Computer Science Journal of Moldova, vol.**23**, no.1(67), 2015

Modeling of meaning acquisition based on NL phrases using state transitions

Sergiu Creţu, Anatol Popescu

Abstract

This study aims to develop techniques for the meaning acquisition of the NL phrases. Our approach is based on modeling meaning extraction by analyzing lexical component, syntactic component and semantic component of the phrases written or spoken in NL. To assure necessary semantic precision we adopt a double-level interpretation of the NL phrases using formal language with well defined semantics. This fact is essential for our approach.

Keywords: NL semantics, meaning, extensional aspect, intensional aspect.

1 Introduction

Meaning of NL phrases could be extracted /derived from their four underlying components: 1. lexical component; 2. syntactic component; 3. semantic component; 4. pragmatic component.

Meaning of the lexical and syntactic components depends on additional factors: 1. timing (i.e., when the unit was written or pronounced); 2. location (i.e., where the unit was written or spoken); 3. modality (i.e., how the unit was written or spoken). These factors taken together constitute the pragmatic component of a phrase.

Relation of individual components (lexical, syntactic, semantic, pragmatic) to the overall meaning of the phrase is not straightforward. More precisely, the meaning of entire phrase is a function of the meanings of its components.

Here we propose an integrated theoretical framework defining relations between the lexical, syntactic, semantic components.

2 Preliminary considerations

To simplify further analysis, the syntactic, semantic and pragmatic components of phrases in the NL were redefined as competences.

Definition 2.1: The syntactic competence refers to the ability of the speaker (emitter, author) to generate correct linguistic phrases with or without meaning.

Definition 2.2: The semantic competence refers to the ability of the speaker (emitter, author) to establish semantic relations between the lexical and syntactic units of phrases in the NL.

Definition 2.3: The pragmatic competence refers to the ability of the speaker (emitter, author) to apply correct linguistic phrases in a proper syntactic-semantic context.

Relations between the competences are unclear and poorly defined. Therefore, to model the ability of the speaker (emitter, author) to generate complex phrases in the NL bearing meanings, three competence models were generated: 1. a syntactic model, describing the employed syntax; 2. a semantic model, storing the semantic component; 3. a pragmatic model, defining and describing the pragmatic component.

The syntactic competence model was developed using a categorial grammar [1], [2].

Definition 2.4: Categorial grammar G, defined for the vocabulary V, is a finite relation as follows:

$$G \subseteq V \times Cat(B),$$

where vocabulary V – a finite set with its elements representing the words of a NL (the terminal elements), B – a countable set of categories, including a special S – a set of basic categories (to be immediately defined), $Cat(B)$ – algebra of terms generated by operators "/" and "\" and containing the set B. If the grammar G defines a single category

for each element of the vocabulary V, it is considered to be a classical categorial rigid grammar.

Definition 2.5: I. For every vocabulary V of terminal elements two reduction rules can be applied to Definition 2.1:

1. FA (forward application) – $/(A,B)A \to B$.
2. BA (backward application) – $A\backslash(A,B) \to B$.

II. In general, a set of categories from $Cat(B)$ should be attributed to every element of the vocabulary V with the help of operators "/" and "\ ".

III. Definitions **I** and **II** are necessary and sufficient to generate language L:

$$L = \{t_1...t_n \in V^* \,|\exists A_i \in Cat(B) \wedge < t_i, A_i > \in$$
$$G \wedge A_1...A_n \xrightarrow[FA,BA]{}{}^* S \wedge (\forall i \in [1,n])\Big\}.$$

Example: Let "John expertly hoists the flag" be a phrase to be modeled. The classical categorial grammar (CCG) model of this phrase is composed of:

1. The basic categories of the set B: $B = \{N, CN, S\}$;
2. The vocabulary V: $V = \{John, flag, expertly, to\ hoist\}$;
3. The classical grammar G will be:

$$G = \{< \text{John}, N >, < \text{flag}, CN >, < \text{to hoist}, \backslash(N, /(CN,S)) >,$$
$$< \text{expertly}, /(\backslash(N, /(CN,S)), \backslash(N, /(CN,S))), \backslash(\backslash(N, /(CN,S)),$$
$$\backslash(N, /(CN,S))) >\}.$$

To efficiently interpret (i.e. to determine the meaning) a NL sentence it should be converted to a logical object. The used logical language

should be a typed one. That is, for each logical object we are to assign its type. In general, the type is a label that refers to a subset of elements belonging to a set containing all the elements in use for interpretation.

For example, the elements of vocabulary V containing the NL words may be considered as belonging to the so called Universe set [3].

Definition 2.6: The $Type$ set expression is a minimal set which includes the following elements:

1. $e \in Type$. Element e denotes the individuals – the elements belong to the Universe set.

2. $t \in Type$. Element t denotes just only two values: true and false, also belonging to the Universe set.

3. If $a \in Type$ and $b \in Type$, then $< a, b > \in Type$,

where $< a, b >$ – a function with its definition domain D_a (a set of elements having the type a) and variation domain D_b (a set of elements having the type b).

For example, the type expression $< e, t >$ refers to a subset of Universe set individuals and $<< e, t >, t >$ is an expression that denotes a second degree predicate, i.e. it represents the definitions of the set of subset of Universe set individuals.

The logical (formal) language used for the interpretation of NL sentences has then two components: 1. the syntactic component; 2. the semantic component. The syntactic component comprises:

1. A set containing all the types for a given vocabulary adopted as Universe set;

2. A set of all non-logical constants – Con (e.g., Con_a denotes the set of the constants of the type a);

3. A set of all the variables – Var (e.g., Var_a – the set of the variables of the type a);

4. A set of all the expressions of the type a – ME_a (abbreviation from **M**eaning **E**xpression, as usually). For more details it should be consulted [3].

The semantic component involves a model M interpreted as follows:

$M =< U, F, g >$, where U – a non-null set of elements from the Universe (it may include the entire Universe), F – a function attributing values of the type a to every single constant from the set Con_a, g – a function attributing values of the type a to every single variable from the set Var_a.

For details it may be consulted [4], [5], [6].

Finally, to accurately interpret the linguistic phrases generated with the rigid classical categorial grammar approach in the context of the proposed logical language, a correspondence (it will be defined below) between the syntactic categories and the semantic ones has to be defined.

Definition 2.7:

1. For the basic grammar categories (definition 2.4) a translation function f should be parsed as follows:

2. $f : N \to e$, proper nouns are associated with the elements of the vocabulary V;

3. $f : S \to t$, sentences are associated with the element t (true, false) from the Universe set;

4. $f : CN \to < e, t >$, common nouns are associated with the first degree predicates;

5. For the other categories of the set $Cat(B)$, the following relation should be defined:

   $f(\backslash(A, B)) =< (f(A), f(B) >$ and $f(/(A, B)) =< f(A), f(B) >$, where A and $B \in Cat(B)$.

Example: Let "John expertly hoists the flag" be a phrase to be interpreted. Using the translation f, described above, it can be easily derived that:

$< \text{John}, N > \rightarrow e \rightarrow John^{"}$

$< \text{flag}, CN > \rightarrow < e, t > \rightarrow \lambda x[flag^{"}(x)]$

$< \text{hoist}, \backslash(N, /(CN, S)) > \rightarrow < e, << e, t >, t >> \rightarrow \exists x[flag^{"}(x) \wedge hoist^{"}(John^{"}, x)]$

$< expertly, \backslash(\backslash(N, /(CN, S)), \backslash(N, /(CN, S))) > \rightarrow$

$\rightarrow << e, << e, t >, t >>, < e, << e, t >, t >> > \rightarrow$

$\rightarrow expertly^{"}(\exists x[flag^{"}(x) \wedge hoist^{"}(John^{"}, x)]).$

Comment: The presented interpretation is an interpretation "de re" of the sentence, that is, in other words, it is an extensional one. Here, this kind of interpretation is just the only possible one.

3 Modeling NL sentence interpretation by transition networks

As we underlined above, the formulae for double-level semantic interpretation of NL phrases have been obtained, exclusively, manually. But it is possible to model this generation process automatically using the so called transition networks. For the first time, the transition networks were presented in [7]. In the context of syntactical categories used for NL entities definition, we may consider them as a means for encoding of the semantic types. We will modify this formalism as follows:

Definition 3.1: The semantic transition network (STN) is an object $STN = \{V, IdN, F, N, i0\}$ containing the following components: V is the input set of the terminal symbols (words), the set IdN refers to the lexical entities, accessed through its elements, the set N denotes the transition states of network, $i_0 \in N$ is the initial state of network and, finally, F is a symbol for interpretation function defined below.

Comment 1: The semantic transition network represents an oriented graph with marked edges and, actually, it includes the following components:

1. A finite set N of the vertices and a set E of the edges: $E \subset N \times N$. For simplicity the vertices are represented by integer positive numbers.

2. A label function F on edges which is given by:

   (a) A finite set V of terminal symbols (words) known as the input.

   (b) IdN – a finite set of symbols referring to the lexical entities of NL.

   (c) $F : H \rightarrow (R \times CatE) \cup ExL \cup Proc$, is a function, where R is the subset of pairs $(V \times IdN')$ and $IdN' = \{Id | Id \in IdN$ and $A(Id, x), x \in V\}$.

Comment 2:

1. We consider the concept of lexical entities as belonging to the syntactical level. Thus, the set IdN is nothing more than a set of basic grammar categories.

2. The predicate A is defined as it is shown below:

$$\lambda Id[A(Id, ^{\wedge} \lambda P \exists x[^{\vee} P(x)])],$$

where the variable Id assumes the type e and the variable P has the type $< i, < e, t >>$. Here we suppose $i \in N$, where N denotes the set of so called indexes (state, world, vertices of transition network). The operators $^{\vee}$ and $^{\wedge}$ can be called, deliberately, as "abstract" operator and "concrete" operator, respectively. For example, the expression $^{\vee} P$ is a predicate (type $< e, t >$). The expression $P(x)$ is an incorrect one, because of mismatching of their types. It is rather clear that x is considered of the type e. Finally, λ-sign was used for λ-calculus evocation. So, this approach ensures rather natural definition for NL sentence interpretation. For more information it could be recommended [9].

1. The elements of the set $CatE$ are categorial expressions of precise location of the terminal element in NL sentence.

2. The ExL is the logic expression situated on the edge of the network. The edge is passed iff the expression on the edge is 'true'.

3. The set $Proc$ is the set of procedures attached to the edges which, in turn, might be appealed if the transition on the edge is possible.

Definition 3.2: Semantically, the inference process can be represented by certain object, usually named semantic translator $T = \{\Sigma, \Delta, STN\}$ that comprises the components:

- Input band, which represents a string of cells, containing the symbols (just one per cell) from Σ (the same as the vocabulary of the STN-object);

- Output band with the same structure as the input band, but symbols are taken from Δ, i.e. taken from the output vocabulary of T;

- Internal memory containing as a program the certain STN-object, defined above;

- Processor device executes the program in the internal memory with its stack memory to assure the inference process;

- One (read or write) device for each (input or output) band assures the information processing on these bands.

It is well known how such device operates. Next we will reinterpret and adapt it for STN execution.

The state of the translator T is defined by the pair (Id, i), where Id is the name of some basic category taken from IdN set and i represents the number (integer) of the transition network's vertex. In short, to translate a NL phrase into the formulae of logical language, represented here through output vocabulary Δ, we must assure the translation of

each syntactic component of this phrase. That is, we have to manage a kind of homomorphism and, consequently, the Compositionality Principle is still valid.

The processor must be able to interrupt the transition between two vertices of the network when the Id name on the edge is not compatible with $CatE$ expression on this edge. To show how the translator T operates, we present certain typical situations of the form:

$$((Id, i), \alpha, \gamma, \beta),$$

where (Id, i) is the current state of the translator, α – the string of the input symbols, β – the string of the output symbols, γ – the content of the stack memory.

So, the concept of state allows for measuring the discrete time of inference process, modeled through semantic translator.

The following typical situations for inference process are possible:

a) $((Id^i, i), a\alpha, CatE_i\gamma, \beta)| - ((Id^j, j), \alpha, \gamma, b\beta)$, where the terminal symbol a is written on the input band and the edge of the transition network is marked by the same terminal symbol a; The symbol b signifies the output symbol and j – the second vertex of the edge.

b) $((Id^i, i), a\alpha, CatE_i\gamma, \beta)| - ((Id^{ki}, i), a\alpha, CatE_{ki}\gamma, \beta)$, where the terminal symbol a is visualized on the input band, while the edge is labeled by Id^i. This situation models the case when Id^i is incompatible with the terminal symbol a.

There are no other situations.

Remark: The situations presented above have been simplified.

Definition 3.3: The result of the semantic translator T is defined as follows:

$$Res = \{(x, z)|((Id^1, 1), x, \phi, \phi| - *T((Id^f, f), -|\phi, z))\},$$

where $(Id^1, 1)$ is the initial state and (Id^f, f) is the acceptance state (sentence, for example), the name of the initial transition network and f is its final state. Asterisk "$*$" represents the closure of the relation $|-$, which is a derivation from one state to another.

Comment 3: Obviously, the string z represents certain formula belonging to the logical language. Consequently, the presented semantic translator T implements the translation from NL phrases into formulae of logical language as it was intended.

Example. As the example we will use the same NL sentence analyzed earlier, namely: "John expertly hoists the flag". This sentence has been already represented at the syntactic level by a classical categorial grammar elsewhere in this study. Subsequently, we will illustrate the inference process for obtaining the corresponding logical language formulae by presenting just only the content of processor's stack memory. The obtained result is presented in Fig. 1.

We have presented another look at meaning of the natural language. Namely, when the meaning of the sentence is identified by the transition networks, it assures the obtaining of the logical expressions in a double-level manner. The expressions verify the truth conditions necessary for understanding the sense of sentence by some agent.

Categorial grammar model of the phrase is composed of:

1. Basic categories : $B = \{N, CN, S\}$.

2. Vocabulary: $V = \{John, flag, expertly, tohoist\}$.

3. Classical grammar G is:

$G = \{<$ John$, N >, <$ flag$, CN >, <$ to hoist$, \backslash(N, /(CN, S)) >,$
$<$ expertly$, /(\backslash(N, /(CN, S)), \backslash(N, /(CN, S))), \backslash(\backslash(N, /(CN, S)),$
$\backslash(N, /(CN, S))) >\}$.

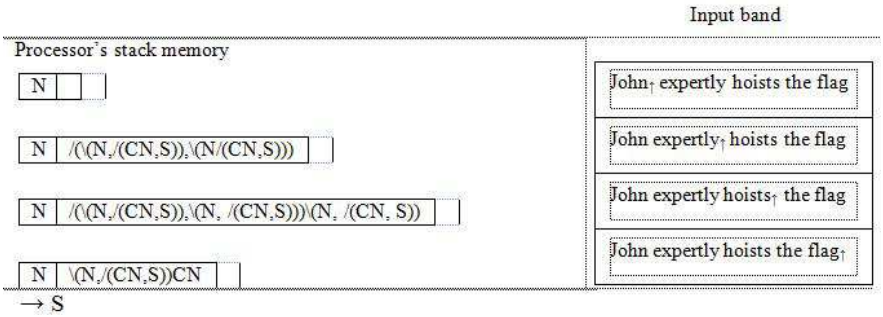

Figure 1. Inference processes for the sentence "John expertly hoists the flag"

4 Conclusions and perspectives:

This study is aimed at developing systems for the interpretation of natural language texts. The proposed approach allows one to elaborate a system that takes into account the complexity of the problem. It is clear that the elaborated techniques do not encompass all the details of the inference process [8], [9], [10]. Our aims were just only to sketch the main principles for solving the problem. What must be done for the future time? The rather incomplete list of the main charges to be solved is presented below:

1. To elaborate for the NL phrases interpretation such a logical language that would "capture" the so called intensional aspects of NL.
2. To elaborate a mechanism that processes the intensional aspects of NL sentences like the transition networks process the extensional aspects.
3. To extend the developped theory including NL coherent phrases (texts).

Many problems remain unsolved in theory. For example, it is important to investigate the structure of possible worlds (for intensional

aspects of NL), the relationship between the types of the analyzed sentence: assertion, order etc. and its interpretation processes.

References

[1] C.Casadio. *Semantic Categories and the Development of Categorial Grammars*, Categorial Grammars and Natural Language Structures, D. Reidel Publishing Company, Dordrecht, 1988, pp. 95–123.

[2] J.Lambek. *The Mathematics of Sentence Structure*, American Mathematical Monthly, 65 (1958), pp.154–170.

[3] S.Creţu, A.Popescu. *Meaning of the Sentence in the Natural Language: semantic insight*, Meridian ingineresc, no. 4, ed. Tehnica UTM, Chişinau, 2013, pp. 46–50.

[4] S.Soames. *Semantics and Semantic Competence*, Philosophical Perspectives, no.3, 1989, pp. 575–596.

[5] R.Montague. *Universal Grammar*, Reprinted in Formal Philosophy; Selected Papers of Richard Montague, 1974, pp. 222–246.

[6] R.Montague. *The Proper Treatment of Quantification on Ordinary English*, Reprinted in Formal Philosophy; Selected Papers of Richard Montague, 1974, pp. 247–270.

[7] W.Woods. *Transition network grammars for natural language analysis*, Communications of the ACM, 13, no. 10, 1970, pp. 591–606.

[8] S.Creţu. *A system for natural language text syntactic – semantic interpretation (SSI)*, The 2nd supplement of the review Informatica Economică, International Conference Knowledge Management: Projects, Systems and Technologies, Bucharest, vol. 1, November, 2006, p. 171–174.

[9] S.Creţu, A.Popescu. *Defining the semantics of natural language sentence*, The 7th International Conference, Microelectronics and Computer Science, september 22-24 2011, UTM, Chişinau, 2011, pp. 174–177. (in Romanian)

[10] S.Creţu, A.Popescu. *Semantic and pragmatic aspects of the meaning sentence*, Meridian ingineresc, no. 3, ed. Tehnica UTM, Chişinau, 2013, pp. 18–23. (in Romanian)

Sergiu Creţu, Anatol Popescu Received March 17, 2015

Sergiu Creţu
E.S. Academy of Moldova
E–mail: *srgcretu@yahoo.com*

Anatol Popescu
Technical University of Moldova,
E–mail: *an.popescul@gmail.com*