_i is a vector of morphological information token l_i As a result of parsing, a graph of dependences G is formed at the vertices of which contains tokens. The vertices are joined by arcs that indicate the direction of the link from the parent word to the dependent one.

Statistical text processing is not required for every natural language processing system. It is usually available in search engines and automated abstracting systems. These systems are discussed in more detail in the section. Statistical methods are based on the frequency characteristics of text: the frequency of occurrence of words in text, the frequency of co-occurrence of several words, the weighted frequency of occurrence, etc. In these methods, the relations between words are not analyzed linguistically. During the statistical analysis, the search for input word sequences is performed and the concepts by which words and phrases are understood, as well as their frequency characteristics are defined. It is especially important to find the substantive noun phrases that are given by the schema: consistent word + noun [15].

3.5 Paraphrasing semantic analysis

The purpose of semantic analysis is to define for each word and phrase some substantive characterization. The content of the phrase is usually presented as a fragment of the semantic network. The basis for constructing a fragment is a graph of dependencies. The result of semantic analysis is the transformation of the dependency graph into a fragment of the semantic network. The construction of a concept graph combines two components: the construction of a single semantic network and the extraction of pragmatic information, that is, the analyzed text extracts its pragmatic content. Link Grammar Parser (http://www.link.cs.cmu.edu/link) software was used to construct the conceptual graph of the scientific text. The linguistic support knowledge base consists of three parts: a fact base, a rule base, and a software knowledge base. The fact sheet contains a dictionary of ready-made word forms, a dictionary of endings, a dictionary of inflectional classes and a dictionary of basics. The rule base consists of the production rules of lexical, morphological, syntactic, statistical and semantic analyzes. A complete description of all methods is given in [16, 17].

3.6 Methods of ontology construction

Ontology describes the concept of particular software and the relationship between them. In this sense, knowledge becomes possible for re-use by people, databases and software systems. In addition, the efficiency of both intellectual systems and traditional information systems is greatly enhanced [18]. This determines the relevance of creating ontologies. Currently, quite a few systems have been developed that allow the creation of ontologies in dialog mode. However, this process is characterized by considerable complexity. Therefore, knowledge of the concept must be obtained from full-text sources of knowledge and automatically build ontologies. For example, to create a terminology system that is the nucleus of software ontology, knowledge can be obtained from terminological and interpretative dictionaries [19]. Projections of the terms system on specific fields of knowledge (task, type of activity) are called nomenclatures [20]. To build a nomenclature, knowledge can be extracted from scientific and educational publications.

4 Building a subject domain system

The possibility of automated construction of software ontology is ensured by the acquisition of knowledge of qualitative terminological and / or interpretative dictionaries. In addition, the terminology based on vocabulary knowledge is the nucleus of software ontology. The final version of the ontology should be created with the help of combining several ontology kernels based on different terminology dictionaries. It should be noted that the terminology dictionaries used to create the software ontology should be selected by a knowledge expert. To build complete software ontology, it is necessary to build software nomenclatures that are built on the basis of knowledge from such scientific texts as monographs, textbooks, articles and more. Then you need to combine the terminology and nomenclature.

4.1 Terminological dictionaries as sources of knowledge

There are several classifications (typologies) of dictionaries. The type of any dictionary is determined by the nature of the lexical material and the practical meaning [20]. Yes, encyclopedic (from Greek *enkyklios paideia* is learning from a whole range of knowledge) dictionaries contain extra-linguistic information about the language units described. These dictionaries contain information about scientific concepts, terms, historical events, persons, geographies, and more. The encyclopedic dictionary has no grammatical information about the word, and depending on the volume and destination of the dictionary, more or less detailed scientific information about the subject matter is defined. The object of describing linguistic (language) dictionaries is language units - words, word forms, morphs. In such a dictionary the word can be characterized in various aspects depending on the purposes, volume and tasks of the dictionary: in terms of content, word formation, orthography, orthopedic, corrects usage.

In addition, vocabulary selection vocabularies are distinguished: the dictionaries of the non-vocabulary type and the dictionaries in which the vocabulary is selected according to certain parameters. For example, the scope distinguishes colloquial, voluminous, dialectical, terminological dictionaries. Historically are dictionaries of archaisms, historicisms, neologisms, and more. In terms of disclosure of certain aspects (parameters) words in dictionaries can be - etymological, grammatical, spelling, etc. In terms of revealing the systemic relationships between words, they distinguish nested, word-forming, homonymous, paronymic (expression plan), synonymous, antonymic (content plan) dictionaries. Let us consider dictionaries in terms of their use as a source of knowledge. To build subject ontology requires only qualitative words, containing not only the definition of the term, but also a description of the properties, relations, synonyms and other elements of knowledge about the term. Therefore, it is better to use dictionaries that provide more or less comprehensive scientific information about the subject. Such information is found in encyclopedic, explanatory, terminological dictionaries. Any dictionary consists of dictionary entries. Dictionaries differ in the structure of dictionary articles. Most dictionaries do not have a clear structure for dictionary articles. As a rule, the dictionary article gives one or more definitions (definitions) of concepts, and then describes the relation of the term with other concepts, by which I can explain the essence of the term. These relationships can be generic, part-whole, set a metric for the term, describe the properties of the term, and determine the processes that occur with or over the term something.

The dictionary article of any dictionary starts with the title word, which is the name of a term or terminological phrase. The title word can be typed in capital letters, bold or otherwise. The headline is followed by a text that explains the headline in the dictionary and describes its main characteristics. According to the degree of structure of the text it is possible to distinguish dictionaries that have a strict structure of dictionary articles. So a clear structure of the dictionary article has a terminology dictionary on the basics of computer science and computer engineering [21], the dictionary article of which has four parts:

- A title section containing the title of the dictionary article and the definition of the term;

- The part that discloses the relationship of the term or terminology to other words in the sentence or text;
- The illustrative part demonstrates the actual use of the term or terminological phrase;
- The help section reveals the origin of the term or terminological phrase.

Each part of the dictionary article can be distinguished by structural elements, which have a certain order of following. Yes, the title starts with the title of the dictionary article. For terminology, the headline phrase indicates its abbreviation: for example, information technology (IT). In the title part of the word-terms, the heading follows the grammatical characteristic: the endings of the generic singular, the full plural form, the end of the plural, and an indication of the genus of the noun are given. Then an expanded interpretation that reveals the structure of the concept of the term and its constituent parts gives its English and German correspondence. The individual meanings of multivolume terms are indicated by the ordinal number, followed by their interpretation. Therefore, terminology dictionaries contain the terminology of one or more specific fields of knowledge or activity used in the modern world. They give the basic concepts without which it is difficult to do in a specific activity, and quite detailed explanations. The structure of vocabulary articles of terminological dictionaries is different and is developed by the compilers for each dictionary. The degree of structure of the content of the dictionary article, the order of following its structural elements, completeness of presentation depend on the purpose of the dictionary, the specifics of the field of knowledge.

4.2 Construction of a semantic network of sign-frames as a model of the term system representation

Interpretation of the sign "concept" t is the centerpiece of the knowledge representation model and is identified with the elemental fragment Φ SF software semantic network: $t \xrightarrow{def} \Phi$, where Φ is semantic network character frame. Since each vertex of such a frame-sign is a vector or a set, it is revealed by a bundle of components of a vector or set, which in turn can also be revealed by a bundle of components. Therefore, a single semantic network of SF character frames will be built.
Construction of semantic network of sign-frames, analysis of the constructed network, integration of networks is carried out by means of methods, which are defined in the form of product systems. Initially, products that reveal many of the title terms are activated $T = \{t_i\}$. Capitalized terms t_i frames of prototype frames of conceptual object "Concepts" are filled, as a result a lot of exoframes is formed $\{\Phi_{Vi} \mid i = 1,...|T|\}$. So, initially we will have many isolated frames $\{\Phi_{Vi}\}$, which, as the slots of the prototype frame are filled into one network:

$$SF = \bigcup_{i=1}^{n} \Phi_{Vi} \tag{3}$$

where n is the number of software terms; Φ_{Vi} is frames that describe all conceptual objects of the software. Therefore, the basic procedure for building a network is a consistent analysis of each glossary of the terminology dictionary, which consists of the following recognition processes: definitions; quantitative relationships; qualitative attitude.

4.3 Recognizing multiple definitions of a term

The following situations arise when mining a definition:

1. A dictionary article may have one or more definitions;
2. If there is one definition in the article, it starts after the first symbol '-', which is encountered in the dictionary article;
3. If there are several definitions in the article, they can be numbered either in Arabic numerals or in the Latin alphabet;
4. In the case of numbering definitions, either the number '.' Or the symbol ')' may be after the number.

Therefore, the beginning of the definition is indicated by the '-' symbol or the appearance symbol "#.", "#)". Here, the '#' symbol indicates an Arabic numeral or Latin letter. In addition, the definition is always found in the first sentence of the dictionary article. This means that to determine the definition, it is necessary to determine whether the first sentence of a dictionary article contains these features.

It should be noted that the direction of research of the automated on (development) structure of ontologies, BR with the help of natural-language texts and systems on their basis is actively developing. In particular, the annual European Conference on Artificial Intelligence holds a meeting of a separate section on ontology training, at which it considers advances in the field of their automated formation.

5 Conclusions and prospects for further scientific research

Thus, the analysis of the state of research and development in the field of extraction of knowledge of natural texts is conducted. The general algorithm, the necessary methods and means for extracting new knowledge from the natural text are proposed, it is shown that such algorithm should be multi-stage and include a hierarchical multi-level procedure for recognizing the concepts, connections, predicates and rules that result in ontology from the method performing the recalculation of expected utility.

An ontology development method is built, which is based on the use of an existing ontology in the analysis of text documents used in the construction of the nomenclature and ontology term system. Relationships have been classified and appropriate templates have been developed to search for them in natural language texts. All this made it possible to automate the process of ontology development, which means a significant reduction in costs.

References

1. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. In: Cambridge University Press. (2007)
2. Lytvyn, V., Sharonova, N., Hamon, T., Cherednichenko, O., Grabar, N., Kowalska-Styczen, A., Vysotska, V.: Preface: Computational Linguistics and Intelligent Systems (COLINS-2019). In: CEUR Workshop Proceedings, Vol-2362. (2019)
3. Vysotska, V., Lytvyn, V., Burov, Y., Berezin, P., Emmerich, M., Basto Fernandes V.: Development of Information System for Textual Content Categorizing Based on Ontology. In: CEUR Workshop Proceedings, Vol- 2362, 53-70. (2019)
4. Shanjian L., Katsuhiko, M.: A composite approach to language. In: encoding detection. https://www-archive.mozilla.org/projects/intl/universalcharsetdetection. (2002)
5. Emmerich, M., Lytvyn, V., Yevseyeva, I., Fernandes, V. B., Dosyn, D., Vysotska, V.: Preface: Modern Machine Learning Technologies and Data Science (MoMLeT&DS-2019). In: CEUR Workshop Proceedings, Vol-2386. (2019)
6. Lytvyn, V., Vysotska, V., Demchuk, A., Demkiv, I., Ukhanska, O., Hladun, V., Kovalchuk, R., Petruchenko, O., Dzyubyk, L., Sokulska, N.: Design of the architecture of an intelligent system for distributing commercial content in the internet space based on SEO-technologies, neural networks, and Machine Learning. In: Eastern-European Journal of Enterprise Technologies, 2(2-98), 15-34. (2019)
7. Bisikalo, O., Ivanov, Y., Sholota, V.: Modeling the Phenomenological Concepts for Figurative Processing of Natural-Language Constructions. In: CEUR Workshop Proceedings, Vol- 2362, 1-11. (2019)
8. Miller, G.A.: WORDNET: A lexical database for English. In: Communications of ACM, 11, 39–41. (1995)
9. Burov, Y., Vysotska, V., Kravets, P.: Ontological Approach to Plot Analysis and Modeling. In: CEUR Workshop Proceedings, Vol- 2362, 22-31. (2019)
10. Chyrun, L., Gozhyj, A., Yevseyeva, I., Dosyn, D., Tyhonov, V., Zakharchuk, M.: Web Content Monitoring System Development. In: CEUR Workshop Proceedings, Vol- 2362, 126-142. (2019)
11. Babichev, S., Gozhyj, A., Kornelyuk A., Litvinenko, V.: Objective clustering inductive technology of gene expression profiles based on SOTA clustering algorithm. In: Biopolymers and Cell, 33(5), 379–392. (2017)
12. Babichev, S., Taif, M.A., Lytvynenko, V., Osypenko, V.: Criterial analysis of gene expression sequences to create the objective clustering inductive technology. In: 2017 IEEE 37th International Conference on Electronics and Nanotechnology, 244-248. (2017)
13. Rusyn, B., Lytvyn, V., Vysotska, V., Emmerich, M., Pohreliuk, L.: The Virtual Library System Design and Development, Advances in Intelligent Systems and Computing, 871, 328-349 (2019)
14. Kravets, P.: The control agent with fuzzy logic. In: Perspective Technologies and Methods in MEMS Design, MEMSTECH'2010, 40-41 (2010)
15. Rusyn, B., Vysotska, V., Pohreliuk, L.: Model and architecture for virtual library information system. In: Computer Sciences and Information Technologies, CSIT, 37-41 (2018)
16. Vysotska, V., Burov, Y., Lytvyn, V., Demchuk, A.: Defining Author's Style for Plagiarism Detection in Academic Environment, Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing, DSMP 2018, 128-133 (2018)
17. Vysotska, V., Lytvyn, V., Burov, Y., Gozhyj, A., Makara, S.: The consolidated information web-resource about pharmacy networks in city, CEUR Workshop Proceedings, 239-255 (2018)

18. Lytvyn, V., Vysotska, V., Dosyn, D., Lozynska, O., Oborska, O.: Methods of Building Intelligent Decision Support Systems Based on Adaptive Ontology, Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing, DSMP 2018, 145-150 (2018)
19. Buchheit, M, Donini, F.M., Schaerf, A.: Decidable Reasoning in Terminological Knowledge Representation Systems. In: Journal of Artificial Intelligence Research, 1, 109-138. (1993)
20. Lytvyn, V., Vysotska, V., Rusyn, B., Pohreliuk, L., Berezin, P., Naum O.: Textual Content Categorizing Technology Development Based on Ontology. In: CEUR Workshop Proceedings, Vol-2386, 234-254. (2019)
21. Demchuk, A., Lytvyn, V., Vysotska, V., Dilai, M.: Methods and Means of Web Content Personalization for Commercial Information Products Distribution. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 332–347. (2020)
22. Lypak, H.,.Rzheuskyi, A., Kunanets, N., Pasichnyk, V: Formation of a consolidated information resource by means of cloud technologies. In: International Scientific-Practical Conference on Problems of Infocommunications Science and Technology. (2018)
23. Rzheuskyi, A., Kunanets, N., Stakhiv, M.: Recommendation System: Virtual Reference. In: 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), 203-206. (2018)
24. Kaminskyi, R., Kunanets, N., Rzheuskyi, A.: Mathematical support for statistical research based on informational technologies. CEUR Workshop Proceedings,2105, 449-452 (2018).
25. Kazarian, A., Kunanets, N., Pasichnyk, V., Veretennikova, N., Rzheuskyi, A., Leheza, A., Kunanets, O.: Complex Information E-Science System Architecture based on Cloud Computing Model. In: CEUR Workshop Proceedings, Vol- 2362, 366-377. (2019)
26. Veres, O., Rishnyak, I., Rishniak, H.: Application of Methods of Machine Learning for the Recognition of Mathematical Expressions. In: CEUR Workshop Proceedings, Vol- 2362, 378-389. (2019)
27. Basyuk, T.: The main reasons of attendance falling of internet resource. In: Proc. of the X-th Int. Conf. Computer Science and Information Technologies, CSIT'2015, 91-93. (2015).
28. Su, J., Sachenko, A., Lytvyn, V., Vysotska, V., Dosyn, D.: Model of Touristic Information Resources Integration According to User Needs. In: International Scientific and Technical Conference on Computer Sciences and Information Technologies, 113-116 (2018)
29. Lytvyn, V., Vysotska, V., Kuchkovskiy, V., Bobyk, I., Malanchuk, O., Ryshkovets, Y., Pelekh, I., Brodyak, O., Bobrivetc, V., Panasyuk, V.: Development of the system to integrate and generate content considering the cryptocurrent needs of users. In: Eastern-European Journal of Enterprise Technologies 1(2-97), 18-39 (2019)
30. Lytvyn, V., Kuchkovskiy, V., Vysotska, V., Markiv, O., Pabyrivskyy, V.: Architecture of system for content integration and formation based on cryptographic consumer needs. In: Computer Sciences and Information Technologies, CSIT, 391-395 (2018)
31. Gozhyj, A., Chyrun, L., Kowalska-Styczen, A., Lozynska, O.: Uniform Method of Operative Content Management in Web Systems. In: CEUR Workshop Proceedings (Computational linguistics and intelligent systems, 2136, 62-77. (2018)
32. Lytvyn, V., Vysotska, V., Rzheuskyi, A.: Technology for the Psychological Portraits Formation of Social Networks Users for the IT Specialists Recruitment Based on Big Five, NLP and Big Data Analysis. In: CEUR Workshop Proceedings, 2392, 147-171. (2019)
33. Vysotska, V., Burov, Y., Lytvyn, V., Oleshek, O.: Automated Monitoring of Changes in Web Resources. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 348–363. (2020)

34. Vysotska, V., Mykhailyshyn, V., Rzheuskyi, A., Semianchuk, S.: System Development for Video Stream Data Analyzing. In: In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 315-331. (2020)
35. Lytvynenko, V., Wojcik, W., Fefelov, A., Lurie, I., Savina, N., Voronenko, M. et al.: Hybrid Methods of GMDH-Neural Networks Synthesis and Training for Solving Problems of Time Series Forecasting. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 513-531. (2020)
36. Babichev, S., Durnyak, B., Pikh, I., Senkivskyy, V.: An Evaluation of the Objective Clustering Inductive Technology Effectiveness Implemented Using Density-Based and Agglomerative Hierarchical Clustering Algorithms. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 532-553. (2020)
37. Bidyuk, P., Gozhyj, A., Kalinina, I.: Probabilistic Inference Based on LS-Method Modifications in Decision Making Problems. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 422-433. (2020)
38. Chyrun, L., Chyrun, L., Kis, Y., Rybak, L.: Automated Information System for Connection to the Access Point with Encryption WPA2 Enterprise. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 389-404. (2020)
39. Kis, Y., Chyrun, L., Tsymbaliak, T., Chyrun, L.: Development of System for Managers Relationship Management with Customers. In: Lecture Notes in Computational Intelligence and Decision Making, 1020, 405-421. (2020)
40. Chyrun, L., Kowalska-Styczen, A., Burov, Y., Berko, A., Vasevych, A., Pelekh, I., Ryshkovets, Y.: Heterogeneous Data with Agreed Content Aggregation System Development. In: CEUR Workshop Proceedings, Vol-2386, 35-54. (2019)
41. Chyrun, L., Burov, Y., Rusyn, B., Pohreliuk, L., Oleshek, O., Gozhyj, ., Bobyk, I.: Web Resource Changes Monitoring System Development. In: CEUR Workshop Proceedings, Vol-2386, 255-273. (2019)
42. Chyrun, L., Gozhyj, A., Yevseyeva, I., Dosyn, D., Tyhonov, V., Zakharchuk, M.: Web Content Monitoring System Development. In: CEUR Workshop Proceedings, Vol-2362, 126-142. (2019)
43. Sachenko, S. Rippa, S., Krupka, Y.: Pre-Conditions of Ontological Approaches Application for Knowledge Management in Accounting. In: IEEE International Workshop on Antelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 605-608. (2009)
44. Sachenkom, S., Lendyuk, T., Rippa, S.: Simulation of Computer Adaptive Learning and Improved Algorithm of Pyramidal Testing. In: International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), 2, 764-770. (2013)
45. Sachenko, S., Lendyuk, T., Rippa, S., Sapojnyk, G.: Fuzzy Rules for Tests Complexity Changing for Individual Learning Path Construction. In: Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 945-948. (2015)
46. Rzheuskyi, A., Gozhyj, A., Stefanchuk, A., Oborska, O., Chyrun, L., Lozynska, O., Mykich, K., Basyuk, T.: Development of Mobile Application for Choreographic Productions Creation and Visualization. In: CEUR Workshop Proceedings, 2386, 340-358. (2019)
47. Mukalov, P., Zelinskyi, O., Levkovych, R., Tarnavskyi, P., Pylyp, A., Shakhovska, N.: Development of System for Auto-Tagging Articles, Based on Neural Network. In: CEUR Workshop Proceedings, Vol-2362, 106-115. (2019)