The Semantic Space in Factorial of the Theory of Automatic Generation of Knowledge Architecture (TAGKA-2)

Aliaksandr Hardzei
Belarus State University
Minsk, Belarus
Email: alieks2001@yahoo.com

***Abstract*—Combinatorial possibilities of TAGKA-2 [1] in Natural Language Processing, Natural Language Understanding, Inventive Problem Solving are presented by 696 192 modifications of 112 classes of actions in base calculation multiplied by 31 roles of individuals in each Field of Knowledge. Thus, the total number of TAGKA-2 formalisms is 21 581 952, while the Dictionary of Modern Russian Literary Language (in 17 volumes) contains 120 480 words. The amount claimed by authors of Russian Large Academic Dictionary is 150 000 words and the available electronic resources of the Institute of Linguistic Studies of Russian Academy of Sciences contain about 5 million Russian-speaking words of XVIII-XXI centuries based on 1,4 billion word usages [2].**

Keywords*—*semantic code, semantic primitive, calculus of fields of knowledge, modification of action; recursion, artificial intelligence.

I. Introduction

T. Winograd in his research understood what is required from semantics: "converter working with the parser and providing data suitable for a logical deductive system. If you have the English grammar parser and the deductive system on the basis of knowledge about a particular subject, the role of semantics is reduced to filling the space between them" [3]. However, how to achieve that no one knew.

To the credit of the Belorussian science, V.V. Martynov reached essential progress in this area in creating a semantic coding approach [4]. He proposed a list of tasks to equip computer with encyclopedic knowledge consisting of five components:

1) To calculate semantic primitives, i.e. semantically irreducible kernel words and define rules of their combinatorics.
2) To define the necessary and sufficient set of formal characteristics constituting 'dictionary entry'.
3) To define a set of semantic operations for calculating a subject domain of any kind.
4) To propose heuristic teaching rules to work with the system.
5) To build a system of mutual references based on semantics [5].

In 1994, under the supervision of V.V. Martynov, A. Hardzei [6] proposed the first procedure of calculating the subject domains in the form of a directed graph of complex strings.

Use of the procedure has required to establish one-to-one (vector) transition between actions in basic semantic classifier and has led to creation of the Theory of Automatic Generation of Knowledge Architecture (TAGKA) founded on: the formal theory, the semantic counterpart, the table of actions as semantic elements, the algorithm defining roles of individs, and the graph of search of hyponyms through hypernyms [7].

In 2014 the new version – TAGKA-2 was developed. TAGKA-2 differs from the previous in: simplified algebraic apparatus, increased number of rules of interpretation of the standard superposition of individs, and minimized semantic calculus.

The number of operations with the strings of semantic code reduced to two. Now it is the algebra of the type:

$$A =< M, \star, ^- > \text{ [1].}$$

Where: M is a set of elements, '$\star$' is the operation of superposition, '‾' is the operation of extension.

II. Semantic Coding Evolution

A practical example of notation evolution for the semantic formula of the action "compress":

USC-1: $S(a)A_1\bar{S}(a)AO/O$ [8]

USC-6: $((XY)Z)((ZW)W')$ [9]

TAGKA-2: $Z((\bar{Z}\bar{W})W)$ [1]

Theoretically, there is a set of changes:

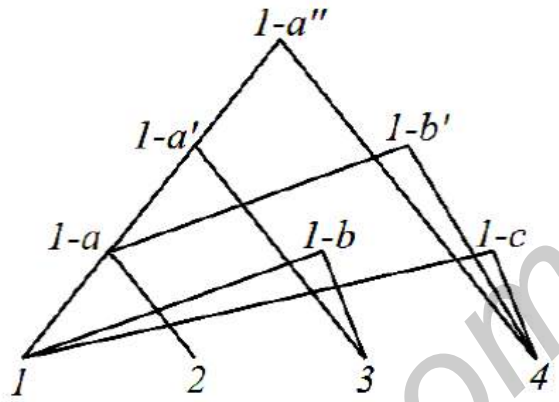
- A geometric model is introduced and consistency of the algebra is verified. Constructions leading to mathematical and semantic paradoxes are forbidden:
  XX
  XYX
  $XYZZWZ$
  $XYZZWZ(x)$
  A strict rule of right margin extension of multipliers is established:
  $X \to Y \to Z \to W$
- Algebraic operations are refined.

- Rules of creation, restriction, reduction, and transformation of algebraic expressions are presented in an explicit form.
- Now, there is only one rule of transformation – transposition.
- Non-commutative one-to-one (vector) transitions between algebraic expressions and their arguments are established:

$$\underset{\substack{\text{attract}\\ Y_2 \text{ imp } X_2}}{((X*\overline{X})*\overline{\overline{Y}})*\overline{Y}} \rightarrow \underset{\substack{\text{cumulate}\\ Y_2 \text{ imp } X_2}}{(X*(\overline{X}*\overline{\overline{Y}}))*\overline{Y}} \rightarrow$$

$$\underset{\substack{\text{constrict}\\ Y_2 \text{ imp } X_2}}{X*((\overline{X}*\overline{\overline{Y}})*\overline{Y})} \rightarrow \underset{\substack{\text{connect}\\ Y_2 \text{ imp } X_2}}{X*(\overline{X}*(\overline{\overline{Y}}*\overline{Y}))} \rightarrow$$

- The procedure to calculate subject domains and its semantics in the view of an oriented graph is defined:



- The algorithm of roles of individs (macro objects) is developed.
- 112 semantic primitives (abstract actions or classes of actions) are computed and grouped in the table as rows of semantic elements.
- All micro-actions are considered as modifications of class actions. The number of micro-actions only on the first modification level is $112 \cdot 111 = 12432$. On the second level it reduced in the progression n-1 for the second multiplier $112 \cdot 110 = 12320$. So the number of micro-actions almost doubles $12432 + 12320 = 24752$. It means the power of the system lays in the interval of factorials (9!; 10!), that is very large but finite [1].

- Were defined the interpretation rules of regular superposition of individs to define semantic counterparts for algebraic expressions, for example:
$\bar{\bar{\alpha_1}} * \bar{\alpha_1}$ after $\alpha_1 * \bar{\alpha_1}$ (superposition of the surroundings of α_1 with its shell as a result of physical effect on the surroundings of α_1) – *mold* α_1;
$\bar{\bar{\alpha_1}} * \bar{\alpha_1}$ after $(\alpha_1 * \bar{\alpha_1}) * \bar{\bar{\alpha_1}} \simeq \alpha_1 * \bar{\bar{\alpha_1}}$ (superposition of the surroundings of α_1 with its shell as a result of informational effect on the surroundings of α_1) – *predispose* α_1;
$\bar{\alpha_1} * \alpha_1$ after $\alpha_1 * \bar{\alpha_1}$ (superposition of the shell of α_1 with α_1 as a result of physical effect) – *form* α_1;
$\bar{\alpha_1} * \alpha_1$ after $(\alpha_1 * \bar{\alpha_1}) * \bar{\bar{\alpha_1}} \simeq \alpha_1 * \bar{\bar{\alpha_1}}$ when $\alpha_1 = \alpha, \beta$ (superposition of the shell of α_1 with α_1 as a result of informational effect) – *bring up* α_1.
- Was defined a procedure of consequent extension of multipliers in algebraic expressions when recursively expanding the geometric model:

$$S_1 \vec{A}_1 I_1 \vec{A}_2 M_1 \vec{A}_3 O_1$$
$$S_{0.1} \vec{A}_{0.1} I_{0.1} \vec{A}_{0.2} M_{0.1} \vec{A}_{0.3} O_{0.1}$$
$$S_{0.0.1} \vec{A}_{0.0.1} I_{0.0.1} \vec{A}_{0.0.2} M_{0.0.1} \vec{A}_{0.0.3} O_{0.0.1},$$

where: S – subject, A – action, I – instrument, M – mediator, O – object.

- Compound strings are excluded and a description of the event is represented as non-commutative superposition of actions or as their non-commutative preposition. Moreover, superposition or preposition of actions covers the same superposition or preposition of micro-actions in recursion.

Let us consider some examples for superposition. The action "carry" is not considered as a combination of actions "hold" and "move" (see USC-6 [9]), but it is considered as a modification of the action "move" in the microsystem of the compound instrument and for the action "transport" of the action "move" in the microsystem of the compound mediator.

In that case, the action "fix" is in superposition to the action "move" because it is possible to hold without moving but it is not possible to move without fixing. It is impossible to move a motor car from the parking if a handbrake relatively to the ground fixes the car. It is necessary to release the brake and fix the car relatively to the hands by pressing them to the car.

One more example: it is impossible to move a wall from the standing point because it fixed relatively to the ground. To do that, under the wall, let say, a sled should be inserted to unfix it relatively to the ground and creating a condition to push the wall by hands. So, "fixation" is a preliminary condition for any kind of movement. There is obvious superposition of actions: **fix** $\rightarrow$ **move**.

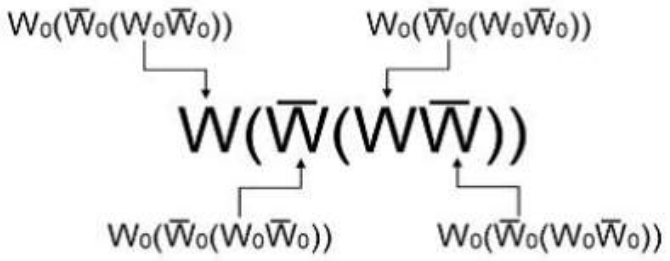
The next example is for preposition of the action. When you are cooking, it is not important what to do first: to place a pan on a gas stove and then to light it or to light the gas stove and then to put the pan. However, for the purpose of safety, it is preferable to put the pan and then to light the stove, or preposition: **put** $\rightarrow$ **light**.

[1] The precise number of micro-actions is calculated by C. Rykovski with the formula:
$S_n = \frac{2a_1+d(n-1)}{2} \times n$, where $L = 112$, $a_1 = L-1$, $d = -1$, $n = L-1$;
$S_n = \frac{2(L-1)+(-1)(L-1-1}{2} \times (L-1) = \frac{2L-2-L+2}{2} \times (L-1) = \frac{L(L-1)}{2}$;
$\sum = 112 \times S_n = \frac{L^2(L-1)}{2} = 696\,192$
Because of 31 roles of individs for each subject domain there is a final number of semantic formulas in TAGKA-2 is 696 19231 = 21 581 952.

The last example is about superposition of actions in recursion. To drill a titan plate it is necessary to cool a drill by a cooling liquid to save a drill. The process starts from pouring the liquid to a drilling point and then drilling is implementing. We can say that drilling is implemented on the condition of pouring the liquid: pour $\rightarrow$ drill.

So, to create the event a vector transition between actions has to be implemented.

Here is the example of recursion of the action “normalize” in TAGKA-2:



A canonized text, for the example, is "A doctor by means of a medicine treats a patient".

III. An Inventive Problem

How to remove the shell of an egg? According to the TAGKA-2 table of semantic elements, three possible solutions can be generated [1]:

1) In the subgroup “surroundings – shell” the solving action is “fractionize” form outside.
2) In the subgroup “shell – core” the solving action is “dissolve” (inset a dissolving reagent under the shell).
3) In the subgroup “core – shell” the solving action is “unclamp” from inside by expanding a volume of the core.

IV. Conclusion

TAGKA-2 is one of possible models of calculation of semantics and it demonstrates essential progress in semantic coding development especially in subject domain calculation. However, TAGKA does not claim to be exclusive and complete.

Semantics of natural language is versatile and allows different ways of formalizing. However, all methods, like Euclidean and non-Euclidean geometry, should be consistent and effective in its problem solving, and those who argue with that, as Reichenbach aptly said, only "confuse a rigor of the method with a limitation of a goal" [10].

References

[1] A. Hardzei, Theory of the knowledge architecture automatic formation (TKAAF) and further minimization of semantic calculus. // Conference proceeding: Open Semantic Technologies for Intelligent Systems (OSTIS-2014), Minsk, 2014.

[2] L.E.Kruglikova, «l gran diccionario académico de lalengua rusa» como continuador de las tradiciones de la lexicografía académica rusa // Cuadernos deRusística Española. 2012. No 8.

[3] T. Winograd, Understanding Natural Language. Cognitive Psychology, 1972, 3:1. p.26.

[4] V.V. Martynov, Cybernetics. Semiotics. Linguistics, Minsk, Science and technics, 1966, p.148. (in Russian)

[5] V.V. Martynov, Foundations of semantic coding. Experience of knowledge representation and conversion, Minsk, EHU, 2001. (in Russian)

[6] A. Hardzei, Procedural semantics and calculus of subject domains. // Language: Semantics, syntactics, pragmatics: I International scientific conference, MSLU, Minsk, 1994. (in Russian)

[7] A. Hardzei, The deductive theory of language. Minsk, Belarus science, 1998. (in Russian)

[8] V.V. Martynov, Semiology foundations of informatics, Minsk, Science and technics, 1974. (in Russian)

[9] V.V. Martynov, In the center of conscious of a human, Minsk, BSU, 2009. (in Russian)

[10] M. Reichenbach, English translation: The Philosophy of Space and Time. Dover. 1957, ISBN 0-486-60443-8 (1928, Philosophie der Raum-Zeit-Lehre)

СЕМАНТИЧЕСКОЕ ПРОСТРАНСТВО В ФАКТОРИАЛЕ ТЕОРИИ АВТОМАТИЧЕСКОГО ПОРОЖДЕНИЯ АРХИТЕКТУРЫ ЗНАНИЙ (ТАПАЗ-2)

А. Н. Гордей

Комбинаторные возможности ТАПАЗ-2 в компьютерной обработке данных на естественном языке, понимании естественного языка, решении изобретательских задач представлены в виде 696 192 модификаций 112 макропроцессов базового исчисления с умножением на 31 роль индивидов каждой предметной области. Таким образом, общее количество формализмов ТАПАЗ-2 составляет 21 581 952, тогда как Словарь современного русского литературного языка в 17-и томах содержит 120 480 слов, заявленный объём Большого академического словаря русского языка – 150 000 слов, а имеющиеся электронные ресурсы Института лингвистических исследований РАН на 1,4 млрд словоупотреблений содержат около 5 млн русскоязычных слов XVIII–XXI вв.