

Linguistic Data Model for Natural Languages and Artificial Intelligence. Part 6. The External Logic

Oleg M. Polyakov✉

Saint Petersburg State University of Aerospace Instrumentation, St Petersburg, Russia

✉road.dust.spb@gmail.com

Introduction. The article continues a series of publications on the linguistics of the relationship (hereafter R-linguistics) and is concerned with the semantic interpretation in terms of the linguistic model that is the initial stage to consider the logic of natural language (external logic).

Methodology and sources. The results obtained in the previous parts of the series are used as research tools. In particular, the verbal categorization method is used to represent concepts and verbs. To develop the necessary mathematical representations in the field of logic and semantics of natural language, the previously formulated concept of the interpretation operator is used. The interpretation operator maps the sentences of the language into the model, taking into account the previously interpreted sentences.

Results and discussion. The problems that arise during the operation of the natural language interpretation operator are analyzed using examples of text translation and utterance algebra. The source of problems is the dependence of the interpretation of sentences on the already accumulated results of interpretation. The features of the interpretation of negation and double negation in the language are analyzed. In particular, the negation of a sentence affects the interpretation of previous sentences, and double negation usually denotes a single negation with an indication of its scope. It is shown that even from the point of view of classical logic, linguistic negation is not unconditional, and the operation of concatenation is not commutative and associative. General rules of text interpretation in the form of step-by-step mapping of sentence elements into a linguistic model are formulated.

Conclusion. From the considered examples of the implementation of the interpretation operator, it follows that the negation of a sentence requires a change in the meaning of the operation of attributing sentences in the text. For this reason, the negative particle "not" in the language is actually a label for changing the interpretation rule. The double negation rule in sentence logic does not hold, so sentences containing double negations are likely to contain information about the scope of the sentence negation in the text. Based on the analysis, the contours of the interpretation operator for the linguistic model are indicated.

Keywords: R-linguistics, ascription operation, interpretation operator, semantics.

For citation: Polyakov O. M. Linguistic Data Model for Natural Languages and Artificial Intelligence. Part 6. The External Logic. DISCOURSE. 2021, vol. 7, no. 2, pp. 127–134. DOI: 10.32603/2412-8562-2021-7-2-127-134

Conflict of interest. No conflicts of interest related to this publication were reported.

Received 07.01.2021; adopted after review 10.03.2021; published online 23.04.2021

© Polyakov O. M., 2021



Контент доступен по лицензии Creative Commons Attribution 4.0 License.
This work is licensed under a Creative Commons Attribution 4.0 License.

Лингвистическая модель данных для естественных языков и искусственного интеллекта. Часть 6. Внешняя логика

О. М. Поляков✉

Санкт-Петербургский государственный университет
аэрокосмического приборостроения, Санкт-Петербург, Россия

✉road.dust.spb@gmail.com

Введение. Статья продолжает серию публикаций по лингвистике отношений (далее R-лингвистика) и посвящена вопросам семантической интерпретации в рамках лингвистической модели, что является первоначальным этапом к рассмотрению естественной логики языка (внешней логики).

Методология и источники. В качестве инструментов исследования используются результаты, полученные в предыдущих частях серии. В частности, для представления понятий и глаголов применяется глагольный способ категоризации. Для разработки необходимых математических представлений в области логики и семантики естественного языка используется сформулированное ранее понятие оператора интерпретации. Оператор интерпретации отображает предложения языка в модель с учетом ранее интерпретированных предложений.

Результаты и обсуждение. На примерах перевода текста и алгебры высказываний проанализированы проблемы, которые возникают при работе оператора интерпретации естественного языка. Источником проблем является зависимость интерпретации предложений от уже накопленных результатов интерпретации. Проанализированы особенности интерпретации отрицания и двойного отрицания в языке. В частности, отрицание предложения оказывает влияние на интерпретацию предшествующих предложений, а двойное отрицание, как правило, обозначает одиночное отрицание с указанием области его действия. Показано, что даже с точки зрения классической логики языковое отрицание не является безусловным, а операция конкатенации (приписывания) не является коммутативной и ассоциативной. Сформулированы общие правила интерпретации текста в виде пошагового отображения элементов предложения в лингвистическую модель.

Заключение. Из рассмотренных примеров реализации оператора интерпретации следует, что отрицание предложения требует изменения смысла операции приписывания предложений в тексте. По этой причине отрицательная частица «не» в языке фактически является меткой изменения правила интерпретации. Правило двойного отрицания в логике предложений не выполняется, поэтому предложения, содержащие двойные отрицания, скорее всего, содержат информацию об области действия отрицания предложения в тексте. На основе проведенного анализа обозначены контуры оператора интерпретации для лингвистической модели.

Ключевые слова: R-лингвистика, операция приписывания, оператор интерпретации, семантика.

Для цитирования: Поляков О. М. Лингвистическая модель данных для естественных языков и искусственного интеллекта. Часть 6. Внешняя логика // ДИСКУРС. 2021. Т. 7, № 2. С. 127–134. DOI: 10.32603/2412-8562-2021-7-2-127-134

Конфликт интересов. О конфликте интересов, связанном с данной публикацией, не сообщалось.

Поступила 07.01.2021; принята после рецензирования 10.03.2021; опубликована онлайн 23.04.2021

Introduction. The questions discussed in the previous article [1] are not just philosophical questions, although there the semantic model is extremely general. This, as stated, means that the postscript to the text s^{-1} (the postscript of the negation of the sentence s) formally affects not only the sentence s nearest to the left, but also the sentences preceding the sentence s . In the following two examples, we will consider how the results described above are implemented in specific situations.

Methodology and sources. The results obtained in the previous parts of the series are used as research tools. In particular, the verbal categorization method is used to represent concepts and verbs. To develop the necessary mathematical representations in the field of logic and semantics of natural language, the previously formulated concept of the interpretation operator is used. The interpretation operator maps the sentences of the language into the model, taking into account the previously interpreted sentences.

Results and discussion. The first example is related to the translation of tests. So, let P_1 and P_2 be sets of sentences of two languages and the text composed of sentences of the set P_1 is translated into the text composed of sentences of the set P_2 . The semantics of the text t in the first language means its translation s in the second language. For example, the semantics of the English text in this case is the Russian text obtained as a result of translation. Translation (interpretation of the English text in Russian) is carried out as follows: first, the first sentence of the text t_1 is translated and the translation of this sentence s_1 is obtained. The second sentence t_2 is translated into sentence s_2 , taking into account the already received sentence s_1 , and so on. In order to fully fit into the semantic model described in [1], it is necessary to make two assumptions about the translation. First, an empty sentence of the first language is translated into an empty sentence of the second language, regardless of what translation has already been received. Secondly, either two identical, consecutive sentences in the first language are translated one sentence regardless already received a translation (though translation itself is, of course, depends on the already obtained translation), or they can be two sentences, but then we believed hypothesis idempotency applies to the second language.

It is easy to see that the described method of translation exactly corresponds to the model of interpretation described earlier and, therefore, the results of the previous subsection are valid for it. In particular, for example, the translation of a sentence and the negation that follows it must depend on the text that has already been translated, otherwise such a translation may lose its meaning. This provision is important for understanding the features of the semantics of simultaneous translation: a simultaneous translator must remember the already translated text and make the current translation taking into account the already translated text, otherwise he risks destroying the semantics of the translation.

The second example is related to the logic of statements. In logic, the operations of disjunction and conjunction are associative, and negation is unconditional. Does this mean that logic has not only formalized logical operations between language sentences, separating them from meaning, but it has also rendered meaningless any natural language to which it applies? This question ended the previous article in the series and now it's time to answer it.

As you know, in expert systems, the text is divided into separate sentences, and this set of sentences is the basis for subsequent output. This is done because they assume the operation of attributing an identical conjunction (that is, not just associative, but even commutative). All the

denials in expert systems are considered to be unconditional. And yet, contrary to Theorem 1 [1], logic has a model in the form of set theory and, therefore, has a set-theoretic semantics. What's the matter? What is the difference between semantic and logical approaches to text?

To begin with, we note that when we talk about external logic, we mean the logic of actions with the text sentences themselves. At the same time, logical connectives are often found inside sentences, but we will not consider logical operations inside sentences now, but only outside them. We will postpone the consideration of logic within sentences until the next article.

For simplicity, consider suggestions about the possession of an object by some property. Each such sentence is represented by a predicate $P(x)$ (literal), which takes the value TRUE on those objects x that have the property $P(x)$. Thus, each such predicate defines a set of A_p objects that have the property $P(x)$.

The logical approach to the text from the standpoint of truth makes us consider the text as a conjunction of sentences, where sentences are considered as literals. Thus, the truth of the text is represented as a conjunction of the truth values of sentences. Since the conjunction corresponds to the intersection of the sets of A_p , then with each sentence the result of the intersection (the truth area of the text) decreases.

However, our semantic experience says that in the process of reading a text, the "amount" of meaning increases. For example, one sentence in a book makes less sense than the whole book. So, contrary to the logical view, in this logical example, we will look at the attribution operation as a disjunction of literals $P(x)$, which corresponds to the union of sets A_p . An empty sentence corresponds to the property P_\emptyset , which defines an empty set.

When we look at the text from a semantic point of view, we have in mind the following situation. Someone says to us: "Please write down such information: ..." We are not required to verify the truth of the text, and in this sense, the logical and semantic approaches are opposed to each other. In the semantic approach, we are required to interpret, because truth analysis is a matter of matching the received semantics and the model of the world. For example, owned by Bertrand Russell [2] the dilemma "the current king of France is bald" is quite normally interpreted. Another thing is that in the model of the world of a modern European, the current France does not have a king. However, there are many people in the world with a model of the world that does not respond to Russell's sly look.

Disjunction as an attribution operation satisfies two introduced axioms of the language. In fact, for any $P(x)$ we have $P(x) \vee P_\emptyset(x) = P(x)$ and $P(x) \vee P(x) = P(x)$, and these equalities, due to the associativity of the disjunction, are fulfilled for any previous disjunction of literals (that is, they are unconditional).

Now note that no disjunction can transform the non-empty set A_p of the predicate $P(x)$ into an empty set. This means that we cannot define semantic negation in terms of disjunction. We can only define the attribution of negation as the conjunction $P(x) \& \square P(x)$ (the intersection of A_p and $\square A_p$), and more precisely, as the second operand, we can take any set belonging to $\square A_p$. As follows from Theorem 1 [1], such a negation cannot be an unconditional right negation, since then, for example, $P_\emptyset(x) = P(x) \& \square P(x) = P(x) \vee P(x) \& \square P(x) = P(x)$. Thus, if $Q(x)$ is some disjunction of literals, then the negation of the last literal is $(Q(x) \vee P(x)) \& \square P(x) = (Q(x) \& \square P(x)) \vee (P(x) \& \square P(x)) = Q(x) \& \square P(x)$. In this situation, since $Q(x)$ can be any, it is necessary to choose as a negation the maximum subset of the set $\square A_p$, that is, $\square A_p$ itself.

So, the need to circumvent Theorem 1 [1] in connection with the associativity of disjunction leads to the need to replace the chosen attribution operation, which was a disjunction, with another operation, and so that this operation affects the previous result of combining sets (accumulated semantics), which removes the question of the unconditionality of negation. Note that $Q(x)*P(x)*\square P(x) = Q(x)\&\square P(x)\neq Q(x)\&P(x) = Q(x)*\square P(x)*P(x)$. In other words, the literal and its negation cannot be swapped (not commutativity).

The considered example from the point of view of semantics denotes a number of problems.

1. From the logical or set-theoretic point of view, the predicate $\square P(x)$ and its corresponding set are no different from other similar predicates and sets. However, to determine $\square P(x)$, you need to know the entire area of change of x (universe U). This is not always known. Therefore, the set corresponding to $Q(x)\cap\square P(x)$ must be considered as the difference between the sets A_Q and A_P . In this case $A_{\square P} = U - A_P$ and for $Q(x)*\square(\square P(x))$ we get $Q(x)*\square(\square P(x)) = Q(x) - (U(x) - P(x)) = Q(x)\&P(x) \neq Q(x)\vee P(x) = Q(x)*P(x)$, so there is no double negation in the semantic sense.

2. Since the occurrence of a negative sentence in the text leads to a change in the way of interpretation (to the transition from disjunction to difference), there is a need to somehow mark (highlight) negative sentences. In other words, positive and negative sentences and corresponding literals are not equivalent, and the logical system loses its symmetric character. In Russian, such a label is the particle “not”. Apparently, any language that has a developed logical structure must have a “negativity” label for the sentence. This label is not needed to indicate a connection with a positive sentence that formed a negative one. This label has not an illustrative, but a fundamental semantic character, since when it appears, the way the meaning is formed changes.

3. So, when a negative literal appears, it is necessary to remove from the sets corresponding to the preceding literals the elements belonging to the set from which the negation is formed. Do I need to do this with subsequent text literals? If yes, how long and for what? For example, if as a result of negating the literal “x-round”, the object “ball” is removed from the previous sets, then is it necessary to react somehow to the appearance of this object in subsequent sets? Maybe if this conflict occurs, you should ask a question at the end of the paragraph: “You said that the ball should not be included in the semantics of the text, because it is round?” This will allow you to adjust the semantic result. Apparently, double negations in sentences serve as an indication of the method of action when such a collision occurs. For example, the sentence “I will never love you” “not” in the verb simply negates the binary relationship of love between objects A and B. But it is reinforced by the word “never”. Most likely, this means that checking the absence of such a connection should be done not only in subsequent sentences of this text, but also in the models of the world of objects a and b, in order to check any text for the absence of the pair “a loves b”.

4. Finally, it is important to understand what part of the formulas of the algebra of statements can be expressed by texts with the structure described above, taking into account the action of certain reinforcing negatives.

Let’s go back to the language and texts. Let $t = s_1*...*s_n$ be some text. Denote by $s = s_i*...*s_{i+j}$ some part of the text, then $t = s_1*...*s_{i-1}*s*s_{i+1}*...*s_n = s_1*...*s_i*...*s_{i+j}*...*s_n$. In essence, s is several sentences of the text t , highlighted because they form a compound sentence

using the conjunction (union) “and”. This can be denoted by parentheses $t = s_1 * \dots * (s_i * \dots * s_{i+j}) * \dots * s_n$. These parentheses do not indicate a change in the order of actions, but are simply a sign that a certain part of the sentences is combined into one. At the same time, it is generally impossible to rearrange sentences inside parentheses. Let’s say (“it rained and I stayed home”) \neq (“I stayed home and it rained”).

Similarly, we will introduce square brackets into the text, which will mean splitting the text into several branches, depending on the number of sentences in square brackets. Highlighting with square brackets means that the sentences covered by these brackets form one complex sentence using the conjunction (union) “OR”. For example, $s_1 * [s_2 * s_3] * s_4$ means that after the sentence s_1 , the text is split into two chains: $s_1 * s_2 * s_4$ and $s_1 * s_3 * s_4$. In other words, we use square brackets to indicate that two strings of sentences are actually interpreted. The appearance of square brackets causes the interpretation process to branch, so that over time it may require significant resources. That is why when discussing the concept of a paragraph in the previous article, it was noted that it is important to find out the reasons for the completion of a paragraph, since most likely the branching of the interpretation should be completed within the paragraph.

The entered parentheses do not yet carry any logical load: this is only an element of technology in writing texts, which we will need in the future. We will also not consider other conjunctions of compound sentences here, because, first, we are interested in the logical component of texts, and, secondly, other conjunctions are usually reduced to the conjunctions “AND” and “OR” with the addition of some semantic coloring, which is not yet interested in us.

Before we formulate general semantic interpretation rules for the linguistic data model, we will prove one simple fact about the relationship between the linguistic model and its source data. Let be a space and a co-space together with a verb and a co-verb. Is it possible to restore the original relation S from this information? We define S' as follows: $xS'y$ if and only if there are categories X and Y such that $X^\Delta = Y$, $x \in X$, $y \in Y$. In other words, S' is obtained by combining Cartesian products of categories that are connected by a verb.

Theorem 1. $S = S'$.

Proof.

Let xSy . Then $y \in x^\Delta$ and $x \in x^{\Delta\nabla}$, and $x^{\Delta\nabla} x$ and x^Δ are connected by the verb, and, therefore, $xS'y$. Conversely, let $xS'y$, then there are categories X and Y such that $x \in X$, $y \in Y$, and $X^\Delta = Y$. Since $y \in \bigcap_{x \in X} x^\Delta$, then xSy .

It would seem that not all categories can be used to restore S , but only \cap -generators (or Σ -generators). Unfortunately, this is not the case. The problem is that the generators are defined to reconstruct the linguistic space, not the relation. In particular, Σ -generators may not even cover all the elements included in the relation (all kinds). This, however, does not apply to U -generators.

Investigation. The relation S can be restored only by using U -generators spaces and \cap -generators co-spaces, with the addition of some additional categories to the latter.

Proof.

For the proof, as in theorem 1, the categories $x^{\Delta\nabla}$ and x^Δ are used. It remains only to recall that, according to proposition 20 [3], the categories $x^{\Delta\nabla}$ and only they are U -generators. In this case, some co-categories of x^Δ may not be \cap -generators.

In conclusion of this article, taking into account theorem 1 and the considered example with logic, we formulate some ideas about the semantics of the text of narrative sentences from the point of view of R-linguistics.

1. If a proper name occurs in the text of a sentence for which there is a specific image (a tuple of features), then this tuple (or a reference to it) is stored in the RAM of meanings in the model.

2. If a category occurs in the text, then it corresponds to a link to the pattern recognition algorithm of this category in RAM. This means that we can determine the composition of features (parameters) and the set of their values for a given category, as well as restore the set of tuples from the values of features (parameters) that correspond to the category. If the text uses the plural (for example, “trees”), then the reference means that it refers to any tuples that will result in recognition of this category. If we are talking about a single object, then the reference means that this object must be identified (recognized) in this category (belong to this category). Using the values of features and parameters, from the identification algorithm (recognition) and tuples of individual objects for this category, you can create an image of the object. It will contain general features from the identification algorithm (recognition) and specific features, supplemented with parameters from the images of literal memorization. For example, the image of the “tree” should contain features that correspond to the identification (recognition) algorithm. The image of a particular tree can be obtained by adding to the “general image” the missing parameters from the literal images of trees stored in our memory. As a result, one person will have the image of a birch, and the other – an oak.

A category can have several identification algorithms. For example, the category “president” has a different identification algorithm in different countries, described in the constitution or in the regulations (for example, the president of the chess federation). These algorithms have common properties. It is on them that a link is obtained by the name of “president”. In other words, we get a shortened tuple of properties. If in the text we meet the category “president of Russia”, then here the reference is already given to one algorithm.

3. Variables should get values whenever possible. In the text, variables are usually used for the categories specified in the preceding sentences. For example, when you encounter the variable “I” in the text of this book, you do not hesitate to replace it with the category “author”. However, there are cases when a variable cannot be assigned a value. One of these cases will be discussed in the next article.

4. If two categories are connected by a verb in some affirmative sentence, then the categories connected by this verb are stored in RAM, and a reference is given in the model to the verb, which is represented by a generator of trajectories of properties modified by this verb. The noun and complement must have properties that are affected by the verb. If the verb is ternary, then it connects three categories. If there are several indirect additions in the sentence, they are also connected to this verb reference. In any case, using tuples of categories, a relation can eventually be constructed that has a verb name (Theorem 1). The circumstances in the sentence usually correct the operation of the trajectory generators.

5. For negative sentences, everything is the same as for affirmative sentences, with the only difference that the tuples of negative sentences are removed from the tuples of the preceding affirmative sentences.

We have sketched only the outlines of the formation of the semantics of the text within the framework of the linguistic model. There are certainly thousands of subtleties here, but we will not consider them yet, since a separate large work should be devoted to this. The purpose of this article is to give only a general idea at the level of language sentences. Now we leave the sentence level, because our main task will be to understand what is happening inside the sentences. This is what we will do in the next article in the series.

REFERENCES

1. Polyakov, O.M. (2020), "Linguistic data model for natural languages and artificial intelligence. Part 5. Introduction to Logic", *DISCOURSE*, vol. 6, no. 3, pp. 109–117. DOI: <https://doi.org/10.32603/2412-8562-2020-6-3-109-117>.
2. Wittgenstein, L. (2017), *Logiko-filosofskii traktat* [Logisch-Philosophische Abhandlung], Kanon+, Moscow, RUS.
3. Polyakov, O.M. (2019), "Linguistic data model for natural languages and artificial intelligence. Part 1. Categorization", *DISCOURSE*, vol. 5, no. 4, pp. 102–114. DOI: <https://doi.org/10.32603/2412-8562-2019-5-4-102-114>.

Information about the author.

Oleg M. Polyakov – Can. Sci. (Engineering) (1982), Associate Professor at the Department of Information Technology of Entrepreneurship, Saint-Petersburg State University of Aerospace Instrumentation, 67 Bol'shaya Morskaya str., St Petersburg 190000, Russia. The author of 32 scientific publications. Areas of expertise: linguistics, artificial intelligence, mathematics, database design theory, philosophy. E-mail: road.dust.spb@gmail.com

СПИСОК ЛИТЕРАТУРЫ

1. Polyakov O. M. Linguistic data model for natural languages and artificial intelligence. Part 5. Introduction to Logic // *DISCOURSE*. 2020. Vol. 6, № 3. P. 109–117. DOI: <https://doi.org/10.32603/2412-8562-2020-6-3-109-117>.
2. Витгенштейн Л. Логико-философский трактат. М.: Канон+, 2017.
3. Polyakov O. M. Linguistic data model for natural languages and artificial intelligence. Part 1. Categorization // *DISCOURSE*. 2019. Vol. 5, № 4. P. 102–114. DOI: <https://doi.org/10.32603/2412-8562-2019-5-4-102-114>.

Информация об авторе.

Поляков Олег Маратович – кандидат технических наук (1982), доцент кафедры информационных технологий Санкт-Петербургского государственного университета аэрокосмического приборостроения, ул. Большая Морская, д. 67, Санкт-Петербург, 190000, Россия. Автор 32 научных публикаций. Сфера научных интересов: лингвистика, искусственный интеллект, математика, теория проектирования баз данных, философия. E-mail: road.dust.spb@gmail.com