Control and Cybernetics

vol. **22** (1993) No. 1/2

Description of a subset of Polish natural language: a mathematical model

by

Stanisława Guszkowska

Warsaw University of Technology
Warsaw
Poland

The paper concerns formalization of a specific subset of natural (Polish) language, called topic oriented language. Language area limited for the purpose of conduct of a dialogue on a given topic during the didactic process has been considered. Formalization of a topic oriented natural language was carried out by means of graph theory.

On the basis of graph theory a formal description has been provided of the structure of information contained in a fixed dialogue topic. The paper puts forward, under certain assumptions, the definition of a sentence diagram and the definition of a dialogogram — an object checking and generating correct single sentences of a natural language. The model of a topic oriented natural language has been described as a set of fixed dialogograms. It is a part of the model of a dialogue on a chosen topic during computer–assisted teaching of elements of certain subjects.

Introduction

Natural language constitutes one of the most complex systems used by man. According to linguists, Jodłowski (1976), a huge number of concrete individual utterances can be reduced to a relatively small number of formal patterns.

Natural language can approach formal language if the set of its words and syntactic structures is reduced. Certain limitations are reached if the language necessary for conducting a dialogue on a fixed topic is taken into account, as the sentence syntax is determined by its constituents, their arrangement and interrelations. These, in turn, depend mostly on the professional orientation of

In memoriam Professor Edmund Lipiński

the people uttering the given sentence. Each professional group make use of a specific vocabulary and formulate their utterances in a characteristic way. It can be assumed that they make use of "their own language" which is a subset of natural language. We shall call this subset a topic oriented language.

The article will be concerned with the language area limited to the needs of conducting a dialogue while teaching elements of sciences such as mathematics, physics and programming languages. These examples apply to dialogues conducted when teaching the elements of Pascal programming language.

The topic orientation of a language does not limit the syntax to an extent allowing for its formalization. An artificial limitation of syntax can be achieved by dividing information into small bits which shall later be called information modules. A set of basic knowledge constitutes an information module; a set of information modules determines the dialogue topic, the set of topics determines in turn the subject and range of the dialogue.

The problem of dividing information into modules boils down to determining logical and substantial relations between notions and information items constituting the subject matter of the dialogue. Therefore, one has to:

1. define the problem covered in the subject matter of the dialogue; the subject problem title should precisely define the subject matter of the dialogue,
2. isolate the most essential parts of the problem and aim at specifying elementary notions.

Such a system results in a set of elements constituting a fixed order. Chapter I of the paper gives an account of the attempt to formally describe this order and to unify the description of information the carrier whereof are the sentences of a natural language. Chapter II provides a definition of a certain kind of formal diagrams of a natural language (i.e. Polish) adopting certain assumptions. Chapter III attemps to formally describe a topic oriented natural language i.e. to determine its pattern for the purpose of its use in a model of a dialogue on a given topic when teaching elements of certain subjects.

CHAPTER I

Terms necessary to describe a set of information contained in the fixed dialogue topic in general and for information module description in particular will be presented below.

In the proposed formalization, the structure of the above mentioned information will be represented by mathematical objects of tree type.

The following definition of the tree is adopted for our purposes:

The tree is such a set DZ in which:

1. there exists a marked element called a root,
2. the remaining elements belong to disconnected subsets $DZ_1, DZ_2, \ldots, DZ_m$ of the set DZ. Subsets $DZ_1, \ldots, DZ_m$ are also trees. They are called sub-

trees defined by the element marked in point 1. (root).

Elements of set DZ are called vertexes. Each vertex unambiguously defines the tree with a root in this vertex; it is called a tree defined by this vertex. The vertex is called a leaf if the tree it defines is a one–element set.

A non–negative integer n is called the vertex depth:

$$n = \begin{cases} 0 & \text{if the vertex is the tree root} \\ k+1 & \text{if the vertex is the root of a subtree} \\ & \text{defined by another vertex of depth } k. \end{cases} \quad (1)$$

Tree leaves represent basic elements of the problem dealt with in the dialogue. They represent entries, formulas, axioms etc. Vertexes lying at a depth smaller by 1 than leaves define subtrees consisting only of the root and leaves. They represent the title of the problem which can be directly divided into elementary units. The subject range determined by this title is in turn determined by the subtree consisting of the root and leaves and hence by this root and the leaves. The root represents information items connected with the title of the problem. They are, as in the case of the leaves, composed of definitions, formulas etc. applying, however, to the problem as a whole and not to its constituents. Therefore, each element of the subtree in question represents elementary units of knowledge, some of them applying to the title of the problem (i.e. to the problem as a whole), others applying to the problem constituents. Items of information corresponding to the subtree in question belong to a proper category and are characterized by certain features. Their set is precisely defined — it is the sum of the set of information items corresponding to the leaves and the set of information items corresponding to the root. Utterances — sentences of a natural language — on the problem represented by the subtree are carriers of this information set. Therefore the semantic value of these utterances as well as their syntax is determined. This allows for the realization of a dialogue on the problem represented by the subtree consisting of the root and leaves. We call the information module a subtree consisting of the root and leaves.

If we assume that the leaves are at depth t, the information module is called a subtree defined by the vertex of depth $t-1$. The titles of the dialogue will be located at depth $t-2$. It is suggested that the subject area of the dialogue be chosen in a way which makes it possible to move from the problem determining the dialogue range to dialogue topics and from the dialogue topics to information modules.

Example I

Let us consider one of the topics which occur when studying the Alphabet programming language. The tree of the topic "Alphabet" of Pascal programming language is presented in Figure 1. It is a subtree of the topic: "Elements of Pascal programming language".

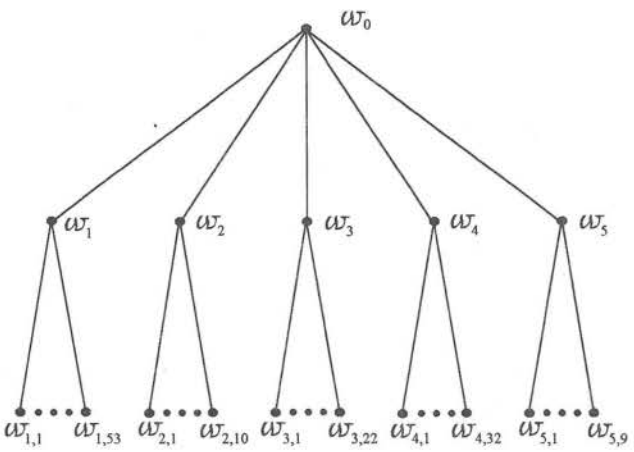

Figure 1. The tree of the topic "Alphabet" of Pascal programming language

w_0 – Alphabet of Pascal programming language
w_1 – letters
w_2 – decimal digits
w_3 – sexadecimal digits
w_4 – special characters
w_5 – double characters

$w_{1,1}$ – a
... . $\left\{ \begin{array}{l} \text{a b c d e f g h i j k l m n o p q r s t u v w x y z} \\ \text{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z} \\ \text{\_} \end{array} \right\}$
$w_{1,52}$ – Z
$w_{1,53}$ – _

$w_{2,1}$ – 0
... . { 0,1,2,3,4,5,6,7,8,9 }
$w_{2,10}$ – 9

$w_{3,1}$ – 0
... . { 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f,A,B,C,D,E,F }
$w_{3,22}$ – F

$w_{4,1}$ – !
... . { ! " # S % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ' { | } ~ }
$w_{4,32}$ – }

$w_{5,1}$ – :=
... . { := <> <= >= .. (* *) (. .) }
$w_{5,9}$ – .)

The tree root represents the title of the topic "Alphabet" of Pascal programming language ("Basic characters of Pascal"). At depth 1 the following vertexes can be found: w_1 – letters, w_2 – deximal digits, w_3 – sexadecimal digits, w_4 – special characters, w_5 – double characters. Each of them defines a subtree of a fixed number of leaves. Therefore these subtrees represent the respective information modules: letters, decimal digits, sexadecimal digits, special characters, double characters.

Example II

Let us describe the information set for the module "letters". The subtree "letters" has 53 leaves. 52 leaves stand for small and capital letters of the English alphabet while the 53rd leave stands for the underlining character. Each of the leaves informs that the object standing for the leaf is a letter of Pascal. From this message the following two elementary information items can be extracted:

- the object — in this case a proper character — standing for a given leaf belongs to the set of basic characters of Pascal (i.e. of the Alphabet),
- the object which stands for a particular leaf is a letter in the set of Pascal basic characters (of the Alphabet).

The information contained in the vertex "letters" consists of the following elementary information items:

1. the set "letters" belongs to "The set of basic characters of Pascal programming language",
2. the set "letters" consists of small and capital letters of the English alphabet and of the underlining character,
3. elements not enumerated in 2. do not belong to the set "letters".

The information on the subtree "letters" applies to one of the 53 leaves of this subtree or to its vertex.

The information module "letters" is the sum of the set of information on the 53 letters and the set of information concerning the title of the problem i.e. the entry "letters". A distinctive feature of these information items is the property of membership of a given element in the proper set. Also, the information on other information modules of the topic "Alphabet" has this property. The property determines both the semantics of utterances relative to this information module and their syntax, for they are carriers of these particular information items. In this way, this property determines the set of utterances relative to a given module.

The specification of properties characteristic of the set of information contained in the information module determines the utterance set for the whole module.

CHAPTER II

In this chapter we shall indicate particular properties of utterances relative to particular information modules of a given topic of the dialogue on a fixed subject. We shall also provide definitions necessary for drawing up sentence diagrams and establishing their properties.

Let us analyze specimen utterances relative to the information module of the topic "Alphabet" chosen from Example I .

Example III

We take into account certain types of utterances relative to the information module "letters". They are determined by transformation features contained in this module and described in Example II.

The following utterances can be connected with the subtree leaf "the underlining character":

1. Does the character _ belong to letters ?
   Does the character _ belong to the alphabet ?
2. The character _ belongs to letters.
   The character _ belongs to the alphabet.

2a. The sign _ does not belong to letters.
   The sign _ does not belong to the alphabet.

If we replace, in these sentences, the underlining character (_) by any letter, we will obtain utterances relative to some other leaf of this subtree, namely the leaf representing the given letter.

Therefore utterances like:

1. Does the character ... belong to letters ?
   Does the character ... belong to the alphabet ?
2. The character ... belongs to letters.
   The character ... belongs to the alphabet.

2a. The character ... does not belong to letters.
   The character ... does not belong to the alphabet.

can only differ in the word in the fixed position indicated by ... . They will correspond to one of the 53 leaves of the subtree "letters" depending on which of the elements of the set "letters" is in the fixed position.

The same goes for utterances of other types relative to the leaves of the subtree "letters", for these letters contain the same information about objects which they stand for.

The information contained in the vertex "letters" also consists of determinate information items (Example II). They are also carried by utterances of a determinate type which belong to a determinate set. These utterances can be exampified by eg.:

3. Write letters.
4. Which letters belong to the alphabet ?

The sum of sets of utterances relative to leaves and utterances relative to the vertex of the subtree "letters" is a set of utterances connected with the module "letters". These utterances are formulated in a proper way as they are carriers of fixed information items. Their syntax and semantics are determined.

Example IV

Let us consider the information module "double characters" of the topic "Alphabet" from Example I.

The subtree "double characters" has 9 leaves which stand for characters: := <> <= >= .. (* *) (. .).

The information contained in the leaves of the subtree "letters" are also contained in the leaves of the subtree "double characters" but in this case they apply to each of the double characters. Besides, the leaves of the subtree "double characters" contain an additional elementary information item i.e. the object represented by a given leaf has a certain meaning. Therefore utterances relative to the leaves of the subtree "double characters" are carriers of information which has the same properties as utterances relative to the leaves of the subtree "letters" and has a property determining the meaning of the given object.

If we replace the word "letter" in the utterances from Example III by the word "double character" in a proper form, we will obtain utterances relative to the subtree "double characters"; in this case, in the fixed position (indicated by ...) there may appear elements of the subset which contains double characters.

Therefore utterances such as:

1. Does the character ... belong to double characters ?
   Does the character ... belong to the alphabet ?
2. The character ... belongs to double characters.
   The character ... belongs to the alphabet.

2a. The character ... does not belong to double characters.
   The character ... does not belong to the alphabet.

3. Write double characters.
4. Which double characters belong to the alphabet ?

belong to a set of utterances assigned to the information module "double characters".

The following two sentences are examples of utterances relative to the meaning of the object represented by a given leaf of the subtree "double characters":

5. What does the character ... mean ?
6. The character ... is a relation character.

Utterances relative to the subtree "double characters" are formulated in a proper way, for they are carriers of the determined information. Their syntax and semantics as well as their set are determined. The set of utterances relative to the subtree "double characters" contains a set of utterances relative to the subtree "letters" if we take into account the fact that the word **letter** may be substituted by the word **double character** and that elements of particular sets

may appear in the same sentences (in this case elements of the subset "letters" are assigned to the word **letter** and elements of the subset "double characters" are assigned to the word **double character**).

What is noteworthy here is the subtree "special characters". Information properties contained in this subtree are identical with information properties contained in the subtree "double characters". Therefore, the sets of utterances relative to these trees will be identical if we only treat these utterances schematically, marking positions in which characters or words belonging to certain sets can appear.

The above examples indicate certain properties of utterances relative to particular information modules of a determined topic. Analysis of these utterances shows that:

- words and whole phrases are repeated in them,
- certain words appear in fixed positions,
- in the set of utterances relative to a given information module one can isolate a set of sentences differing only in words in the fixed position,
- in the set of utterances relative to different information modules one can isolate:
  a) a subset of sentences differing only in words in the fixed position
  b) a subset of identical sentences.

The isolation of properities of this kind provides a basis for an analysis of a topic oriented natural language as a language close to a formal one. This allows for an application of proper mathematical apparatus to describe this subset of a natural language. Adaptation of selected definitions from the graph theory and the formal languages theory makes it possible to formulate definitions necessary for the formalization of a topic oriented natural language.

We adopt the following terms:

1. A word is a finite sequence of symbols different from the space of the set Σ,
2. A finite nonempty set of words $S = \{s_1, \ldots, s_m\}$ is called a dictionary, where m is the number of words in the dictionary.
3. A sentence over dictionary S is a finite (possibly empty) sequence of words of dictionary S. Empty sentence is marked as E, and the set of all sentences is marked as S^* ($E \equiv$ space).
4. Composition (concatenation) of sentences $z' = s'_1 \ldots s'_m$ and $z'' = s''_1 \ldots s''_n$ is a sentence $z = z'||z'' = s'_1 s'_2 \ldots s'_m s''_1 s''_2 \ldots s''_n$.
5. Sentence $z = s_1 \ldots s_m$ is an final sentence if $s_m \in \{., ?, !\}$ (The last word of the sentence is a symbol which ends the sentence in a natural language).
6. If S is a dictionary then every subset J of S^* is called a language over dictionary S.
7. Only a language which contains correct final sentences is considered in this work. Such a language is marked as JP and sentences of JP language are treated as a composition of words of dictionary S.

8. A word which does not influence the semantic value of a sentence is called a marked word in the sentence.
   The word **following** is a marked word. Sentences: "The following letters belong to the alphabet: $a, b, \ldots z, A, B, \ldots Z$." and "Letters $a, b, \ldots z, A, B, \ldots Z$ belong to the alphabet" have the same semantic value and the word **following** has no influence upon this value.
9. A graph is an ordered four: $G =< W, WP, WK, H >$, where $W = \{w_0, w_1, \ldots w_z\}$ — a finite set of vertexes; WP — set of initial vertexes; $WP \subset W$; WK — set of terminal vertexes; $WK \subset W$; $H:(W - WK) \longrightarrow 2^W$ — mapping of the next vertex, if $w \in WK$ then $H(w)$ is indeterminate.
10. Sequence $w_{1,1}, w_{1,2}, \ldots w_{1,m}$ such that $w' = w_{1,1}$, $w'' = w_{1,m}$, and for $i = 1, 2, \ldots m-1$ $w_{1,i+1} = H(w_{1,i})$ is called path $DR(w', w'')$ from vertex w' to vertex w'' in graph G.
11. Graph $G =< W, WP, WK, H >$ which meets the following conditions is called the flow graph: $WP = \{w_0\}, WK = \{w_k\}, \forall w \in W - \{w_0\}$ $EDR(w_0, w)$, $\forall w \in W - \{w_k\}$ $EDR(w, w_k)$.
12. Graph $G = G_1 \cup G_2$, $G =< W, WP, WK, H >$ determined as follows: $W = W_1 \cup W_2$, $WP = WP_1 \cup WP_2$, $WK = WK_1 \cup WK_2$, $\forall w \in W$ $H(w) = H_1(w) \cup H_2(w)$ is called the set sum of graphs $G_1 =< W_1, WP_1, WK_1, H_1 >$ and $G_2 =< W_2, WP_2, WK_2, H_2 >$.

In order to determine sentence diagrams each sentence is considered as a subset of words of a definite dictionary, the words being set up in a certain order.

Definition I

A tuple $GZ =< WZ, WPZ, WKZ, WWZ, WNZ, HZ >$ is a graph of sentence:

$WZ = \{w_1, w_2, \ldots w_k\}$ — a set of vertexes corresponding to words of the sentence

$WPZ = \{w_1\}$, w_1 — vertex corresponding to the initial (beginning) word of the sentence, $WPZ \subset WZ$;

$WKZ = \{w_k\}$, w_k — vertex corresponding to the final (ending) symbol of the sentence, $WKZ \subset WZ$;

WWZ — set of vertexes determining in the sentence the position in which words of particular sets (Examples III, IV) appear; we shall call these vertexes multiple vertexes, $WWZ \subset WZ$, $WWZ \cap WPZ = \emptyset$, $WWZ \cap WKZ = \emptyset$;

WNZ — set of vertexes corresponding to marked words, i.e. words whose function is only syntactical and which not interfere with the semantical value of the sentence; we shall call these vertexes marked vertexes, $WNZ \subset WZ$, $WNZ \cap WPZ = \emptyset$, $WNZ \cap WKZ = \emptyset$;

$HZ:(WZ - WKZ) \longrightarrow 2^{WZ}$ — mapping defining the order of words of sentence. This mapping is not defined for $w \in WKZ$.

The graph of sentences is the flow graph. Sentences of different order are represented by different graphs.

Definition II

The sum of graphs of sentences is called sentence diagram (the pattern of sentences). Sentences of given epistemic structure for which initial vertexes are identical are joined into a sum.

Note:

Syntactical expressions called epistemic structures were determined as a result of applying a parsing criterion leading eventually to an epistemic analysis of a topic oriented natural language. The epistemic analysis consists in isolating in a sentence the members corresponding to forms of reflection of reality. The type of epistemic structure is determined by the utterance character as regards the inventory of its items, their grammatical character, configuration and interrelations. The epistemic analysis and determination of epistemic structures constitutes a very broad issue which is the subject of [8]. In this paper we assume that the sentences under consideration have a definite epistemic structure. We also assume that the sentence diagram is a sum of sentence graphs of identical initial vertexes.

Example V

Let us consider sentences from Example III and IV. Diagrams of sentences of types 1.–4. are presented in Figures 2 and 3.
In these diagrams words corresponding to particular vertexes differ in two positions: in the position of multiple vertexes — sets of certain elements are assigned to these vertexes — and in the position corresponding to the subtree name. In the sum of these diagrams, in the position of the multiple vertex, there will remain a multiple vertex now represented by the sum of the sets: "letters" and "double characters"; in the position corresponding to the name of the subtree there will appear another multiple vertex — in this case its elements being the words:letters, double characters, in a proper form.

After having introduced multiple vertexes, the diagrams of utterances of types 1.–4. related to the subtree "letters" and the subtree "double characters" will have the form presented in Figure 4.

Depending on which element appears in the multiple vertex position in diagrams in Figure 5, they determine utterances connected with any information module of the topic "Alphabet" and with the title "Alphabet" itself.

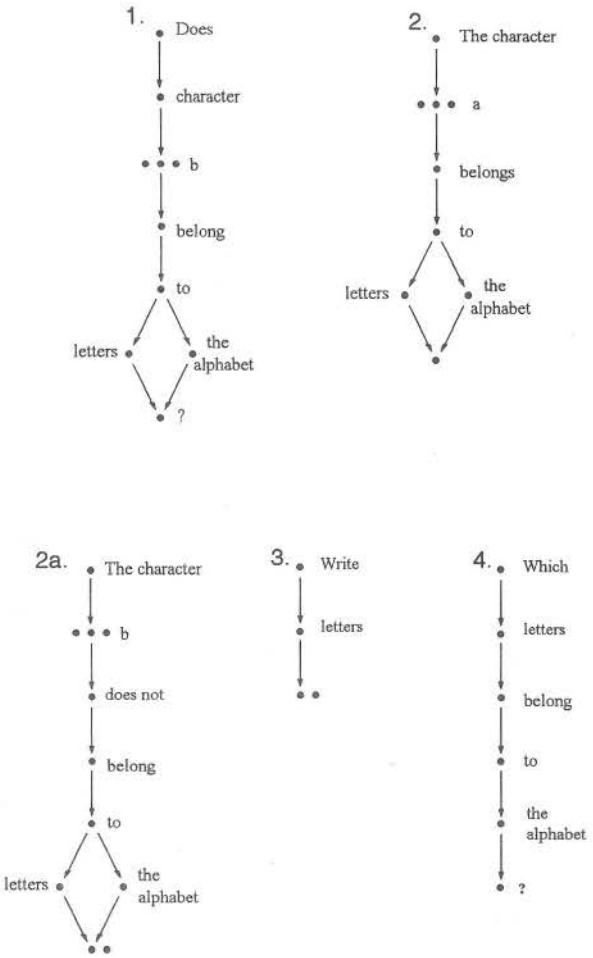

Figure 2. Example of sentence diagrams of the information module "letters":
... — the position of multiple vertexes;
$b \in \Sigma$, Σ — the set of symbols different from space;
$a \in A$, A — the alphabet of Pascal programming language.

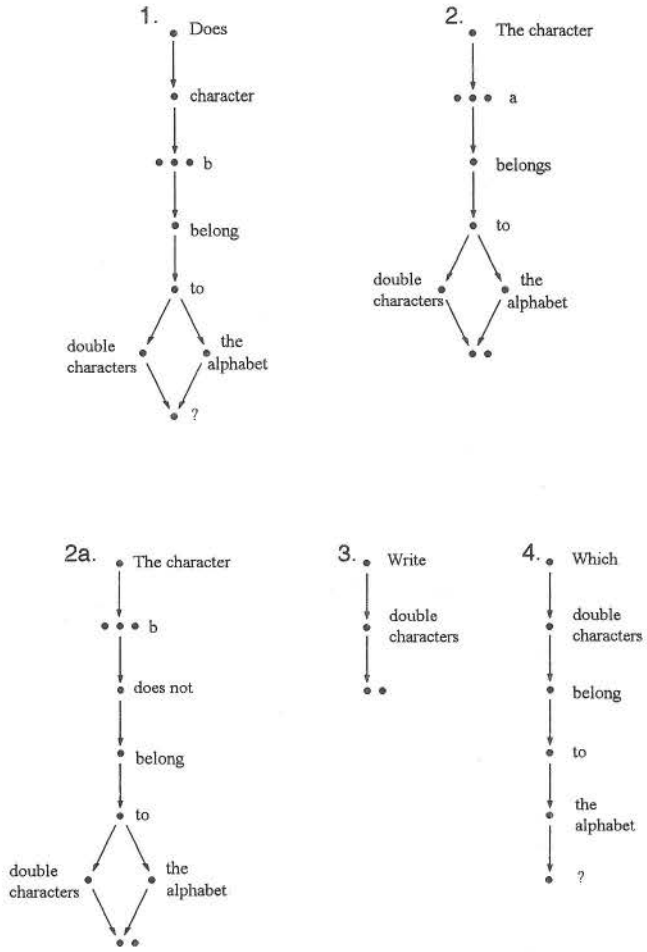

Figure 3. Example of sentence diagrams of the information module "double character":
... — the position of multiple vertexes;
$b \in \Sigma$, Σ — the set of symbols different from space;
$a \in A$, A — the alphabet of Pascal programming language.

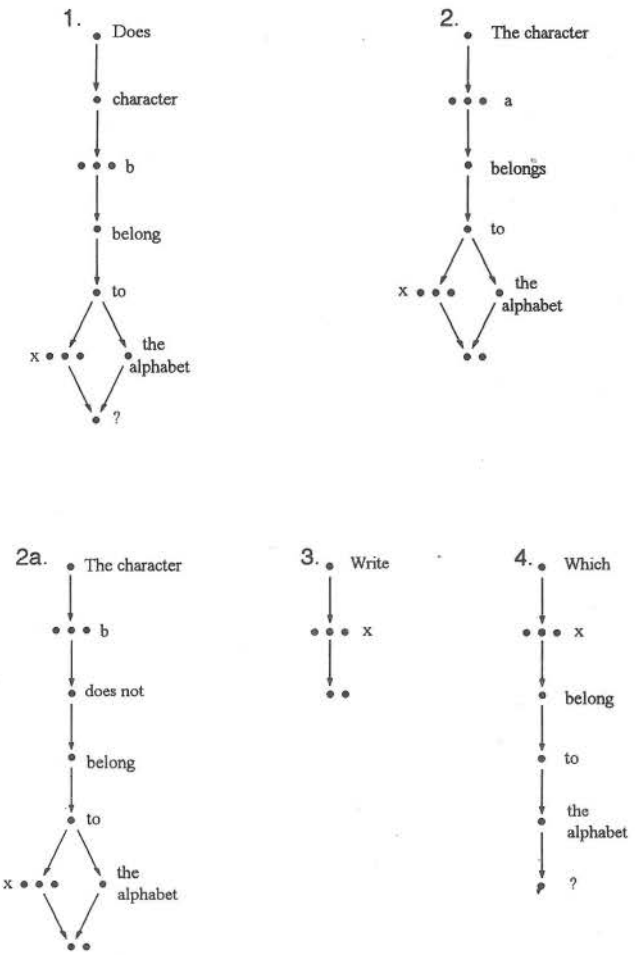

Figure 4. Example of sentence diagrams of the information modules "letters" and "double characters":
... — the position of multiple vertexes;
$b \in \Sigma$; Σ — the set of symbols different from space;
$a \in A$; A — the alphabet of Pascal programming language.
$x \in X$; $X = \{\text{letters, double characters}\}$.

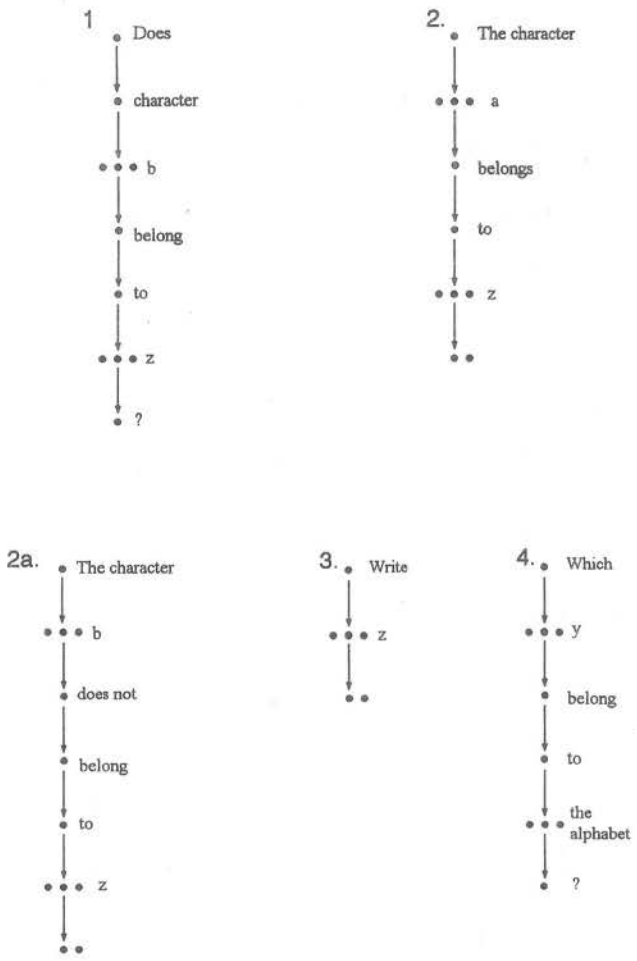

Figure 5. Example of sentence diagrams of the topic "Alphabet" of Pascal programming language:

... – the position of multiple vertexes;

$b \in \Sigma$; Σ – the set of symbols different from space;

$a \in A$; A – the alphabet of Pascal programming language:

$y \in Y$; $Y =$ {letters, double characters, sexadecimal digits, special characters, double characters}.

$z \in Z$; $Z =$ {letters, double characters, sexadecimal digits, special characters, double characters, the alphabet}.

Properties of sentence diagrams:

1. Graph vertexes standing for the same phrases in different sentences constitute one path in the diagram of these sentences.
2. A sentence diagram is a flow graph $G =< W, WP, WK, WN, WW, H >$, where
   $W = \{w_1, \ldots w_k\}$ — set of vertexes,
   $WP = \{w_1\}$, w_1 — initial vertex,
   $WK = \{w_k\}$, w_k — terminal vertex,
   WN — set of marked vertexes, $WN \subset W$,
   WW — set of multiple vertexes, $WW \subset W$,
   $H: (W - WK) \longrightarrow 2^W$ — mapping of the next vertex; if $w \in WK$ then $H(w)$ indeterminate.

CHAPTER III

On the basis of the above considerations we can isolate the following properties of sentences of a topic oriented natural language:

1. these sentences reflect certain formal patterns (diagrams),
2. sentences relating to a fixed information module correspond to a definite set of diagrams,
3. various sentences related to one or a few information modules correspond to identical diagrams.

Sentence diagrams were used to construct the so–called dialogograms.

Definition III

A dialogogram is an ordered tuple $D =< GS, S, t, f, k, h >$, where:
GS — sentence diagram; S — dictionary;
W — set of vertexes in sentence diagram GS, $W = WZ \cup \{w_0\}$ (WZ — defined in Definition 1);
$t: W \longrightarrow S$, $\forall w \in W \; t(w) = s$, $s \in S$,
s — a word determined by the vertex position in sentence diagram GS,
$t(w_0) =$ space $(t(w_0) = E)$,
if $w \in WWZ$ then $t(w) \in Q$, Q — a set of words assigned to the multiple vertex, WWZ — set of multiple vertexes, $WWZ \subset WZ$;
$f: W \to k$, $\forall w \in W \; f(w) = k$, k — concatenation, $||$ concatenation character;

$$k: S^* \times \{t(w)\} \to S^*, \quad k(s^*, t(w)) = \begin{cases} s^*||t(w) & \text{for } w \notin WNZ \\ s^*||E & \text{for } w \in WNZ, \end{cases}$$

WNZ — set of marked vertexes, $WNZ \subset WZ$;
$h: S^* \times (W - WKZ) \to W$, h — transition mapping connected with next vertex mapping:
$H(w) = \{w' \in W \;\; \forall z \in S^* w' = h(z, w) \text{ for } w \in W\}$.

A determinate set of dialogograms is assigned to an appropriate information module.

A set of dialagograms assigned to particular dialogue topics constitutes the model of a topic orientated language.

A dialogogram should be treated as a set of states to which, depending on the dialogue topic, the set of dictionary words is assigned — mapping t assigns to each vertex in the sentence diagram a word unambiguously defined by the position of that vertex; to a multiple vertex there is assigned a word which belongs to a proper set corresponding to that vertex.

Mapping k which performs concatenation of a proper sentence with a proper word and the mapping of transition to the next state are connected with each state. One begins from the zero state to which empty sentence (E) is assigned. In each next state one performs the concatenation of the sentence obtained in the previous state with the word assigned by mapping t to a vertex indicated by mapping h of the transition to this state. This procedure finishes when the final state is reached. Mapping t assigns the symbol of the end of the sentence to the terminal vertex. One obtains a word sequence (word composition) of dictionary S, which constitutes a sentence of natural language.

From the way mapping k is defined, it results that sentences differing in words which fulfil in those sentences a syntactical function only and do not influence their semantic value, may be realized by means of the same dialogograms.

Mappings t, h are determined in the definition of the dialogogram in a general way. Their full form is required when the concept of dialogograms is employed in the model of a dialogue on a fixed topic.

This model would consist of a set of dialogograms and a set of decision rules. Decision rules should describe the operation of mappings t, h and determine the method of choosing and exposing sentences which belong to a given language; this implies interpretation of information contained in these sentences. Notation of decision rules constitutes a problem in itself the discussion whereof is not the aim of this article.

Final remarks

A precise, formal description of a subset of natural language used in a dialogue related to even a narrow topic, supported by a sufficient number of examples and paying attention to special cases resulting from the specific character of a given language, requires a large dissertation.

This paper presents an outline of a method of analyzing a subset of a topic orientated natural language, in order to formally describe it. The article also proposes one of the methods of this description by means of a set of dialogograms.

The supplement contains definitions and theorems defining the properties of dialogograms. These definitions and theorems are the mathematical basis of the

proposed description; they indicate the possibility and provide justification of further research into the field.

The idea of dialogograms has common features with the so–called frames which are frequently used nowadays as a means of knowledge representation.

Making use of the experience of the authors of expert systems using frames will (hopefully) make it possible to develop new, effective methods of formal description of a the natural language.

References

BIELECKI J. (1989) Turbo Pascal 5.0, Warszawa

BOLTZ L., CICHY M., RÓŻAŃSKA L. (1982) Natural language processing, Warszawa (in Polish)

GUSZKOWSKA S. (1979) Information flow in multiaccessible conversational systems, Technical University, Łódź (in Polish).

JAYEZ J.H. (1982) Compréhension automatique du langage naturel, Paris

JODŁOWSKI ST. (1976) Principles of Polish language syntax, Warszawa (in Polish)

ŁĄCKA M. (1981) Information idenitification in multiacceessible conversational computer systems, Technical University, Warszawa (in Polish)

RATYŃSKA J. (1982) Automatic modelling of input information structures in the information systems", Technical University, Warszawa (in Polish)

TAFF E. (1979) Information identification in the information–receiver systems, Technical University, Warszawa (in Polish)

Knowledge engineering and expert systems (1990) Proc. of I Research Conference, Wrocław

Supplement

Definition 1.

Let $g: S^* \times W \to S^*$ be determined in the following way:

$$\forall z \in S^* \quad \forall w \in W \quad g(z,w) = [f(w)](z, t(w)),$$

$$\text{i.e.} \quad \forall z \in S^* \quad \forall w \in W \quad g(z,w) = k(z, t(w)).$$

The dialogogram can then be represented in the following form:

$$D =< GS, S, t, p, k, h > .$$

This definition is more convenient than the former, as regards examining the properties of dialogograms, and thus it will be employed in the Supplement.

Definition 2.

Production in dialogogram D is a function $p(g,h)$ described as follows: $p\colon S^* \times (W - \dot{W}K) \longrightarrow S^* \times W$,

$$p(< z,w >) =< g(z,h(z,w)),h(z,w) >=< k(z,t(h(z,w))),h(z,w) > .$$

Definition 3.

Realization R in dialogogram D giving sentence z_s with first element z_1, $z_1 \in S$, $z_s \in S^*$ is every sequence $p_0 \ldots p_s$ such that $p_0 =< z_0, w_0 >$, $z_0 = E$; $p_{i+1} = p(< z_i, w_{l,i+1} >) = p(p_i) =< z_{i+1}, w_{l,i+1} >$ for $i = 0, \ldots s-1$. Sentence z_s is called the result of realization R. Sequence $p_0 \ldots p_s$ gives final sentence if $w_s \in WK$.

Definition 4.

An ordered pair (z_1, z), $z_1 \in S$, $z \in S^*$ realizes dialogogram D if in dialogogram D exists realization R giving sentence z with first element z_1.

Definition 5.

Dialogograms $D =< GS, S, t, k, f, h >$ and $D1 =< GS1, S1, t1, k1, f1, h1 >$ are equivalent if for every pair (z', z) such that z is a final sentence realizing dialogogram D for given realization R there exists a realization $R1$ for which the pair (z', z) realizes dialogogram $D1$ and vice versa.

Theorem 1.

For every pair (z', z), $z' \in S$, $z \in S^*$, z — final sentence, realizing dialogogram D there exists exactly one realization $\{p_0 \ldots p_s\}$ such that $z' = z_1$, $z = z_s$.
Conclusion: Dialogograms determine sentences in an unambiguous way.

Definition 6.

For the given two dialogograms $D =< GS, S, t, k, f, h >$ and $D' =< GS', S', t', k', f', h' >$ a system of mappings (A, B, C, K, F, E) such that: $A\colon W \longrightarrow W'$; $B\colon S \longrightarrow S'$; $C\colon t \longrightarrow t'$; $K\colon k \longrightarrow k'$; $F\colon f \longrightarrow f'$; $E\colon h \longrightarrow h'$; is called a mapping of D into D'.

Theorem 2.

Given two dialogograms D and D' such that D' is a result of applying system of mappings (A, B, C, K, F, E) to D:

$$A\colon W \longrightarrow W', \quad A(W) = W \cup \{w_{b_1}, w_{b_2} \ldots w_{b_m}\} = W'$$

$$A(w) = \begin{cases} \{w_{b_1}, w_{b_2} \ldots w_{b_m}\} & \text{for } w = w^-, \quad w^- \neq w_0, \quad w^- \neq w_k \\ w & \text{for } w \neq w^-, \text{where} \end{cases}$$

w^- a vertex different from the initial or terminal vertex after which vertexes $w_{b_1}, w_{b_2} \ldots w_{b_m}$ are joined. The place numbered by number b in the graph of such a vertex is determined practically by means of the epistemic structure of the sentence realized by the path in dialogogram containing vertex w^-;

$$B: S \longrightarrow S', \quad B(S) = S \cup SN = S'$$

In practice, $SN \subset S$, hence the following assumption can be made: $B(S) = S = S'$

$$C: t \longrightarrow t' \quad t'(w) = \begin{cases} t(w) & \text{if } w \in W \\ s_n, s_n \in SN & \text{if } w = w_{b_j}, j = 1, \ldots m; \end{cases}$$

$$F: f \longrightarrow f' \quad f' = f;$$

$$K: k \longrightarrow k' \quad k'(s^*, s) = \begin{cases} k(s^*, s) & \text{if } s \notin SN \\ k(s^*, E) & \text{if } s \in SN \end{cases}$$

$$k'(s^*, t(w)) = \begin{cases} k(s^*, t(w)) & \text{if } w \notin WN \\ k(s^*, E) & \text{if } w = w_{b_j}, j = 1, \ldots m; \end{cases}$$

$$E: h \longrightarrow h' \quad h'(s^*, w) = \begin{cases} h(s^*, w) & \text{if } w \in W - \{w^-\} \\ w_{b_1} & \text{if } w = w^- = w_b \\ w_{b,j+1} & \text{if } w = w_{b_j}, j = 1, \ldots m-1 \\ w_{b+1} & \text{if } w = w_{b_m} \end{cases}$$

Dialogograms D and D' are equivalent.

Conclusion.

Sentences which differ only in words used solely for syntactical purposes and not influencing semantics can be realized by the same dialogograms.

Definition 7.

For given two dialogograms: $D' =< GS', S', t', k', f', h' >$ and $D" =< GS", S", t", k", f", h" >$. Composition of these dialogograms D', $D"$ is a dialogogram $D = D' \times D"$ such that $D =< GS, S, t, k, f, h >$, where $W = W' \cup W'' - \{w_0''\}$, and $w_0 = w_0'$, $WK = WK"$, and functions t, k, f, h are defined for every $w \in W$ as follows:

$$t(w) = \begin{cases} t'(w) & \text{for } w \in (W' - WK') \\ st & \text{for } w \in WK', st \in \{,,i\} \subset S \\ t''(w) & \text{for } w \in W'', \end{cases}$$

$$k(s^*, t(w)) = \begin{cases} k'(s^*, t'(w)) & \text{for } w \in W' \\ k''(s^*, t''(w)) & \text{for } w \in W'' \end{cases}$$

for every $s^* \in S^*$,

$$f(w) = \begin{cases} f'(w) & \text{for } w \in W' \\ f''(w) & \text{for } w \in W'', \end{cases}$$

$$h(s^*, w) = \begin{cases} h'(s^*, w) & \text{for } w \in (W' - WK') \\ w_1'' & \text{for } w \in WK' \\ t''(s^*, w) & \text{for } w \in W'' \end{cases}$$

for every $s^* \in S^*$.

A graph of dialogogram determined in this way is a pattern of sentences:

- exactly one initial vertex (w_0) belongs to it, $w_0 = w_0'$,
- exactly one final vertex (w_k) belongs to it, $w_k = w_k''$,
- for every $w \in (W - \{w_0\})$ there exists a path from w_0 to w

$$DR(w_0, w) = \begin{cases} DR'(w_0, w) & \text{for } w \in W' \\ DR'(w_0, w_k')DR''(w_1'', w) & \text{for } w \in W'' \end{cases}$$

- for every $w \in (W - WK)$ there exists a path connecting w and w_k

$$DR(w, w_k) = \begin{cases} DR'(w, w_k')DR''(w_1'', w_k) & \text{for } w \in W' \\ DR''(w, w_k) & \text{for } w \in W'' \end{cases}$$

The effect of application of dialogogram $D = D' \times D''$ corresponds to the effect of application of dialogograms D'' and then D'. Value of vertex $w_k' \in WK'$ of graph of dialogogram D' is replaced by conjunction and the sentence given by dialogogram D' is a initial sentence for dialogogram D''.

Conclusion.

1. If sentence z' is given by dialogogram D' and sentence z'' by dialogogram D'' then sentence z given by dialogogram $D = D' \times D''$ is the sentence obtained by composition of sentence z' and z'' i.e. $z = z'||z''$.
2. Composition of dialogograms is combined[1] but is not commutative (exactly as in the case of sentences).

Theorem 3.

For given dialogograms D^1, D^2, D^3, D^4 such that D^1 and D^3 as well as D^2 and D^4 are equivalent dialogograms $D = D^1 \times D^2$ and $D = D^3 \times D^4$ are also equivalent.

Dialogograms are used for correcting and setting singular sentences and also complex sentences connected by conjunction "and" or a comma. By defining multiple vertexes in graph of dialogogram the problem of determining of a wide,

[1] In the case of $st \in \{,, i\}$ connection is preserved for sentences of natural language, but commutability is not preserved because semantics of complex sentence may be changed or may even not be defined if the order of singular sentences — elements of a complex sentence — is changed. If a set of conjunctions is extended (by adding conjunctions if, while etc.) composition of sentences of natural language will not be combined.

practically not limited, set of sentences corresponding to given patterns is resolved. In such sentences in the places of multiple vertexes there may appear words from fixed sets or even any word.

Theorems 1, 2, 3 give important features of dialogograms, i.e.:
- uniqueness of realization of a sentence ,
- the possibility of simplification of graph of dialogogram without changing its effects,
- preservation of equality of composition of dialogograms.

A deeper and more detailed study of dialogograms features should be carried out, especially as regards:
a) sentences dependent on one another in a more general way than in theorem 2,
b) features of dialogograms realizing complex sentences.

These features enable a class of sentences realized by dialogograms to be widened. The proper organization of dictionaries is needed for fast and exact work of dialogograms. This constitutes one of the most effective means of language normalization.

Organization of dictionaries, however, is out of the scope of this paper.