Logical Methods for Representing Meaning of Natural Language Texts

Tatyana Batura and Fedor Murzin

A.P. Ershov Institute of Informatics Systems
Novosibirsk, Russia
www.iis.nsk.su
tbatura@ngs.ru, murzin@iis.nsk.su

Abstract. The purpose of the work is development of various algorithms of mapping predicates and formulas of the first-order predicate calculus to the texts in a natural language. The results of the work can be used in the computer-aided systems for processing the texts in a natural language, and for constructing the theory of the text sense, that is a subject of research, first of all in linguistics, and also in mathematical logic.

1 Introduction

Within frameworks of the given project it is supposed to develop methods which will help to analyze texts and separate sentences in natural language from different aspects. It is planned to use such methods as text meaning representation in the context of Melchuk's approach and lexical functions proposed by him [1], works of Apresyan [2], Markus's set-theoretical models [3], methods of classical linguistics ([4], [5]), methods used in computer translation systems [6] and to adapt some methods and constructions from mathematical logic for analyzing texts in natural language, e.g. Henkin's construction used in the Model existence theorem and in the omitting types theorem[7], finite forcing etc.

The purpose of this work is to develop different algorithms for matching predicates and formulas of the restricted predicate calculus with natural language texts. The authors also made an attempt to match finite models with text sentences and even the whole text.

In a future, the obtained results may be studied and transformed by means of methods of mathematical logic, which gives a possibility to realize a transferring from a syntactical to semantic level and in some sense teach a machine to understand a meaning of a natural language text.

The results of this work may be applied in automation systems of extracting information from natural language texts, in intellectual systems of searching information in the Internet, in constructing automated summarizing systems, electronic translators and dictionaries.

M. Bubak et al. (Eds.): ICCS 2004, LNCS 3038, pp. 545–551, 2004.

The present work may also help to develop various search systems, in cases when it is needed to extract necessary information from a document by query or to select required documents from large amount of documents by a given query. On basis of this work it will be possible to develop systems that will be able to reconstruct text sense and extract knowledge from the text that may be presented to user in form of compact reports (schemes, abstracts) or referred to the knowledge base.

2 A Review of Methods for Representing Natural Language Text Meaning

One of the algorithms for predicates matching is based on lexical functions proposed by Melchuk. On syntactic level, these functions may be represented as predicates in the following form. Consider the whole set of word forms in a language which appear when nouns are declined, verbs conjugated etc. (i.e. the whole vocabulary) and suppose that x and y are words or word combinations from this set Than we have predicates of the following form:

$Syn(x, y)$, x, y are synonyms;

$Anti(x, y)$, x, y are antonyms;

$Destr(x, y)$, y is a standard name for an "aggressive"action
($x =$ "оса", $y =$ "жалит").

The Markus set-theoretical models are constructed as follows. Consider some class decomposition of a natural language vocabulary (which is supposed to be a finite set). For example this decomposition may consist of classes corresponding to inflectional wordform sets. With the help of such decomposition it is possible to give a formal definition of Gender and Case. Also Markus defines the so-called "syntactic types"which correspond practically with the traditional parts of speech. On the basis of syntactic types operations there is appearing a possibility to establish grammatical correctness of a natural language sentence.

3 Structures Corresponding to Natural Language Sentences

A part of the carried out work may be described as follows. Each sentence is corresponded by several structures $structure_1, ..., structure_q$ and each structure $structure_i$ is corresponded by predicates $predicate_{i1}, ..., predicate_{ij(i)}$.

On the other hand, it is also possible to consider elements of natural language vocabulary as constants, then introduce predicates and get formulas on their basis. The predicates in their turn are at first considered on syntactic level. After that they are regarded as subsets of basic model sets in corresponding Cartesian powers. This approach gives an opportunity to construct models, i.e. to perform transition from syntactic to semantic level.

As an example, let's consider structures that correspond with verbs. They may be obtained in the following way. Suppose that there is only one verb and several nouns in different cases (which are related to this verb) in the sentence. Every case is considered to have no more than one noun. Such sentence may be matched by the following structure

$\boxed{\text{V}}\boxed{\text{NNom}}\boxed{\text{NGen}}\boxed{\text{NDat}}\boxed{\text{NAcc}}\boxed{\text{NInstr}}\boxed{\text{NPrep}}$ where *NPrep* is a noun in Prepositional Case (if there is any), etc. When there is no a noun in this case in the sentence the corresponding position of the structure may be filled up by some auxiliary information about the fact that there is no a noun in this case in the sentence but in principal it can be placed there, or there is no a noun in the given case and it cannot exist there at all.

The predicate $P(v, n_1, ..., n_6)$, corresponds to this structure where v is a verb, $n_1, ..., n_6$ are nouns. The predicate is sixtiary since there are six cases in Russian.

4 Grammatical Predicates

There is one more way of introducing predicates - matching with parts of speech. We call such predicates grammatical predicates.

For example, $N(x, y_1, ..., y_n)$, x is noun, y_i are characteristics used for dividing nouns into several groups.

Record $N(x, y_1, ..., \underset{i}{0}, ..., y_n)$ means absence of i-characteristic.

If characteristics $y_1, ..., y_n$ are alternative, we will denote this as $N(x, y)$, where $y = y_1$, if x has characteristic y_1;...; $y = y_n$, if x has characteristic y_n.

Let's take a look at noun number (singular or plural forms) as an example: it is an alternative characteristic since nouns cannot be in singular and plural form at the same time. However the noun can exist in different cases simultaneously (метро), have masculine and feminine gender (плакса), be animate and inanimate (пень) etc. We don't regard these characteristics as alternative.

Because of this, the XOR operation is defined in a different way. For predicates of the form $P(x, y_1, ..., y_n)$ the XOR is defined as conjunction of disjunctions, for example:

$Prep_1(x, y)$ means that prepositions are divided by their origin into $y =$ "непр", i.e. x is an underivative (prototypal) preposition and $y =$ "пр", i.e. x is a derivative preposition. Derivative prepositions are divided into

a) $Prep_1^1(x)$ – derived from an adverb (adverbial) (близ, около, сквозь etc.);

b) $Prep_1^2(x)$ – derived from a noun (nounal) (вследствие, по пути, по причине etc.);

c) $Prep_1^3(x)$ – derived from a verb (verbal) (благодаря, спустя etc.).

In particular we obtain

$$\begin{aligned}&(\forall x)\Big(\ Prep_1(x, np) \leftrightarrow\\ &\quad\leftrightarrow \mathop{\&}_{\substack{1 \le i, j \le 3\\ i \ne j}} ((Prep_1^i(x) \& \neg Prep_1^j(x)) \vee (Prep_1^j(x) \& \neg Prep_1^i(x)))\ \Big).\end{aligned}$$

For the predicates of the type $P(x, y)$ this operation coincides with the usual "or". For example:

$N_5(x, y)$, $y =$ "отвл", if the noun is abstract, $y =$ "конкр", if the noun is concrete (they represent individual objects, living creatures and some phenomena of environment).
$(\forall x)\ (N_1(x, \textit{собст}) \to \neg\ (N_5(x, \textit{отвл}) \vee N_5(x, \textit{конкр})))$ or
$(\forall x)\ ((N_5(x, \textit{отвл}) \vee N_5(x, \textit{конкр})) \to N_1(x, \textit{нар}))$ — these formulas mean that abstract and concrete nouns are nominal ones.

5 Predicates Associated with Sentence Parts

Furthermore, one can introduce **predicates associated with sentence parts**. Unary predicates of the sentence parts: $P_{sub}(x)$, where x is subject; $P_{pred}(x)$, where x is predicate; $P_{adv}(x)$, where x is adverbial modifier.

Note that a notion "predicate"is used in two senses: as usual and as a grammatical notion. In the second case, it is a word or a sequence of words, i.e. a sentence part. Here we do not consider the second order predicates.

Binary predicates of the sentence parts: $P_{sub}(x, y)$, x – subject; $P_{pred}(x, y)$, x – predicate; $P_{adv}(x, y)$, x – adverbial modifier; where y is a word or word-combination determined (explained).

It is possible to achieve formula representation of these predicates considering x, y as words or word-combinations. Upper index of Q in brackets – predicate arity (quantity of predicate places), lower index of Q shows to what part of the sentence we ask a question.

1. The determined word is a subject

$(\forall x, y)\left(Q_1^{(2)}(x, y) \leftrightarrow (P_{sub}(x, y) \& P_{sub}(x) \& P_{pred}(y))\right)$ – it is possible to raise a question from a subject to a predicate.

2. The determined word is a predicate

$(\forall x, y)\left(Q_2^{(2)}(x, y) \leftrightarrow (P_{adv}(y, x) \& P_{pred}(x) \& P_{adv}(y))\right)$ – it is possible to raise a question from a predicate to an adverbial modifier.

In a general case formulas for n heterogeneous sentence parts may be written in the following form:

$$(\forall x, y_1, ..., y_n)\left(Q_1^{(n+1)}(x, y_1, ..., y_n) \leftrightarrow \left(\mathop{\&}_{i=1}^{n} P_{attr}(y_i, x) \& P_{sub}(x) \& \mathop{\&}_{i=1}^{n} P_{attr}(y_i)\right)\right)$$

describes a case with heterogeneous attributes of the subject.

$$(\forall x, y_1, ..., y_n)\left(Q_2^{(n+1)}(x, y_1, ..., y_n) \leftrightarrow \left(\mathop{\&}_{i=1}^{n} P_{adv}(y_i, x) \& P_{pred}(x) \& \mathop{\&}_{i=1}^{n} P_{adv}(y_i)\right)\right)$$

describes a case with heterogeneous adverbial modifiers of the predicate.

Below several examples of sentences are presented in the form of predicates

I. Купить машину нам не по средствам.

$N(\textit{машину})$, $ProN(\textit{нам})$, $N(\textit{средствам})$, $V(\textit{купить})$, $Prep(\textit{по})$, $PartL(\textit{не})$;
$P_{pred}(\textit{купить})$, $P_{obj}(\textit{машину})$, $P_{obj}(\textit{нам})$, $P_{adv}(\textit{не по средствам})$,
$P_{obj}(\textit{нам, купить})$, $P_{adv}(\textit{не по средствам, купить})$,
$P_{obj}(\textit{машину, купить})$;

1. $(\forall x, y)\left(Q_2^{(2)}(x,y) \leftrightarrow (P_{adv}(y,x)\&P_{pred}(x)\&P_{adv}(y))\right)$ – if $x =$ "купить", $y =$ "не по средствам";

2. $(\forall x, y_1, y_2)\Big(Q_2^{(3)}(x, y_1, y_2) \leftrightarrow$
$\leftrightarrow (P_{obj}(y_1,x)\&P_{obj}(y_2,x)\&P_{pred}(x)\&P_{obj}(y_1)\&P_{obj}(y_2))\Big)$ – if $x =$ "купить", $y_1 =$ "машину", $y_2 =$ "нам".

II. Она шла нетвердой походкой.

$ProN$(*она*), V(*шла*), N(*походкой*), Adj(*нетвердой*);

P_{sub}(*она*), P_{pred}(*шла*), P_{attr}(*нетвердой*), P_{adv}(*походкой*), P_{sub}(*она, шла*), P_{adv}(*походкой, шла*), P_{attr}(*нетвердой, походкой*), P_{pred}(*шла, она*);

1. $(\forall x, y)\left(Q_1^{(2)}(x,y) \leftrightarrow (P_{sub}(x,y)\&P_{sub}(x)\&P_{pred}(y))\right)$ – if $x =$ "она", $y =$ "шла";

2. $(\forall x, y)\left(Q_2^{(2)}(x,y) \leftrightarrow (P_{pred}(x,y)\&P_{pred}(x)\&P_{sub}(y))\right)$ – if $x =$ "шла", $y =$ "она";

3. $(\forall x, y)\left(Q_2^{(2)}(x,y) \leftrightarrow (P_{adv}(y,x)\&P_{pred}(x)\&P_{adv}(y))\right)$ – if $x =$ "шла", $y =$ "походкой";

4. $(\forall x, y)\left(Q_4^{(2)}(x,y) \leftrightarrow (P_{attr}(y,x)\&P_{adv}(x)\&P_{attr}(y))\right)$ – if $x =$ "походкой", $y =$ "нетвердой".

III. Самолет, пролетающий над нами, скрылся в облаках.

N(*самолет*), N(*облаках*), $ProN$(*нами*), V(*скрылся*), $PartP$(*пролетающий*), $Prep$(*над*), $Prep$(*в*);

P_{sub}(*самолет*), P_{pred}(*скрылся*), P_{attr}(*пролетающий над нами*), P_{adv}(*в облаках*), P_{adv}(*в облаках, скрылся*), P_{sub}(*самолет, скрылся*), P_{pred}(*скрылся, самолет*), P_{attr}(*пролетающий над нами, самолет*);

see II.1. – if $x =$ "самолет", $y =$ "скрылся";

$(\forall x, y)\left(Q_1^{(2)}(x,y) \leftrightarrow (P_{attr}(y,x)\&P_{sub}(x)\&P_{attr}(y))\right)$ – if $x =$ "самолет", $y =$ "пролетающий над нами";

see II.2. – if $x =$ "скрылся", $y =$ "самолет";

see II.3. – if $x =$ "скрылся", $y =$ "в облаках".

As an intermediate result we get that it is possible to determine syntactic valencies of a word by means of predicates introduced above.

6 Matching Text with Streams

Let us consider now not a separate sentence but text.

There is a text, i.e. final set of sentences, $p_1 p_2 ... p_N$, at the input. Some streams are formed at the output:

$S_1 = < s_{11}, s_{12}, ..., s_{1m_1}, ... >$

. . . .

$S_k = < s_{k1}, s_{k2}, ..., s_{km_k}, ... >$

An elementary auxiliary stream consists of well-ordered pairs

$< 1, p_1, 2, p_2, ..., N, p_N >$, where the first multiplier is the sentence number the second one is the sentence itself.

Information about word-formation may be placed in streams like $< h, k_1, L_1, k_2, L_2, ... >$, where h is the stream heading, for instance a selected suffix; k_i is the sentence number, where the word with this suffix appears (i.e. k_i are numbers not for all sentences but only for those, where these words appear); L_i is the list of words with the given suffix appearing in the sentence.

Streams may be associated with lexical functions, too. We will also construct finite models matching with source text in the form of streams.

For instance, let's pick all nouns from sentences and write them in stream $< 1, n^1_1, ..., n^1_{l_1}; 2, n^2_1, ..., n^2_{l_2}; ... >$, where sentence numbers and lists of nouns present in this sentence are written in series (l_i - list size). Let's write this stream in other way $<< 1, n^1_1 >, ..., < 1, n^1_{l_1} >, < 2, n^2_1 >, ..., < 2, n^2_{l_2} >, ... >$.

Denote $C = \{< t, n^t_j > | t = \overline{1, N}, j = \overline{1, l_t}\}$ as set of all pairs that appear in the stream. The underlying sets of models will be ones of the following kind $C_0/ \sim$, where $C_0 \subseteq C$, $\sim$ is some kind of equivalence relation.

Equivalence relations will appear almost in the same way as they appear in Henkin's construction when proofing model existence theorem [7], i.e. pairs of the type $< t, c^t_j >$ $(t = 1, ..., N)$ may be considered as constants and depending on different statements about these constants we regard some of them as equivalent.

In a similar manner using the stream obtained it will be possible to apply the types omitting theorem [6] and, besides, to get some models as a result.

Let us note that while using Henkin's constructions it is essential to check consistency of corresponding theories at every stage. However, only partial testing for noncontradictory can be used while running computer processing of a natural language text.

For example we check that relations like "над"or "под"are really transitive; if it is said "white"about an object, then there isn't "black"statement anywhere in the sentence and so on.

7 Conclusion

Different approaches to representing semantics of natural language texts are of great interest now. That is why we have made efforts to analyze the sense of the text on a base on a structural analysis of sentences and a text as a whole ([8], [9]).

Large amount of predicates and logic formulas of the first order there were proposed for such analysis. However we note that in the main the given predicates and formulas are concerned with a grammatical and syntactic structure of sentences.

In future the results achieved may be studied and transformed by mathematical logic means. It gives us an opportunity to make a transition from syntactic to semantic level.

This work can be used for a creation of a text sense theory, and it is possible to apply the results of this work in a mathematical logic area and in linguistic investigations.

Thus, in spite of the fact that this work stage is absolutely necessary, it is important to note that semantic text structure has not been adequately reflected in achieved formulas up till now, and the following investigations are necessary.

We also note that the large volume of factual information from classical and mathematical linguistics, and mathematical logic was used at this simplest (in our opinion) stage. It tells about difficulty of this problem in the whole. Also in this article, we omitted questions connected with computer realizations.

References

1. Melchuk, I.A.: Experience of Theory of Linguistic Models like "Sence $< - >$ Text". Moscow (1974) (in Russian)
2. Apresyan, U.D.: Experimental Semantic Invesigation of a Russian Verb. Nauka, Moscow (1967) (in Russian)
3. Markus, S.: Set-theoretical Models of Languages. Nauka, Moscow (1970) (in Russian)
4. Beloshapkova, V.A.: Modern Russian Language: Manual for Philology Students of Institutes of Higher Education. Azbukovnik, Moscow (1997) (in Russian)
5. Rosental, D.E.: Modern Russian Language: Manual for Philology Students of Institutes of Higher Education. MSU Edition, Moscow (1971) (in Russian)
6. Sokirko, A.V.: The semantic dictionaries in automatic text processing. Thesis. MGPIIA, Moscow (2000) (in Russian)
7. Sacks, G.E.: Saturated Model Theory. W. A. Benjamin Inc. (1972)
8. Batura, T.V., Erkayeva, O.N., Murzin, F.A.: To the problem of analysis of texts in a natural language. "New information technologies in science and Education", Institute of Informatics Systems, Novosibirsk (2003) (in Russian)
9. Batura, T.V., Murzin, F.A.: Logical methods of the sense representation for a text in a natural language. "New information technologies in science and Education", Institute of Informatics Systems, Novosibirsk (2003) (in Russian)