

Linguistic Processor Integration for Solving Planimetric Problems

Sergeyi S. Kurbatov, Scientific Research Centre for Electronic Computer Technology, Russia

ABSTRACT

The research deals with the original algorithms of the linguistic processor integration for solving planimetric problems. The linguistic processor translates the natural language description of the problem into a semantic representation based on the ontology that supports the axiomatics of geometry. The linguistic processor synthesizes natural-language comments to the solution and drawing objects. The method of interactive visualization of the linguistic processor functioning is proposed. The method provides a step-by-step dialog control of syntactic structure construction and its display in semantic representation. During the experiments, several dozens standard syntactic structures correctly displayed in the semantic structures of the subject area were obtained. The direction of further research related to the development of the proposed approach is outlined.

KEYWORDS

Applied Ontology, Interactive Visualization, Linguistic Processor, Semantic Representation, Solving Planimetric Problems, Syntactic Structures

INTRODUCTION

The relevance of this paper is in the lack of elaboration of a comprehensive solution to the problems in artificial intelligence (AI) systems, a low degree of integration of the results obtained in various areas of research in the AI sphere. The complexity of the integration is related to the heterogeneity of methods and the terminology that is used in a variety of ways. This work is devoted to the complex solution of the problems of natural language processing (NL), automatic problem solving based on applied ontology and interactive visualization. The emphasis in the paper is made on the linguistic processor because eventually, it is particularly the NL that describes a complicated problem in the most understandable form. Of course, full integration involves a more complete account of the sources of information and the output of the results obtained by the system (analysis and synthesis of images, audio and video sources), but the complexity of the problem requires a narrowing of the research area.

The first integrated systems appeared in the early stages of AI research. In the traditional works by Vinograd and Shank (Winograd, 1972; Schenk *et al.*, 1975; Seitova *et al.*, 2018). The first one demonstrated the understanding of NL limited by the world of simple bodies (cubes, prisms, pyramids). The program had a graphical interface that displays the results of the interpretation of the NL user's request. In the second system, after the translation of the text into the internal representation, it started

DOI: 10.4018/IJCINI.20211001.0a37

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

the process of reasoning and synthesized the NL-description of the result. The systems demonstrated both the great potential of integrated systems and significant difficulties in its implementation.

With the emergence of advanced systems of knowledge representation, ontologies, and advances in computational linguistics, it was expected that integrated systems would receive a new powerful impetus for its development. However, while these expectations are not met even with significant restrictions on the input language, the systems are fragile and difficult to adapt to the new subject areas. Nevertheless, in the application aspect, some results were achieved, namely, in the implementation of the NL interface to relational databases.

In the last two decades, the center of gravity of its processing has shifted towards the analysis of large amounts of text information from the global network and the decision-making algorithms based on cognitive methods. This is due to the constant growth of information on the Internet, which requires rapid analytical processing both in public institutions and in large commercial structures. The review (PullEnti-SDK extract named entities..., 2019) provides a list with brief characteristics of about 40 systems of linguistic analysis of unstructured texts. However, it remains of interest to the less ambitious but more advanced in terms of ideological systems, in particular, to systems and question/answer. In recent years, the following can be noted from the relevant publications (Zhang et al., 2018; Gan and Yu, 2018; Seo et al., 2015; Shi et al., 2015; Sundaram and Khemani, 2015). The overview of the systems is provided in (Morton and Qu, 2013). Their features in comparison with the described integrated system are briefly discussed in the related works section.

Theoretical Review

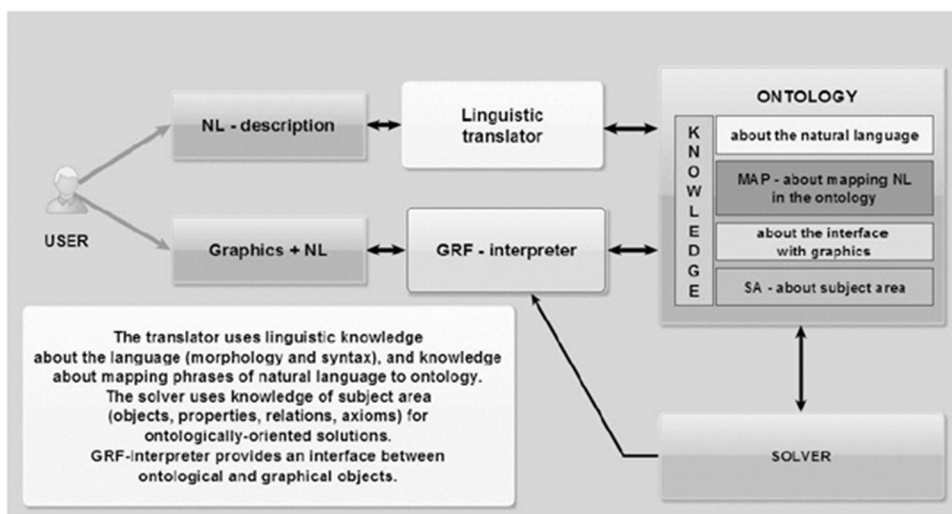
The concept of an integrated system including a natural language interface, a heuristically oriented solver and conceptual visualization is described in (Kurbatov and Vorobyev, 2016; Kurbatov *et al.*, 2018; Kurbatov *et al.*, 2019). The system assumes a qualitatively new level of integration, the mechanisms of the system provide the integration “by knowledge” as opposed to the integration “by data”. In terms of content, the system is focused on algorithmic and software implementation of the Polya methodology (Polya, 1962). It is known that the recommendations of Polya for particular cases have been successfully used by AI researchers (heuristics), but the study of the Polya method in the framework of an integrated system provides a new perspective of the computer embodiment of his ideas.

Certainly, the full implementation of the Polya methodology is extremely complex, the integrated system only assumes some approximation. Nevertheless, the questions posed by Polya (how to move from wording to formalization? how to develop a formal production? (2004)) are answered by the system in the aspects of algorithmization, the software implementation and the representation of heuristics in the knowledge base. The scheme of the integrated system is shown in Figure 1.

The main units of the system are the linguistic processor, ontology, solver and graphical interpreter. The linguistic processor according to the natural language (Russian) description of the problem forms a semantic representation, while the solver, using the knowledge of the subject area, looks for the solution. In case of success, the solution with the comments is visualized in the drawing. An example of IT is a description of a geometric problem and a fragment of a real drawing formed by the system is shown in Figure 2 (the screen-shot fragment with the translation of the task text into English).

The fragment does not reflect the controls that allow you to monitor the formation of the drawing and the justification of the actions performed step by step. The interactive visualization also provides the basis of its creation (by condition, by heuristics, by theorem) for each object of the drawing (point, line, segment, circle, etc.). The implementation of the drawing in jsxgraph (JSXGraph, 2018) allows you to get information about the object with a simple click (by mouse), as well as to change the drawing with the preservation of the task conditions. The dynamics of the drawing can only be demonstrated on the roller.

Figure 1. Scheme of the System for Solving Geometrical Tasks



Materials and Methods

Linguistic Processor

The main stages of the linguistic processor integrated system (in the analysis):

- lexical-morphological;
- syntactic;
- resolution of elliptical designs;
- synonymic transformation;
- semantic.

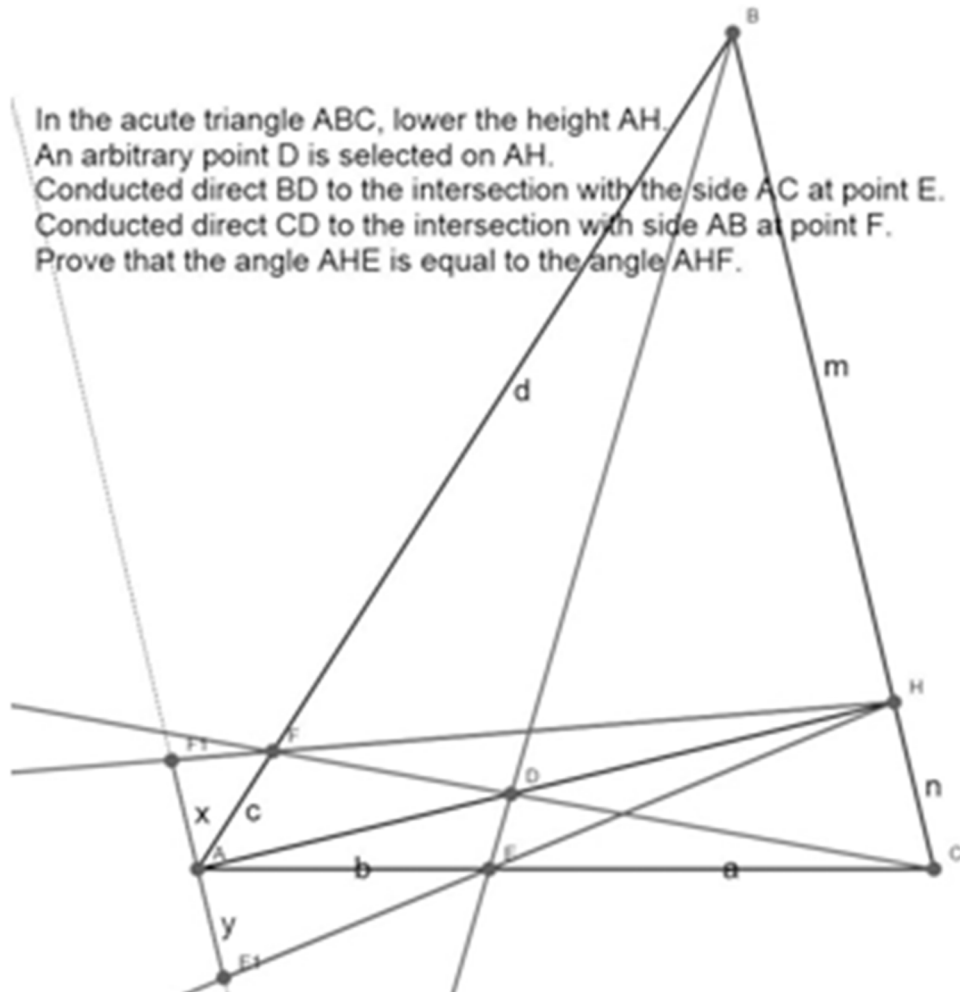
Synthesis of NL in the system is developed in a simplified version and includes the steps:

- formation of syntactic structure for all triplets of semantic structure;
- formation of text from the resulting syntactic structures.

The lexical-morphological stage is performed as standard (allocation of sentences and tokens, determination of morphological characteristics, etc.). A special feature in the selected subject area (geometry) is the identification of the basic objects when the text contains a combination of “ABC triangle”, it is usually assumed that there are three points named ‘A’, ‘B’ and ‘C’ (triangle vertices).

Syntactic analysis is already problematic, it is known that only for nominal groups the question of automatic construction of the syntactic structure has received a complete solution. The issues of processing homogeneous terms and considering semantics with the ambiguity of relations do not yet have a full solution. In the integrated system, the ambiguity at the syntax level is maintained, it is assumed that at the transition to the semantic representation it will be removed. An example of the syntactic structure for the NL-description is shown in Figure 3. The real structure is formed and visualized in Russian. The connections correspond to the interrogative words of the Russian language and bear not only syntactic but also semantic load. The interactive structure allows clicking on the object, for example, the word (“build”) or communication (“what?”) to obtain the information on the characteristics of the object and the grounds for its creation. Ontology is organized in the form of a

Figure 2. A fragment of a Screen-Shot of a Drawing Formed according to a NL-Description of the Task



structured semantic network, allowing arbitrary nesting of fragments and a variety of relationships ('element – element', 'element – fragment', 'fragment – fragment'). Both objects and ontology relationships can have an arbitrary number of properties.

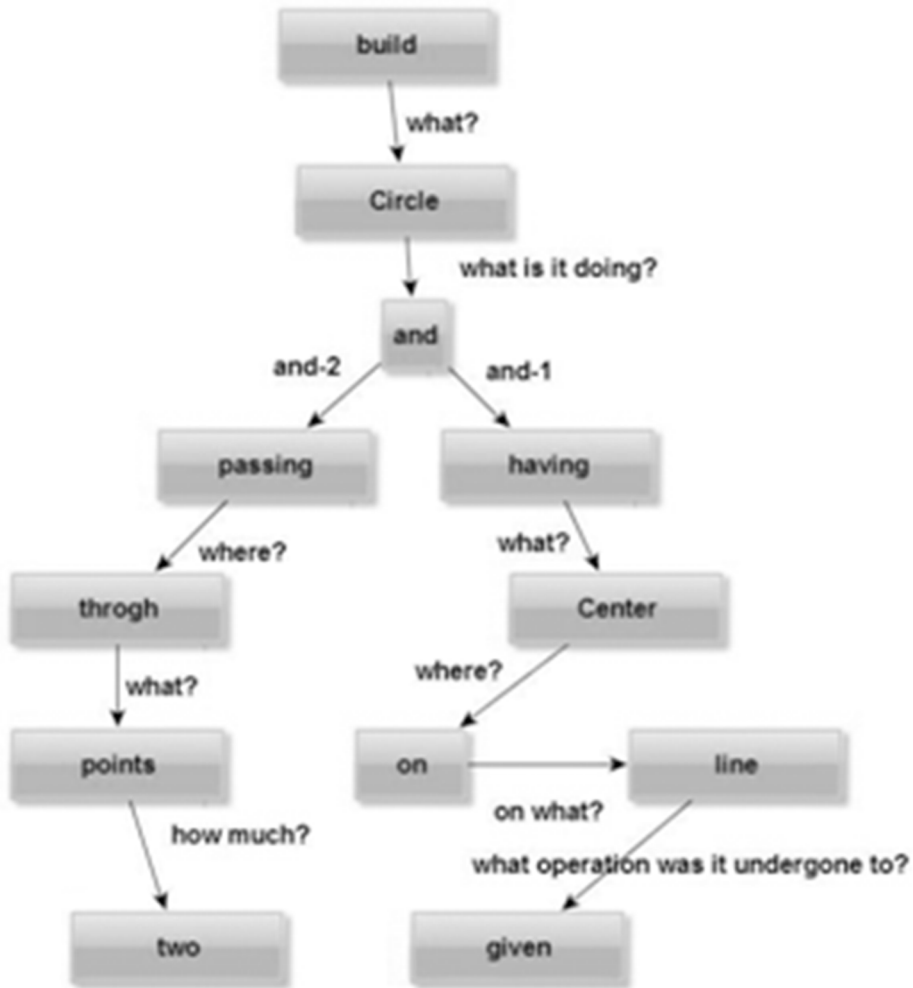
The system's capacities of processing the incomplete (elliptical) designs not only for geometry are described in (Naidenova *et al.*, 2019). At this stage, the full syntactic structure of type "the Radius of the first circle is 10 cm, and the second is 15" or "Point A lies inside the circle, while point B is outside" are restored. Cognitive models are supposed to be used to solve more complex cases (Naidenova *et al.*, 2019; Apresyan *et al.*, 1989).

Algorithms

The syntax algorithm includes many steps. The scheme of the algorithm for obtaining the syntactic structure is illustrated by the following sequence of steps:

- (To build a circle), passing through (two points) and having the center on (a given line).
- (To build a circle), passing (through (two points)) and having the center (on (a given line)).

Figure 3. Interactive Syntax for NL-Wording “To construct a circle passing through two points and having the center on a given line”



- (To build a circle), (passing (through (two points))) and having (the center (on (a given line))).
- (To build a circle), (passing (through (two points))) and (having (the center (on (a given line)))).
- (To build a circle), ((passing (through (two points))) and (having (the center (on (a given line))))).

The ellipticity resolution algorithm is based on the parallelism of syntactic structures and the use of ontology. For the above example we have as follows:

The radius of the first circle is 10 cm and the second is 15 cm.

After determining the incompleteness of the second part of the sentence, its elements are supplied in a parallel syntactic structure of the first part and generate the full second part of the sentence: The radius of the second circle is 15 cm. The word forms replaced considering the ontology are highlighted in bold italics. More details are given in (Naidenova *et al.*, 2019). The algorithm of the synonymic transformations (otherwise rephrasing) is based on the ideas of prominent Russian linguists Apresyan (1989) and Paducheva (1974). In the aspect of a fixed subject area, it is assumed that there are some standard (canonical) NL-descriptions of objects and relations in this area. These canonical descriptions

directly correspond to the semantic formalisms with which the solver can work. The solver does not work directly with syntactic structures. By means of the synonymic transformations, the syntactic structure of any NL-description should be reduced to the canonical form.

Usual transformation rules have syntactic structures at the input and output. The resulting rules are at the entrance of the canonical syntactic structure, and the output is the semantic structure of the subject area. For the geometry, the canonical description can be “point belongs to line”, and the semantic structure is the triplet “ob_type_point on ob_type_line”. Certainly, the freedom of natural language allows one canonical structure to have a set of NL-descriptions. To reduce such combinatorics in the integrated system, a method is developed that allows using a syntactic structure to obtain the text, which is further advisable to analyze using regular expressions (it is known that only regular expressions are not enough for full processing of NL).

Let us consider an example explaining the scheme of this method. The example belongs to another area (visualization of 3-d constructions in a special graphics language grasp (Lobzin *et al.*, 2015)), but in this area, the algorithms developed further in the integrated system were tested. In this area, according to the description of the NL visualized design, an example is provided below. A fragment of such a construction can be described by the phrase “The type B fittings are linked to the type B tank via points 0 and 0.”, and has the following type of the syntactic structure

the type B fittings
are linked to the type B tank
via points 0 and 0

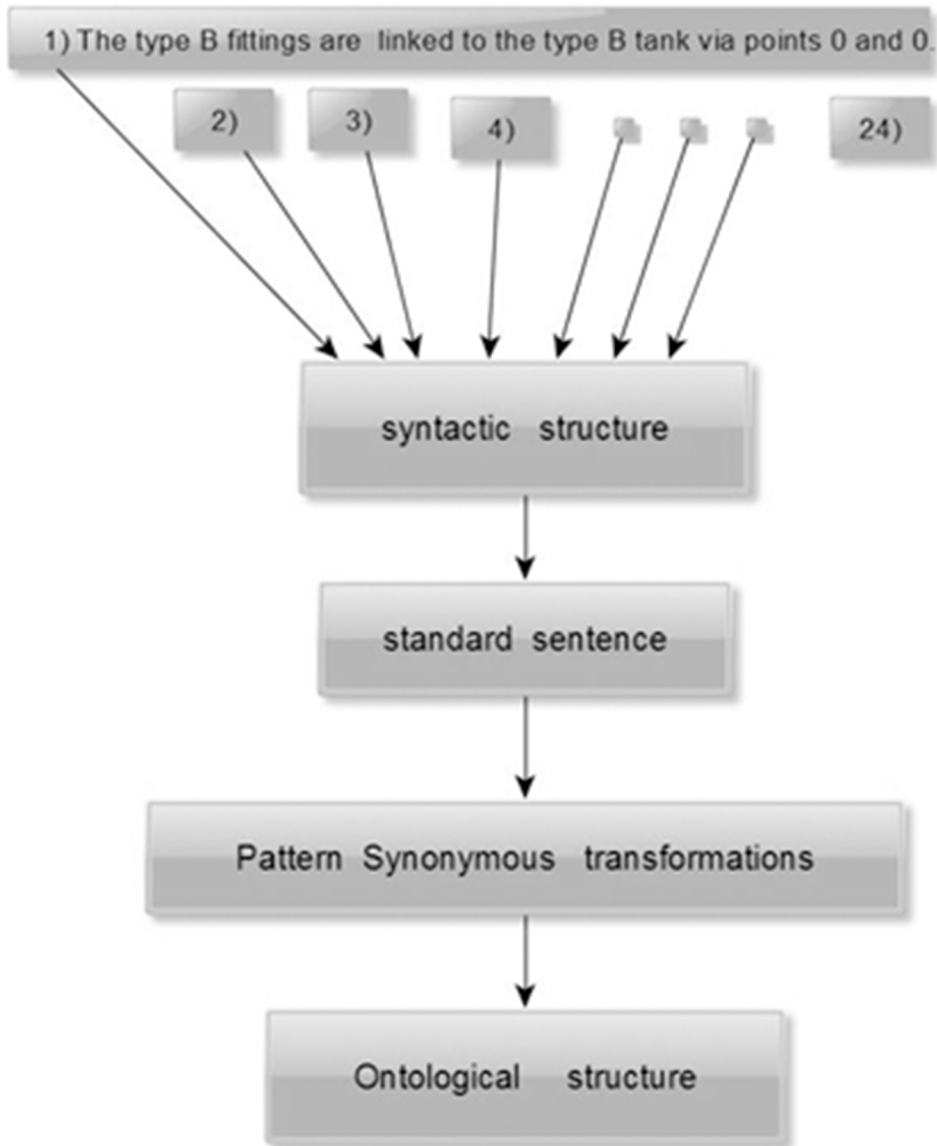
In Russian, this phrase can have 24 variants (4!), they are not stylistically pure, but understandable to an ordinary native speaker. The input can be any of these options, but the syntax is the same. Using the synthesis of this structure, the algorithm generates a standard text description. Further to this description, the template (pattern) performing its parsing styled as regular expressions is applied. The processing scheme is shown in Figure 4. Next, the semantic structure is translated into the grasp language and visualized. A fragment of ontology for this area and a rather complex visualized construction is shown in Figure 5 and 6. The construction is described by 14 triplets, the description of which is similar to the above phrase. The example demonstrates the capabilities of the linguistic processor in a proper way, but unfortunately, it does not allow interactive visualization.

The synthesis algorithm receives the input semantic structure and the initial object in this structure. By sequentially deploying this structure in triads, the algorithm generates an NL-description. To account for the direction of each triad (object-relation-object), two NL-descriptions are mapped. For example, “point belongs to line” and “line to which point belongs”. The synthesized NL-phrase is rather primitive, the real difficulties of its synthesis from the semantic representation are described in the book (Popov, 2011) and remain relevant to the present time. The NL-descriptions of heuristics and rules are prepared in advance in ontology as syntactic structures. This ensures that the syntactically presented synthesis mechanism is used (see Figure 4). The possibility of synonymic transformations of the synthesized text (reversibility of paraphrasing) should be the subject of a special study.

RESULTS AND DISCUSSION

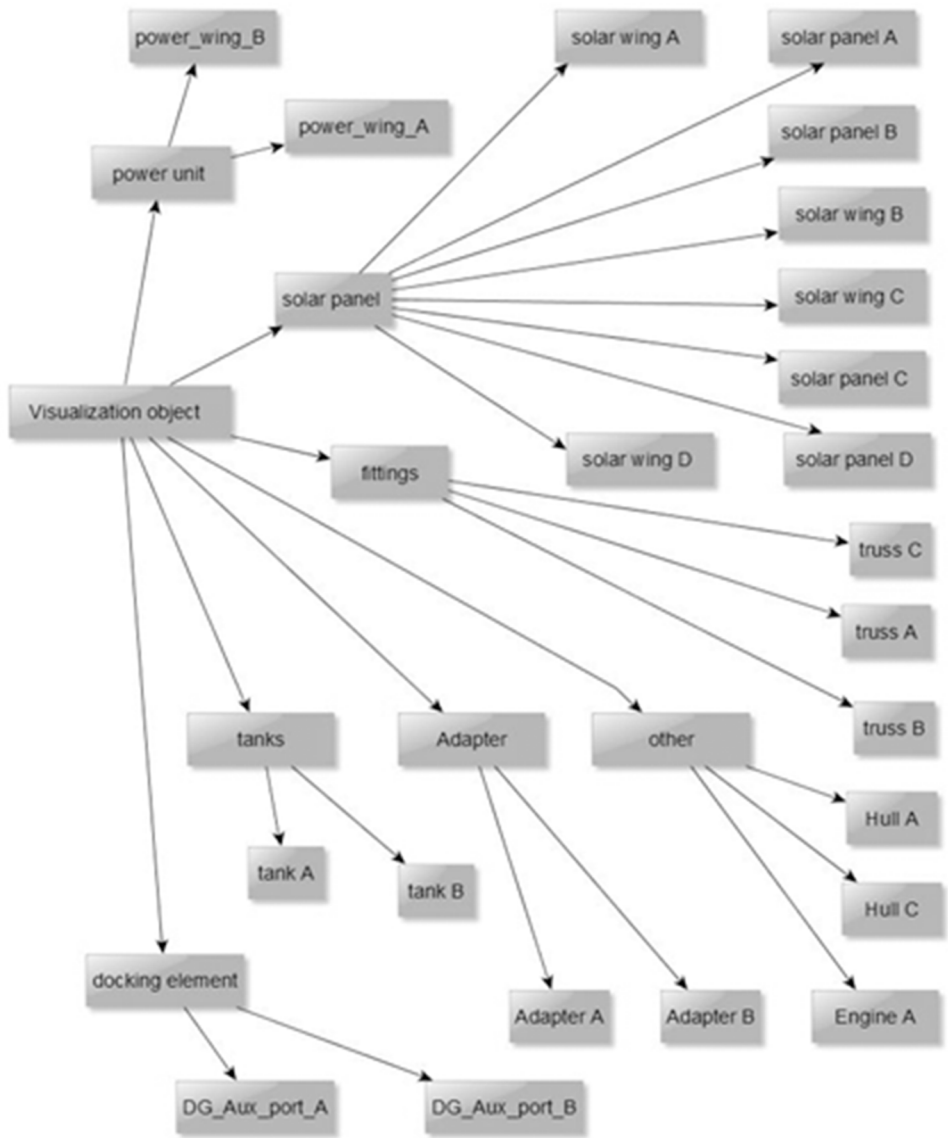
During the experiments, several dozens of standard syntactic structures correctly displayed in the semantic structures of the subject area (geometry) were obtained. The variety of structures obtained by the permutations of nominal groups are at least by an order of magnitude bigger. Elliptic (incomplete) sentences correctly processed by the system are described in (Naidenova *et al.*, 2019). The dictionary of word forms includes more than 10 thousand objects grouped into morphological and syntactic categories. The system provides an automated dictionary completion by accessing the Internet, analyzing the received answer and entering the result into the ontology (Morphological analysis..., 2019).

Figure 4. Scheme of the Transition from Syntactic Structure to the Standard Sentence and further to Synonymous Transformations to obtain Semantic Structure



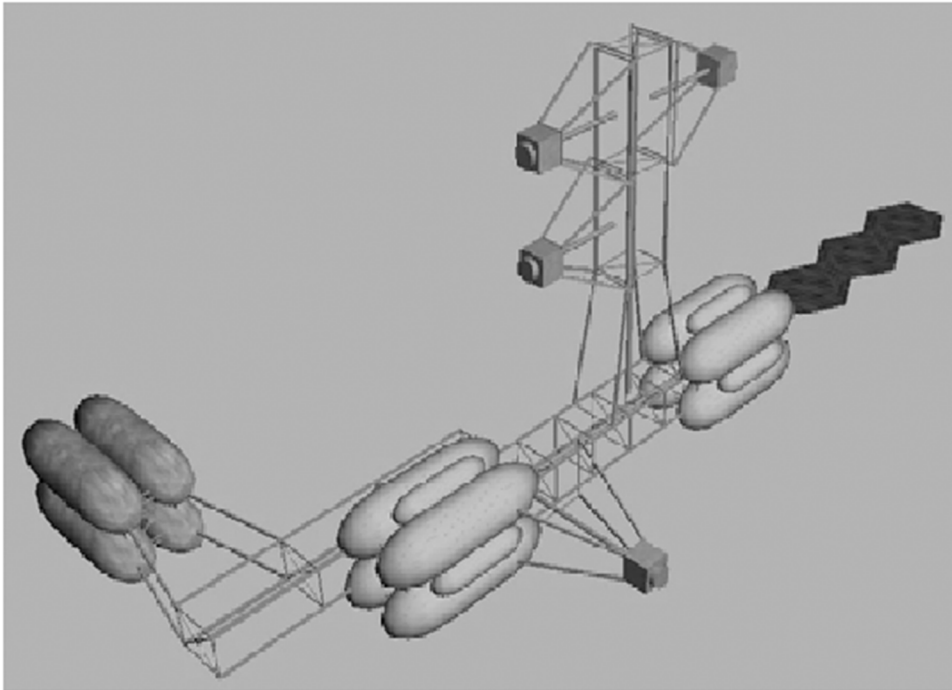
As the test, we took the individual tasks from various sources, but as the primary, we considered (Prasolov, 2006). Its materials fully cover the subject and complexity of Olympiads' tasks of all levels and all types of exams, including the exam and entrance exams to the universities. Several tasks presented in the works on the subject are discussed below in the section "discussion". Tools: the original ontology is implemented based on the DBMS Progress (Progress (DBMS), Progress OpenEdge 11.7), while the interactive visualization is implemented based on javascript. Libraries jsxgraph and MathJax (JSXGraph. 2018; MathJax Documentation..., 2018; JSXGraph Reference, 2018) were used to support graphics and mathematical formulas.

Figure 5. Fragment of Ontology for 3D Visualization Objects



Interactive visualization provides the ability to view it and get detailed information about the process of forming the syntactic structure (Figure 3). As a result, systems become transparent. This greatly simplifies the debugging of the system by declarative editing of the ontology. Software errors, of course, are only eliminated by editing the corresponding code. The test results also include texts that directly describe the drawing (not necessarily related to the task). An example of such text is as follows, “Through the point A, which is the vertex of the ABCD square, several lines are drawn. Two of them cross the CD side. The rest do not cross the sides of the square”. The description of ‘Several lines’ required entering an object of ‘set’ type with an indefinite number of elements into the ontology. A heuristic method was used to generate the drawing: random selection (with appropriate

Figure 6. Visualization of the 3D Construction synthesized from Ontology Objects in Figure 5.



parameters) of the set power. Similarly, the description of ‘others’ required the entry of the ‘set difference’ operation.

The visualization of 3-d objects by the NL-formulation (without interactivity) is described in (Lobzin *et al.*, 2015). Similar algorithms were used in linguistic processing, but the semantic structure was described in a special language (grasp). An example of visualization is shown in Figure 6. The complexity of the design can be quite large (dozens of elements). The lack of interactivity and decision rules limited the importance of the approach developed in (Lobzin *et al.*, 2015; Kurbatov *et al.*, 2011), but this study was very important for the formation of the concept of an integrated system and interactive visualization. The means of automatic proof of theorems (prover, solver) and assistants for automated proof have been developed and successfully applied for more than a dozen years; widely known are Coq (2019), HOL (2018), Isabell (2019). However, the studies of the problem of translating an NL-description of a task into a formal representation that such tools can work with are currently being originated. This is especially just for the problem of NL-dialogue, which explains the process of obtaining a solution. The complexity of the integration stages of NL-treatment and the actual decisions are usually forcing researchers to use ready-made solvers. A full explanation of the solution (in the style of the Polya methodology or at least at the level of expert systems) is significantly difficult.

In particular, large and constantly developing wolfram system (Wolfram release 11.3, 2018) has the interpreter operator, which allows entering the NL-description of the problem. However, the capabilities of the interpreter are very limited (and for the Russian language is not yet implemented, the author worked with version 11.3). From the Russian research in the field of automatic problem solving it is necessary to note the system developed in MSU for more than 20 years (Podkolzin, 2008; Podkolzin, 2016). The system 2008 included about 25 000 decision-making methods, currently – more than 40 000 methods. It is assumed that the methods of computer modeling of logical processes developed by the researchers will be successfully used for NL-processing.

The issues of interactive visualization of the NL-processing results are completely ignored in almost all the works known to us (of course, for the dynamic drawing it is not so, see e.g. (Gan and Yu, 2018)). Nevertheless, these issues are, to our mind, very important, especially in the application aspect. Advanced ontology is central to our research, including the knowledge of both the language of communication and the knowledge of the subject area. The expected level of knowledge in the ontology (integrated system) should be such that when formulating the problems for the construction of the ‘trisection of the angle’ or the ‘quadrature of the circle’, the system is not engaged in a purposeless search for a solution. Ideally, the ontology should explain to the user that these tasks can be solved with the use of the tools different from the compass and ruler: neusis and quadratrix. In general, the described integrated system is quantitatively inferior (body of texts, solved problems) to most of the listed works, but, in our opinion, it is more advanced in the ideological, fundamental direction.

In the reviewed work (Morton, 2013), it is rightly noted that the problem of NL-interface and solvers has great potential for development. It is also noted that large data sets for solvers will be useful both for testing existing systems and for promising developments using machine learning methods (especially, deep learning). The impressive success of AlphaGo (Silver *et al.*, 2016) and AlphaZero (Silver *et al.*, 2017) showed that it is possible to achieve the level of the champion (in go and chess) with the help of machine learning. This does not require a teacher, and that is especially striking when training uses only primitive knowledge (for chess – only the rules). More importantly, it avoids specialization. Of course, there are still difficulties in interpreting the resulting solutions related to the specifics of neural networks. However, the prospects of using deep learning for its processing problems and solvers look very promising. The question of presenting the Polya methodology in neural networks remains open.

Generally, the test results confirmed the prospects of the integrated system concept and the chosen research direction. Let us consider several interesting moments in testing on the problems described in some works in detail. The (Seo *et al.*, 2015) straight-line testing of the NL-description of the problem revealed its insufficiency (it is not indicated that point E is the intersection of diameters and chords). But this was to be expected because the work uses a joint text and graphic description.

The problem “Let D be a point inside the acute triangle ABC so that $\angle ADB = \angle ACB + \pi/2$ and $AC \cdot BD = AD \cdot BC$. Calculate the ratio $(AB \cdot CD)/(AC \cdot BD)$ is provided in ().” According to this NL-description, a LISP-like structure is formed, which goes further to the input of the solver. In (Solving Natural Language Math Problem, 2019) the solution is not provided, namely, it is the most difficult. Since the NL-comment to the solution is included in the tasks of the linguistic processor of the integrated system, the work of its algorithms was considered in this example. It should be noted that the solution of the problem provided in the Internet video (IMO 1993 Problem 2) from the position of the Polya methodology is beneath criticism. The integrated system did not bring the solution to the end, but the approach outlined by the algorithm is purposeful. The algorithm proposed to construct a triangle similar to this one and including the segment CD. Heuristic motivation: CD is included in the expression to be evaluated. This expression is a proportion, and similar triangles are often used to calculate proportions.

In the problem from paper (Wang and Su, 2015), it was required to prove the equality of segments under the provided conditions, but without resorting to trigonometry. The algorithm found the solution given in (Wang and Su, 2015), but as the second one. The first decision was to direct the calculation of one of the segments by using the Pythagorean theorem. The solution is less elegant than purely geometric, but also quite short (algebraic) and most importantly it is also purposeful. The NL-comments to the heuristics used gave the justification taken by the algorithm solutions. The details of the operation of the solver are beyond the scope of this paper. For the task “to construct a right triangle by the hypotenuse and the bisector of the right angle”, the linguistic processor forms two different semantic representations. The choice of one of them is carried out by means of heuristics (“the choice of the minimum number of unknowns is preferable”) which has a rather general character.

Interactive visualization of the syntactic structure (Figure 3) provides not only a justification for the creation of elements of the structure but also allows you to follow the steps of its formation. In the text form, this process is described at the beginning of the “algorithms” section. Interactive visualization of the drawing (Figure 2) also provides a step-by-step view of the drawing formation and the NL-comment steps. There is the following comment example, “ABC triangle is constructed; justification: by the condition of the problem” or “to apply the Cheva theorem (Ceva); justification: three segments in a triangle intersecting at one point”. In general, the considered papers differing in linguistic models, the body of texts and types of tasks, are focused on the number of tasks, highly simplified text of the problem and on the solution itself. At the same time, the depth of linguistic study and explanatory capabilities of the system recede into the background. But this is what the integrated system focuses on and, together with the interactive visualization, this is its main difference from the systems under consideration.

CONCLUSION

The main contribution of this paper to the fundamental problem of understanding natural language is as follows:

- the original algorithms of the linguistic processor of the integrated system for solving planimetric problems have been developed; the developed algorithms successfully combine the method of synonymic transformations with the extended method of processing regular expressions; in addition, the linguistic processor synthesizes natural language comments to the solution and the objects of the drawing;
- the authors proposed a method for interactive visualization of the linguistic processor operation, which provides a step-by-step control of building a syntactic structure and its mapping to semantic representation; the method involves a dialog reference to the ontology, which significantly increases the visibility and the credibility of the adopted system solutions. For the knowledge engineer, the method greatly facilitates and accelerates the process of adaptation and maintenance of the integrated system.

Recommendations

The study allowed us to take a new look at the problem of visualization of the process of linguistic support for solving the problem – from the graph representation of the syntactic structure in the natural language description of the problem and the interactive graphs of semantic transformations to the visualization of the solution Protocol and drawing.

The applied aspect of the research is the creation of automatic training systems of a qualitatively new level, including linguistic processing of the problem, its automatic solution, visualization of the drawing and comments of the decision process based on the methodology of Polya. The proposed further direction of research is related to the following:

- with the extension of the class of solved problems (with the corresponding extension of rules and heuristics in the ontology);
- with the creation of a task bank: displaying their texts, solutions and related drawings and explanations of the solution;
- with the introduction based on this body of knowledge (task bank) of machine learning elements in the process of building solutions.

ACKNOWLEDGEMENT

The research work was carried out with the support of the Russian Foundation for Basic Research, project No.18-07-00098 A, “The Integrated System of Solving Problems with Natural Language Interface”.

REFERENCES

- Apresyan, Y., Boguslavsky, I., Iodmin, L., Lazursky, A. V., Sannikov, V. Z., & Tsinman, L. L. (1989). *Linguistical maintenance of the system ETAP-2*. Nauka.
- Gan, W., & Yu, X. (2018). Automatic understanding and formalization of natural language geometry problems using syntax-semantics models. *International Journal of Innovative Computing, Information and Control ICIC*, 14(1), 83–98.
- HOL Interactive Theorem Prover. (2018). Retrieved December 21, 2019, from <https://hol-theorem-prover.org/>
- Isabelle is a generic proof assistant. (2019). Retrieved December 21, 2019, from <https://isabelle.in.tum.de/>
- JSXGraph. (2018). *Dynamic Mathematics with JavaScript, JSXGraph is a cross-browser JavaScript library for interactive geometry, function plotting, charting, and data visualization in the web browser*. Retrieved December 21, 2019, from <http://jsxgraph.uni-bayreuth.de/wp/index.html>
- JSXGraph Reference. (2018). Retrieved December 21, 2019, from <http://jsxgraph.uni-bayreuth.de/docs/symbols/JXG.Board.html>
- Kurbatov, S., Naidenova, X., & Khakhalin, G. (2011). Integrating intelligent systems of analysis synthesis of images and text: project outlines INTEGRO. Contours of the INTEGRO project. In *Proceedings of the International Scientific Conference "Open Semantic Technologies for Intelligent Systems" (OSTIS-2011)* (pp. 213-232). Minsk: BSUIR.
- Kurbatov, S., & Vorobyev, A. (2016). Ontological solver of geometry problems in natural language description. *Proceedings of 15th National Conference on Artificial Intelligence with International Participation (CAI-2016)*, 1, 56-63.
- Kurbatov, S. S., Fominykh, I. B., & Vorobyev, A. B. (2018). Polya-method: computer embodiment of the methodology D. Polya. *Conference on Artificial Intelligence, KII-2018*, 2, 96-104.
- Kurbatov, S.S., Fominykh, I.B., & Vorobyev, A.B. (2019). Algorithmic and software cognitive agent based on the methodology D. Poya. *Software products and systems/Software & Systems*, 32(1), 12-19.
- Lobzin, À., Khakhalin, G., Kurbatov, S., & Litvinovich, A. (2015). Integration based on natural language and image ontology in the system Text-To-Picture. In *Proceedings of 8-th Scientific-Practical Conference "Integrated Models and Soft Computing in Artificial Intelligence"* (vol. 2, pp. 296-305). Moscow: Fizmatlit.
- MathJax Documentation. Release 2.7. (2018). Retrieved December 21, 2019, from <https://media.readthedocs.org/pdf/mathjax/latest/mathjax.pdf>
- Morphological analysis of the word with examples and online. (2019). Retrieved December 21, 2019, from <https://uchim.org/russkij-yazyk/morfologicheskij-razbor-slova>
- Morton, K. F., & Qu, Ya. (2013). A novel framework for math word problem solving. *International Journal of Information and Education Technology (IJIET)*, 3(1), 88–93. doi:10.7763/IJIET.2013.V3.240
- Naidenova, X., Kurbatov, S., & Ganapol'skii, V. (2019). *An analysis of plane task text ellipticity and the possibility of ellipses reconstructing based on cognitive modeling geometric objects and actions*. Retrieved December 21, 2019, from <http://ceur-ws.org/Vol-2303/paper3.pdf>
- Paducheva, E. V. (1974). *On the semantics of syntax. Materials to the transformational grammar of the Russian language*. Nauka.
- Podkolzin, A. S. (2008). *Computer simulation of logical processes. Architecture and problem solver languages*. Fizmatlit.
- Podkolzin, A. S. (2016). The study of logical processes by computer simulation. *Journal "Intelligent Systems. Theory and Applications"*, 20, 164–168.
- Polya, G. (1962). *Mathematical discovery: on understanding, learning and teaching, problem solving*. Wiley.
- Polya, G. (2004). *How to solve it*. Princeton University Press.

- Popov, E. V. (2011). *Talking with computers in natural language*. Springer-Verlag.
- Prasolov, V. V. (2006). *Tasks on planimetry*. Moscow Center for Continuous Mathematical Education.
- Progress (DBMS). (n.d.). *Progress OpenEdge 11.7*. Retrieved December 21, 2019, from <https://www.progress-tech.ru/>
- PullEnti-SDK extract named entities from unstructured texts. (2019). Retrieved December 21, 2019, from <http://pullenti.ru/CompetitorPage.aspx>
- Schenk, R. C., Goldman, N. M., Rieger, Ch. J., & Riesbeck, Ch. K. (1975). *Conceptual information processing*. North-Holland Publishing Company.
- Seitova, S., Smagulov, Y., Khaimuldanov, Y., Adilbaeva, A., Tulymsheva, G., & Abdykarimova, A. (2018). Methodological requirements for the arrangement of independent work on neuro-linguistic programming techniques for mathematical disciplines at the university. *Periódico Tchê Química*, 15(30), 330–337.
- Seo, M., Hajishirzi, H., Farhadi, A., Etzioni, O., & Malcolm, C. (2015). *Solving geometry problems: combining text and diagram interpretation*. Retrieved December 21, 2019, from <http://geometry.allenai.org/assets/emnlp2015.pdf>
- Shi, Sh., Wang, Yu., Lin, Ch.-Ye., Liu, X., & Rui, Y., Yo. (2015). Automatically solving number word problems by semantic parsing and reasoning. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1132-1142). Lisbon: Association for Computational Linguistics. doi:10.18653/v1/D15-1135
- Silver, D., Huang, A., Maddison, Ch. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., & Hassabis, D. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587), 484–503. doi:10.1038/nature16961 PMID:26819042
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, Th., Baker, L., Lai, M., Bolton, A., Chen, Yu., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, Th., & Hassabis, D. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676), 354–359. doi:10.1038/nature24270 PMID:29052630
- Solving Natural Language Math Problem. (2019). Retrieved December 21, 2019, from <http://aitp-conference.org/slides/TM-AITP.pdf>
- Sundaram, S. S., & Khemani, D. (2015). Natural Language Processing for Solving Simple Word Problems. In *Proceedings of the 12th International Conference on Natural Language Processing* (pp. 394-402). Goa: NLP Association of India.
- The Coq Proof Assistant. (2019). Retrieved December 21, 2019, from <https://coq.inria.fr/news/coq-890-is-out.html>
- Wang, K., & Su, Zh. (2015). Automated geometry theorem proving for human-readable proofs. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence* (pp. 1193-1199). Palo Alto: AAAI Press.
- Winograd, T. (1972). *Understanding natural language*. Academic Press. doi:10.1016/0010-0285(72)90002-3
- Wolfram Release 11.3. (2018). Retrieved December 21, 2019, from <http://blog.stephenwolfram.com/2018/03/roaring-into-2018-with-another-big-release-launching-version-11-3-of-the-wolfram-language>
- Zhang, D., Wang, L., Xu, N., Dai, B. T., & Shen, H. T. (2018). *The gap of semantic parsing: a survey on automatic math word problem solvers*. Retrieved December 21, 2019, from <https://arxiv.org/pdf/1808.07290.pdf>

Sergey S. Kurbatov is a PhD in Technical Sciences, Leading Researcher of Scientific Research Centre for Electronic Computer Technology, Member of the Russian Association of Artificial Intelligence (RAII). Research interests: automatic processing of a natural language (linguistic translation), presentation of knowledge (ontology), automatic problem solving, conceptual visualization. He is the author of more than 25 publications in the field of AI.