A State of the Art Technique in Semantic Analysis of Natural Language Utterances

Marija Brkić, Maja Matetić
Department of Informatics
Faculty of Arts and Sciences
University of Rijeka
Omladinska 14, 51 000 Rijeka, Croatia
Telephone number: 051-345 034
E-mail: mbrkic@ffri.hr, maja.matetic@ri.t-com.hr

Abstract: This paper presents computational approaches to the problem of semantic analysis, which is a crucial term in language technology applications, e.g. dialogue systems and machine translation. Explanations are supported by Croatian language example sentences. The focus is on syntax-driven semantic approach that is based on the principle of compositionality and on the process of augmenting Context-Free Grammar rules. There are also cases where the meaning of a constituent is not based on the meaning of its parts in straightforward compositional sense, and this paper presents a way of dealing with such cases as well.

I. INTRODUCTION

Language and speech resources are of crucial importance for research and development in language technology. It is equally important to develop language tools to help people communicate effectively in foreign countries or with foreigners. Most researches are done on English language. Since Croatian and other Slavic languages are essentially very different from English, we are in the need of developing language-specific tools. Research on languages with similar characteristics should serve as guidance; consider [1], [2], [3], and [10]. A typical spoken language understanding system has a speech recognizer and a speech synthesizer. The speech recognition results are parsed into semantic forms by sentence interpretation component. Sentence interpretation module often needs discourse analysis to track context and resolve ambiguities. Dialog manager is the central component and it communicates with discourse analysis module, sentence interpretation component and lastly, message generation module [6]. We will focus on the sentence interpretation module. The scope of the semantic analysis is wide which can be seen through [5], [8], [9], [11], and [12]. Our attention will be directed to one of the approaches to semantic analysis, to syntax-driven semantic analysis approach.

II. SYNTAX-DRIVEN SEMANTIC ANALYSIS

Semantic analysis is the process of relating syntactic structures (phrases, clauses, sentences, text) to their language-independent meanings. Idioms, being cultural elements, also have to be converted into relatively invariant meanings. They are special in that they consist of groups of words in a fixed order that have a particular meaning different from the meanings of each word understood on its own.

One of the approaches to semantic analysis is syntax-driven approach. It is based on the principle of compositionality. The key idea of the principle of compositionality is that the meaning of a sentence can be composed from the meanings of its parts. However, this idea should not be literally interpreted. The meaning of a sentence is not just based on the meaning of the words that make it up, but also on the grouping, ordering and relations among the words in the sentence.

Mathematical system used for modeling constituent structure in natural languages which is the most commonly used is the Context-Free Grammar, or CFG. It consists of a set of rules or productions. These rules express the ways that symbols of the language can be grouped and ordered together.

Context-free grammar rules need to be augmented with semantic attachments which instruct how to construct a meaning representation of a construction from the meanings of its constituent parts. [7] These augmented rules have the following structure:

$$A \rightarrow \alpha_1 \ldots \alpha_n \qquad \{f(\alpha_j.\text{sem}, \ldots, \alpha_k.\text{sem})\} \qquad (1)$$

The meaning representation assigned to the construction A can be computed by running the function f on some subset of the semantic attachments of A's constituents.

Meaning representations will be presented in First Order Predicate Calculus, a flexible and well-understood meaning representation language. Let us shortly describe FOPC. It provides three ways to represent an object - constants, functions, and variables. Constants refer to specific objects in the world (e.g. single capitalized letter or concrete words). FOPC functions refer to concepts. They are syntactically the same as single argument predicates but they are actually terms in that they refer to specific objects without having to associate a named object. The last mechanism that refers to objects is a variable which can be used to refer to anonymous object or generically to all objects in a collection. They give us the ability to make inferences or make assertions. These uses are made possible by the use of quantifiers, existential quantifier $\exists$ (there exists) and the universal quantifier $\forall$ (for all). Let us turn now to the mechanisms used to state relations that hold among objects. Predicates are symbols that refer to the relations that hold among some fixed number of objects in a given domain. FOPC sentences can be assigned a value of *True* or False based on whether the prepositions are in accord with the world or not. Sentence with universally quantified variables must be true under all possible substitution, while those with existentially quantified

variables must have at least one substitution that results in a true sentence. The various logical connectives give us the ability to create larger representations. In that respect, we can use three operators represented in Table I.

Consider the sentence:

Ana poslužuje jelo. (*Ana serves meal.*) (2)

The concrete entities are represented by the FOPC constants *Ana* and *jelo*. The lexical rules that introduce these words into the sentence are:

ProperNoun → *Ana* {Ana}

CommonNoun → *jelo* {Jelo}

The *NPs* (Noun Phrases) obtain their meaning representations from the meanings of their children. The semantic expression associated with the child is simply copied to the parent for non-branching grammar rules which we have in our example (2).

NP → *ProperNoun* {*ProperNoun.sem*}

NP → *CommonNoun* {*CommonNoun.sem*}

A generic event *Posluživanje* (Serving) involves *Poslužitelj* (Server) and something *Posluženo* (Served):

$$\exists e, x, y\ Isa(e, Posluživanje) \wedge Poslužitelj(e, x) \wedge Posluženo(e, y) \quad (3)$$

The formula in (4) presents the semantic attachment of the verb *poslužuje:*

$$Verb \rightarrow poslužuje$$
$$\{\exists e, x, y\ Isa(e, Posluživanje) \wedge Poslužitelj(e, x) \wedge Posluženo(e, y)\} \quad (4)$$

The meaning of the *NP* needs to be incorporated into the meaning of the verb and the resulting representation needs to be assigned to the *VP.sem* (Verb Phrase semantic attachment). The variable *y* will be replaced with the logical term *Jelo* as the second argument of the *Posluženo* role of the *Poslužuje* event:

$$\exists e, x\ Isa(e, Posluživanje) \wedge Poslužitelj(e, x) \wedge Posluženo(e, Jelo) \quad (5)$$

The *VP* semantic attachment must have two capabilities. It has to know which variables within the *Verb's* semantic attachment are to be replaced by the semantics of the *Verb's* arguments, and it has to have the ability to perform such a replacement.

TABLE I
TRUTH TABLE GIVING THE SEMANTICS OF THE VARIOUS LOGICAL CONNECTIVES

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$
False	*False*	*True*	*False*	*False*	*True*
False	*True*	*True*	*False*	*True*	*True*
True	*False*	*False*	*False*	*True*	*False*
True	*True*	*False*	*True*	*True*	*True*

This functionality is provided by a notational extension to FOPC called the lambda notation. The extended FOPC syntax includes expressions of the following form:

$$\lambda x P(x) \quad (6)$$

These *lambda*-expressions undergo a process of lambda reduction which means that they can be applied to logical terms to yield new FOPC expressions where the formal parameter variables are bound to the specific terms.

$$\lambda x P(x)(A) \quad (7)$$

$$P(A) \quad (8)$$

One *lambda*-expression can be used as the body of another. After the first reduction the resulting expression is still a lambda-expression. This technique is called *currying* and it actually converts a predicate with multiple arguments into a sequence of single argument predicates. It is worth noting that arguments are limited to FOPC terms.

$$\lambda x \lambda y\ Near\ (x, y) \quad (9)$$

Now we can go back to our example in (2).

$$Verb \rightarrow poslužuje$$
$$\{\lambda x \exists e, y\ Isa(e, Posluživanje) \wedge Poslužitelj(e, y) \wedge Posluženo(e, x)\} \quad (10)$$

The attachment for our *VP* rule specifies a lambda-application. *Lambda*-expression is provided by *Verb.sem* and the argument is provided by *NP.sem., Jelo.*

$$VP \rightarrow Verb\ NP \qquad \{Verb.sem(NP.sem)\} \quad (11)$$

This *lambda*-application results in the binding of the single formal parameter *x* of the *lambda*-expression with the value in *NP*.sem.

$$\exists e, y\ Isa(e, Posluživanje) \wedge Poslužitelj(e, y) \wedge Posluženo(e, Jelo) \quad (12)$$

What we still need is the semantic attachment for the *S* (Sentence) rule. It must incorporate an *NP* argument into the appropriate role in the event representation in the *VP*.sem.

$$S \rightarrow NP\ VP \qquad \{VP.sem(NP.sem)\} \quad (13)$$

However, the *lambda*-application performed at the *VP* rule resulted in a generic FOPC expression. The *Verb* attachment has to consist of an embedded *lambda*-expression to make the *Poslužitelj* role available for binding at the *S* level of the grammar.

$$Verb \rightarrow poslužuje$$
$$\{\lambda x \lambda y \exists e\ Isa(e, Posluživanje) \wedge Poslužitelj(e, y) \wedge Posluženo(e, x)\} \quad (14)$$

The *Verb* attachment consists of a *lambda*-expression inside a *lambda*-expression. The outer expression provides the variable that is replaced by the first *lambda*-reduction, while the inner provides the variable that is replaced by the second *lambda*-reduction. The ordering of variables in the multiple layers *lambda*-expressions in semantic attachment of the verb encodes facts about the expected location of a *Verb's* arguments in the syntax. This is a fairly simple example when considered in English language which has relatively fixed word order, but not in Croatian language. The parse tree for this example is shown in Figure 1.

Let us now look at a phrase 'Lijepa djevojka' (beautiful girl). An obvious and often incorrect proposal for the semantic attachment of the NP is illustrated in the following rules:

$$\text{Nominal} \rightarrow \text{Adj Nominal}$$
$$\{\lambda x\ \text{Nominal.sem}(x)\ \Lambda\ \text{Isa}(x, \text{Adj.sem})\} \quad (15)$$
$$\text{Adj} \rightarrow \text{lijepa } \{\text{Lijepa}\} \quad (16)$$

This yields the following fairly reasonable representation:

$$\lambda\, x\text{Isa}(x, \text{Djevojka})\ \Lambda\ \text{Isa}(x, \text{Lijepa}) \quad (17)$$

This is an example of intersective semantics since the meaning of the phrase can be thought of as the intersection of the category stipulated by the nominal and the category stipulated by the adjective. This amounts to the intersection of the category of beautiful things with the category of girls.

Consider example 'bivši prijatelj' (former friend).

$$\lambda\, x\text{Isa}(x, \text{Prijatelj})\ \Lambda\ \text{Isa}(x, \text{Bivši}) \quad (18)$$

It asserts that the person in question is a friend, which is not true, and it makes use of a fairly unreasonable category of former things. The best approach is to note the status of a specific kind of modification relation and replace this vague relation with some further procedure.

$$\text{Nominal} \rightarrow \text{Adj Nominal}$$
$$\{\lambda x\ \text{Nominal.sem}(x)\ \Lambda\ \text{AM}(x, \text{Adj.sem})\} \quad (19)$$

Applying this rule to 'Lijepa djevojka' results in the following formula in which AM stands for adjective modifier:

$$\exists\, x\ Isa(x, Djevojka) \wedge AM(x, Lijepa) \quad (20)$$

This proposal does not work with former friend where the solution has to be based on the specific semantics of the adjectives and nouns in question.

In general, lexical rules provide content level predicates and terms for meaning representations. The semantic attachments to grammar rules just put predicates and terms together in the right ways.

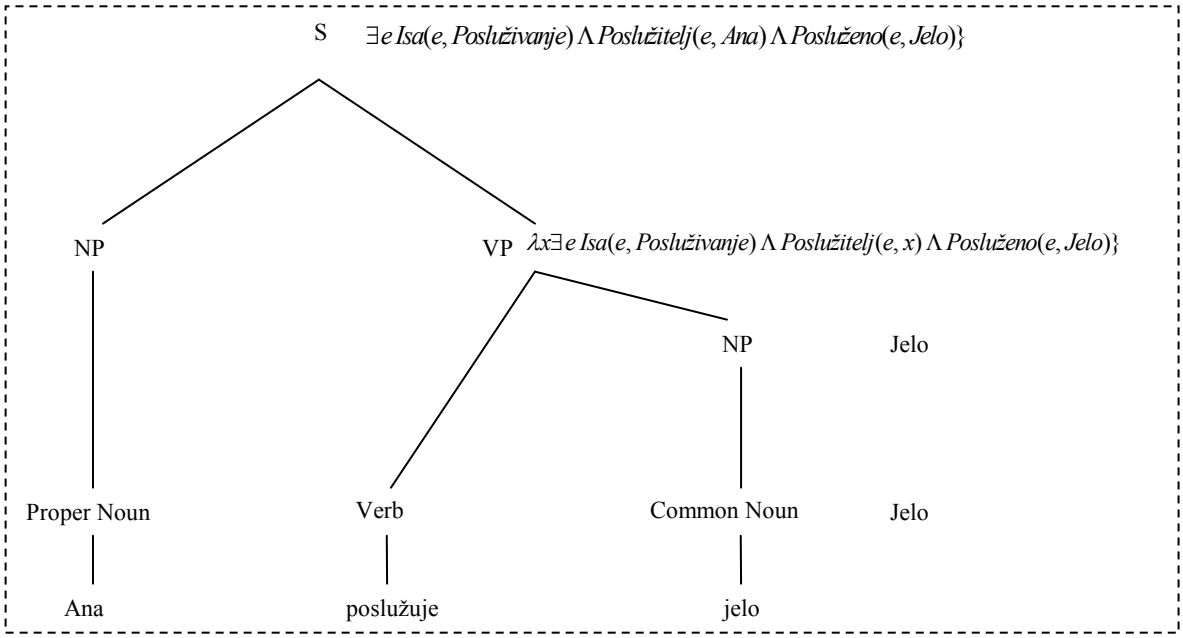

Figure 1. Parse tree with semantic attachments for *Ana poslužuje jelo*.

Up to now, we focused only on declarative sentence. Let us consider the following examples: *Ana poslužuje ručak.* (*Ana serves lunch.*) *Posluži ručak!* (*Serve lunch!*) *Da li Ana poslužuje ručak?* (*Does Ana serve lunch?*) *Tko poslužuje ručak?* (*Who serves lunch?*) These sentences all contain propositions concerning the serving of lunch on flights but they differ with respect to the role they are intended to serve. The first of these sentences conveys factual information to the listener, the second is a request for an action, and the last two are requests for information. To differentiate between these sentences, we can introduce a set of operators which will be applied to the FOPC sentences: *DCL* (declaratives), *IMP* (imperatives), *YNQ* (yes-no questions), and *WHQ* (wh-questions).

By altering the basic sentence rule we have been using so far, we will get a rule for declarative sentences:

$$S \rightarrow NP\ VP\ \{DCL(VP.sem(NP.sem))\} \quad (21)$$

Imperative sentences begin with a verb phrase and do not have an overt subject.

$$S \rightarrow VP\ \{IMP(VP.sem(DummyYou))\} \quad (22)$$

Yes-no-questions consist of a sentence initial auxiliary verb, followed by a particle, a subject noun phrase and then a verb phrase.

$$S \rightarrow Aux\ NP\ VP\ \{YNQ(VP.sem(NP.sem))\} \quad (23)$$

Yes-no-questions can be answered pretty simply, by just determining whether the preposition is in the knowledge base, or whether it can be inferred from the knowledge base.

Wh-subject-questions are the last type we are going to deal with. They have the following attachment [7]:

$$S \rightarrow WhWord\ NP\ VP\ \{WHQ(NP.sem.var, VP.sem(NP.sem))\} \quad (24)$$

II. IDIOMS

As one could expect, the principle of compositionality does not work with idioms. Idiom is an expression whose meaning cannot be deducted from the meaning, grouping and ordering of its parts. It refers to a figurative meaning that is known only through conventional use [7]. Consider the following idioms: *Kako prostreš, tako ćeš i leći (As you make your bed, so you must lie upon it)*, *Mi o vuku, a vuk na vrata (Speak of the devil, and in he walks), Kuj željezo dok je vruće (Make hay while the sun shines)*, *Tiha voda brege dere (Still waters run deep), To je kap koja je prelila čašu (That was the last straw), Trčati pred rudo (Jump the gun),* etc.

The best way to deal with these expressions is to introduce new grammar rules designed for them. These rules mix lexical items with grammatical constituents. They also introduce semantic content that is not derived from any of its parts.

Consider the following rule:

$$S \rightarrow trčati\ pred\ rudo \quad \{Prebrzo\} \quad (25)$$

The constant *Prebrzo* should not be taken for granted as the meaning representation for this idiom. However, it illustrates that the meaning of the idiom has nothing to do with the meanings of its parts. Of course, special attention in these rules for Croatian and similar languages should be paid to person marking. [4] (*Trčim pred rudo. Trčiš pred rudo.*)

There are some idioms that allow for some variation and those should be represented by more general rules. Let us take the following idiom as an example: *Tresla se brda, rodio se miš* (*Much ado about nothing*). We could define the following rule for this idiom:

$$S \rightarrow tresla\ se\ brda,\ rodio\ se\ miš \quad \{Pretjerivati\} \quad (26)$$

However, somebody could say *Tresla se brda, rodio se mali miš* and this rule would not work any more. That is the reason why we should make more general rules [7]:

$$S \rightarrow tresla\ se\ brda,\ rodio\ se\ mišNP \quad \{Pretjerivati\} \quad (27)$$

III. SEMANTIC GRAMMARS

Grammars that are needed for compositional semantic analysis and that represent one of the ways of instantiating a syntax driven approach in practical systems are known as semantic grammars and they differ a lot from traditional grammars. The need of having uniform semantic attachments often results in constituents that are at the right level of generality for the syntax, but at too high level for semantic purposes. Let us consider the rule for the phrase *kineski restoran* (*Chinese restaurant*):

$$Nominal \rightarrow Adj\ Nominal \quad \{\lambda x\ Nominal.sem(x) \wedge AM(x, Adj.sem)\} \quad (28)$$

It results in the following meaning representation:

$$\exists x\ Isa(x, Restoran) \wedge AM(x, Kineski) \quad (29)$$

This is just an indication that the nominal is modified by the adjective. The expression we want to represent means that food is prepared in a particular way. These problems can be solved by means of semantic grammars. Rules in these grammars are made no more general than is needed

for sensible semantic analysis. The rule that could be used to parse the phrase 'kineski restoran' could be:

$$RestaurantType \rightarrow Nationality\ RestaurantType \quad (30)$$

Although convenient at first sight, semantic grammars need to be huge in size in order to be efficient. Consider the expression 'kanadski restoran' (*Canadian restaurant*). It matches the rule, although such interpretation would be utterly incorrect. Such phrase simply refers to the restaurant located in Canada.

Semantic grammars are useful when dealing with anaphors and ellipsis because they enable prediction. Unfortunately, cannot be reused because they are domain-specific [7].

IV. CONCLUSION

Since Croatian language syntax essentially differs from English language for which a great deal of research has been done, language-specific tools need to be developed. This paper outlines one of the approaches to semantic analysis, syntax-driven approach. It is based on the principle of compositionality, but taking care of grouping, ordering and relations among words in the sentence. Context-free grammar rules augmented with semantic attachments specify how to construct a meaning representation of a construction from the meanings of its constituent parts. Meaning representations are presented in First Order Predicate Calculus. Outlined rules can be further developed. Completely new grammar rules are introduced for idioms since they refer to figurative meaning. These rules mix lexical items with grammatical constituents as shown through examples. Grammars needed for compositional semantic analysis differ a lot from traditional grammars and they are known as semantic grammars.

REFERENCES

[1] Cussons, J., Džeroski, S., Erjavec, Ž., „Morphosyntactic Tagging of Slovene using Progol“, Lecture Notes in Computer Science, *Proceedings of the 9th International Workshop on Inductive Logic Programming*, vol. 1634, 1999., str. 68-79, www-ai.ijs.si/SasoDzeroski/files/1999_CDE_MorphosyntacticTaggingSlovene.pdf

[2] Drodzynski, W., Homola, P., Piskorski J., Zinkevicius, V., „Adapting SProUT to processing Baltic and Slavonic languages“, *IESL'03 Workshop held in conjunction with the RANLP Recent Advances in Natural Language Processing*, Borovets, Bugarska, 2003.

[3] Erjavec, T., Džeroski, S., „Machine Learning of Morphosyntactic Structure: Lemmatizing Unknown Slovene Words“, *Applied Artificial Intelligence*, 18: 17-41, 2004.

[4] Franks, S., „Slavic Languages“, in: Cinque, G., Kayne, R. (ed.), *Handbook of Comparative Syntax*, 2005.

[5] Gildea, D. and Jurafsky, D., *Automatic Labeling of Semantic Roles*, Computational Linguistics 28(3) (2002) 245--288 14

[6] Huang, X., Acero, A., Hon, H., *Spoken Language Processing. A Guide to Theory, Algorithm, and System Development*, Prentice Hall, Upper Saddle River, New Jersey, 2001.

[7] Jurafsky, D., Martin, J. H., *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[8] Lin, D. and Pantel, P., *Discovery of Inference Rules for Question Answering,* Natural Language Engineering 2001., 7(4):343-36

[9] Lin, D. and Pantel, P., 2001 *Induction of Semantic Classes from Natural Language Text*, In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2001., pp. 317-322

[10] Nenadić, G., Vitas, D., Krstev, C., „Local Grammars and Compound Verb Lemmatization in Serbo-Croatian“, Zybatow, G., Junghanns, U., Mehlhorn, G., Szucsich, L. (ed.), *Current Issues in Formal Slavic Linguistics*, Peter Lang, Frankfurt/Main, 2001., str. 469-477

[11] Riloff, E. and Thelen, M., *A Rule-based Question Answering System for Reading Comprehension Tests"*, Workshop on Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems, ANLP/NAACL, 2000.

[12] Tsang, V., Stevenson, S. and Merlo, P., *Crosslinguistic Transfer in Automatic Verb Classification*, Proceedings of the 19th International Conference on Computational Linguistics (COLING), 2002., 1023-1029.