Lexical Functions as a Tool of ETAP-3[1]

Jurij D. Apresjan, Igor M. Boguslavsky, Leonid L. Iomdin, Leonid L. Tsinman

Laboratory of Computational Linguistics
Institute for Information Transmission Problems, Russian Academy of Sciences,
Moscow
{apr,bogus,iomdin,cin}@cl.iitp.ru

Résumé

Cet article décrit l'utilisation du concept de fonctions lexicales, tel que proposé par la "théorie sens-texte" (TST) d'Igor Mel'čuk, dans des applications avancées de TAL telles que les analyseurs, la traduction automatique (TA) de bonne qualité, les systèmes de paraphrase et l'apprentissage du lexique assisté par ordinateur, applications intégrées dans le processeur linguistique ETAP-3. Dans l'analyse, les fonctions lexicales de collocation sont utilisées pour résoudre ou réduire l'ambiguïté syntaxique et lexicale. Le système de TA a recours aux fonctions lexicales pour fournir des équivalents idiomatiques dans la langue cible pour des phrases source dans lesquelles l'argument et la valeur de la même fonction lexicale sont tous les deux présents. Le système de paraphrase, qui produit automatiquement une ou plusieurs paraphrases pour une phrase ou un syntagme donné, peut être utilisé dans bon nombre d'applications avancées de TAL, allant de la TA jusqu'à l'aide à la rédaction et la planification de texte. Le système d'apprentissage du lexique assisté par ordinateur est aussi fondé sur le concept de fonctions lexicales comme outil de description formelle de la partie du vocabulaire qui est à la fois systématique et idiomatique et qui est par conséquent des plus difficiles pour l'acquisition du langage.

Mots-clefs

Fonctions lexicales, traitement automatique des langues naturelles, traduction automatique, analyse, paraphrase.

---

[1] This work has been supported in part with grants Nos. 02-06-80106 and 01-06-80453 of the Russian Foundation for Fundamental Research, grant No. NSH-1576.2003.6 of the President of the Russian Federation for support of the leading scientific schools and a grant from the Program of Fundamental Research of the Departments of the Russian Academy of Sciences.

Abstract

The paper describes the use of lexical functions, an instrument proposed in Igor Melčuk's "Meaning ⇔ Text Theory" (MTT), in advanced NLP applications as exemplified in the ETAP-3 linguistic processor, including parsers, high quality machine translation (MT), a system of paraphrasing and computer-aided learning of lexica. In parsing, collocate LFs are used to resolve or reduce syntactic and lexical ambiguity. The MT system resorts to LFs to provide idiomatic target language equivalents for source sentences in which both the argument and the value of the same LF are present. The system of paraphrasing, which automatically produces one or several synonymous transforms for a given sentence or phrase, can be used in a number of advanced NLP applications ranging from MT to authoring and text planning. The computer-aided system of learning lexica is also based on the concept of LFs as a tool of formal description of that part of vocabulary which is simultaneously systematic and idiomatic and is therefore most difficult for language acquisition.

Keywords

Lexical Functions, Natural Language Processing, Machine Translation, Parsing, Paraphrasing

1 ETAP-3 and the Meaning ⇔ Text Theory

ETAP-3 is a multi-language and multifunctional linguistic processor developed by the Computational Linguistics Laboratory of the Institute for Information Transmission Problems of RAS[2]. It is based on I. Mel'čuk's Meaning ⇔ Text theory[3] (MTT) and is targeted at various NLP applications.

ETAP-3 shares with the classical version of MTT the following basic principles:

a) The concept of language as a many-to-many correspondence between meanings and texts. Technically, language may be considered to be an encoding and decoding device which ensures the transition from the meaning the speaker wishes to convey to a set of (synonymous) texts which express it (synthesis), and from the text the addressee perceives to its meaning or a set of meanings in cases of homonymy (analysis). In other words, ETAP-3, like MTT, is not a generation but a translation model.

b) The stratificational approach to language description. The transition from meanings to texts and vice versa is effected through a number of intermediate stages called the levels of utterance representation. ETAP-3 is divided into the same principal levels of utterance representation as MTT – morphological, syntactic, and semantic.

---

[2] For the general structure and practical applications of this linguistic processor see Apresjan *et al*. 1992, Apresjan & Tsinman 2002, Iomdin & Tsinman 1997, Boguslavsky *et al*. 2000a, 2000b, 2002, Boguslavsky 2001a, 2001.

[3] See, for example, Mel'čuk 1974, 1996, 1997, Mel'čuk & Zholkovskij 1984, Mel'čuk *et al*. 1984, 1988, 1992, 1999.

c) Priority of dictionaries. Dictionaries are considered to be the pivotal point of linguistic modeling because the transition from meanings to texts or vice versa cannot be effected without great amounts of linguistic information on lexical items stored immediately in their dictionary entries.

d) Insistence on dependency trees as opposed to immediate constituents as the proper way of representing syntactic structures.

Apart from these basic theoretical tenets we have borrowed from MTT most of the mechanisms and devices of sentence[4] representation at various levels.

However, ETAP-3 was designed as a system intended for diverse NLP applications, and much of its specificity can be accounted for by this bias. Below we give a list of those particularities of ETAP-3 that make it distinct from its MTT prototype – in fact, make it a separate dialect within the MTT territory.

a) The classical version of MTT is a linguistic model of synthesis. It simulates the activity of the speaker, that is, the transition from some semantic representation to a set of synonymous texts which express it. ETAP-3 is above all a model of analysis; it ensures the transition from the given text to its deeper representation, the depth of representation determined by the task at hand. In a processor intended for applications this was the only possible choice. Our first system was that of machine translation, so our starting point could be nothing other than a NL text in the source language.

An even stronger claim could be made: texts are the only reality through which meanings become accessible to the researcher. As a matter of fact, written or oral texts are the only physical objects liable to direct linguistic observation and hence the only reality the linguist is faced with.

This emphasis on analysis does not mean that models of synthesis were totally ignored in our work. For a number of applications, machine translation included, we had to develop such models as well: after ETAP-3 produces a sufficiently deep representation of the processed sentence, the latter has to go through all the stages of synthesis to reach the level of a NL sentence in the target language.

b) Dependency trees of the MTT are principally unordered, while dependency trees of ETAP-3 are linearly ordered.

c) The rules of MTT belonging to the same level are also unordered. In ETAP-3 blocks of rules or parts of the same rule are ordered not only by the succession of levels (morphological $\Rightarrow$ syntactic $\Rightarrow$ semantic) but sometimes also within the same level. For example, syntactic rules may specify two types of conditions to be checked before the operation envisaged in the rule can be fully performed: (i) linear conditions (e. g., the presence or absence of certain word forms or lexemes in the sentence, the presence or absence of certain punctuation marks or word order etc); (ii) tree conditions (e. g., the presence or absence of a certain node subordinated to one of the word forms mentioned in the rule). The parser proceeds in the

[4] In computer implementations of MTT we are dealing with sentences rather than utterances.

following order: first, the linear conditions are checked, and a set of hypotheses compatible with them is generated; then it goes on to checking tree conditions and purges the set of previously generated hypotheses of the ones that are incompatible with the tree conditions.

Linguistically, ETAP-3 consists of various sets of rules and dictionaries.

As has been noted above, the rules mirror the levels of sentence representation of the MTT. Hence the main sets of rules are those of: (a) morphological analysis (transition from an NL sentence to its morphological structure); (b) surface-syntactic analysis, or parsing (transition from the morphological structure of the sentence to its surface-syntactic structure in the form of a dependency tree); (c) quasi-deep syntactic analysis resulting, among other things, in the LF interpretation of the nodes of the surface-syntactic structure. For certain options, e.g. translation from and into artificial languages in systems of NL interface with data bases and the like, one more set of rules has been developed – that of semantic analysis.

All the rules occurring in various blocks of ETAP-3 are subdivided into three main types: (i) general rules that apply to all the sentences in the course of their processing; (ii) class-specific rules that hold for compact groups of words and are referred to by their names in the dictionary entries of the respective items; (iii) word-specific rules that are characteristic of individual lexical items and are stored directly in their dictionary entries. The second and third types of rules are activated only on condition that the processed sentence contains the relevant lexical items (the principle of self-tuning of the system).

The input to ETAP-3 in any of its options is an NL text, which is processed sentence by sentence. The main formal object of ETAP-3 yielded by the parser and crucial for all its further options is the surface-syntactic structure (or structures) of the processed sentence. As has been noted above, a surface-syntactic structure is a linearly ordered dependency tree. Its nodes represent the word-forms of the sentence, pairs of nodes being linked by one of several dozen (up to 80 for Russian, over 60 for English) syntactic dependency relations.

In all further options ETAP-3 makes use of various systems of tree transformation rules which effect all the necessary changes in the surface-syntactic structure to reach a deeper level of sentence representation. ETAP-3 has a highly sophisticated modular structure ensuring the selection of just those modules which are necessary for the given option.

The main options of ETAP-3 are machine translation, multilingual communication via an interlingua (namely, the UNL), and paraphrasing. The main NLP system developed within ETAP is a bi-directional English-to-Russian and Russian-to-English machine translation system; we have also staged smaller-scale experiments with French, German, Spanish, and Korean.

The main dictionaries are combinatory dictionaries of English and Russian, counting over 65,000 lexical entries each. Every entry in such a dictionary stores the following types of information on a lexical item: part of speech, standard translation into the other working language, syntactic features, semantic features (primarily used as selectional restrictions), pattern of government, lexical functions (LFs) and all sorts of rules specific for the given lexical item, non-trivial translation rules included. To show the amount of information on LFs, which are our principal concern in the present paper, we shall quote the respective fragment from the entry **CONTROL 1,** a noun.

V0: CONTROL 2

MAGN: STRICT / RIGID

ANTIMAGN: LAX

OPER1: HAVE

INCEPOPER1: ESTABLISH / SET UP

FINOPER1: LOSE

LIQUOPER1: DEPRIVE (somebody of control)

OPER2: BE (under control)

INCEPOPER2: FALL (under control)

FINOPER2: GO OUT (of control)

LIQUOPER2: FREE (somebody of control)

LABOR1-2: KEEP (somebody under control)

INCEPLABOR1-2: TAKE (somebody under control)

LIQUFUNC2: CANCEL (control over something)

After this cursory account of ETAP-3 we can turn to the general notion of a LF. We presume sufficient familiarity of the reader with the apparatus of LFs and therefore shall limit ourselves to a brief list of those types and properties of LFs that make them interesting for linguistic applications. We shall also sketch the algorithm of their identification in an arbitrary sentence.

2 Types of LFs, Algorithm of their Identification in the Text and their Properties

I. Mel'čuk singles out two types of LFs – paradigmatic (substitutes) and syntagmatic (collocates, or, in Mel'čuk's terms, parameters).

Substitute LFs are those which replace the keyword in the given utterance without substantially changing its meaning or changing it in a strictly predictable way. Examples are synonyms, antonyms, converse terms, various types of syntactic derivatives and the like. They play an important role in paraphrasing sentences. In the course of parsing each such function becomes accessible to the linguistic processor as part of the dictionary entries of the lexical items occurring in the sentence.

Collocate LFs are those which combine with the keyword in the given utterance and syntactically either subordinate it or are subordinated to it. Typical examples of collocate LFs are support verbs of the OPER / FUNC family and such adjectival LFs as MAGN. They play an important role not only in paraphrasing but in a number of other applications as well.

Collocate LFs occurring in the processed sentence are identified by means of special rules which use three types of information: (a) information from the LF zone of the combinatory dictionaries, see above; (b) the definition of the respective LF, in particular, the specification

of the syntactic relation by which it is connected with its keyword, or argument; (c) the syntactic hypotheses for the processed sentence generated by the parser. Suppose the parser is processing the following sentence: *The government has rigid control of the exchange rates of foreign currencies*. The LF zone of the entry for *control* includes the verb *have* as the value of the LF OPER1 and the adjective *rigid* as the value of the LF MAGN. The definitions of the respective LFs state that OPER1 in the active voice subordinates its keyword by the first completive relation and that MAGN is subordinated to its keyword by the modificative relation. Precisely these syntactic hypotheses are generated by the parser for the processed sentence. As a result *has* is identified as OPER1 of *control* and *rigid* – as MAGN of *control*.

Now we shall briefly list the properties of LFs which are crucial for applications.

1. Universality: LFs are universal in the sense that several dozen LFs describe the basic semantic relations between lexical items in the vocabulary of any natural language and the basic semantic relations which syntactically connected word forms can obtain in the text.

2. Double idiomaticity. First of all, LFs are idiomatic intralinguistically. A handy example is the LF MAGN = 'a great degree of what is denoted by the key lexeme'. We say, in good English, *to sleep soundly* and *to know firmly,* not the other way round: *??to sleep firmly* or **to know soundly* would be non-idiomatic or downright bad English. Secondly, the LFs are idiomatic cross-linguistically. For example, in Russian the combination *krepko spat'* (literally *to sleep firmly*) is quite acceptable although it is rejected in English.

3. Paraphrasability. The LFs of the support-verb family (OPER-FUNC) and some other families can form combinations with their arguments which are synonymous to the basic verb. E.g., *The government controls prices – The government has control of prices – The government keeps prices under control – The prices are under the government's control.* Most paradigmatic LFs (synonyms, antonyms, converse terms, various types of syntactic derivatives and the like) can also substitute for the keyword to form synonymous sentences.

4. Semantic diversity. Sometimes the values of the same LF from the same argument lexeme are not synonymous. This is especially characteristic of the LF MAGN. We can describe a great degree of *knowledge* (in the sense of erudition) in the following three ways: a) as *deep* or *profound*; b) as *firm*; c) as *broad* or *extensive*. Although all these adjectives are valid values of the LF MAGN, the three groups should somehow be distinguished from each other because the respective adjectives have very different scopes in the semantic representation of the keyword. *Deep* and *profound* characterize *knowledge* with regard to the depth of understanding; *firm* specifies the degree of its assimilation; *broad* and *extensive* refer to the amount of acquired knowledge. To keep such distinctions between different values of the same LFs in the computerized algebra of LFs it is sufficient to ascribe to the standard name of an LF the symbol NS (non-standardness) plus a numerical index and maintain the correspondences between the two working languages by ascribing the same names to the respective LFs in the other language.

As has already been stated, the above-mentioned properties of LFs make possible their practical use in a variety of NLP applications. Here an aside is in order. We are not the only or the first people who thought of applying LFs in NLP. There have been other people before us, see, for example, Arsentyeva *et al*. 1969, Streiter 1996, Polguère 1998, Mel'čuk, Wanner

2001, to mention but a few. Our claim to originality is that what is reported below has been actually run on the computer and tested against really representative material.

Below we shall discuss the following four uses of LFs in computer implemented applications: (a) syntactic and lexical ambiguity resolution in parsers; (b) idiomatic translation of a large class of set expressions in MT; (c) sentence paraphrasing; (d) computer-aided learning of lexica.

3 LFs as a Disambiguation Tool in Parsing

The basic LF tool of disambiguation are collocate LFs. They are used to resolve syntactic and lexical ambiguity.

Among the collocate LFs the LFs of the support verb family are of greater importance because they serve to resolve both types of ambiguity – syntactic and lexical. Non-verbal types of collocate LFs (MAGN, ANTIMAGN, BON, ANTIBON, MULT, CAP, EQUIP, FIGUR etc) are typically used to resolve only lexical ambiguity.

3.1 Syntactic Ambiguity Resolution

Consider the well-known *amor patris* type of syntactic ambiguity under which the dependent word in the genitive case may represent either the first or the second actant of the keyword, that is, either the semantic subject (agent) or the semantic object. This type of syntactic ambiguity is quite frequent in English, Russian and many other European languages. Consider such English phrases as *support of the parliament* or *support of the president.* Two most important actants of *support* are subject (agent) and object. Both these actants may be represented by a nominal group preceded with the preposition *of*. Therefore the phrases under consideration may mean both, 'support given by the parliament <by the president>' (subject interpretation with the agentive syntactic relation between *support* and the subordinated noun), and 'support given to the parliament <to the president>' (object interpretation with the first completive syntactic relation between *support* and the subordinated noun).

This type of ambiguity is often extremely difficult to resolve, even within a broad context.

LF support verbs can be successfully used to disambiguate such phrases because they impose strong limitations on the syntactic behaviour of their keywords in texts.

Consider the phrase *The president spoke in support of the parliament,* where the verb *to speak in* is a non-standard OPER1 from the noun *support*. Now, verbs of the OPER1 type may form collocations with their keyword only on condition that the keyword does not subordinate directly its first actant. The limitation is quite natural: OPER1 is by definition a verb whose grammatical subject represents the first actant of the keyword. Since the first actant is already represented in the sentence in the form of the grammatical subject of OPER1, there is no need to express it once again. This is as much as to say that the phrase *The president spoke in support of the parliament* can only be interpreted as describing the support given to the parliament, with *parliament* fulfilling the syntactic function of the complement of the noun *support*.

Conversely, verbs of the OPER2 type may form such collocations only on condition that the keyword does not subordinate directly its second actant. Again, the limitation is quite natural: OPER2 is by definition a verb whose grammatical subject represents the second actant of the keyword. Since the second actant is already represented in the sentence in the form of the grammatical subject of OPER2, there is no need to express it once again. This is as much as to say that a phrase like *The president enjoyed* [OPER2] *the support of the parliament* implies the support given to the president by the parliament, with *parliament* fulfilling the syntactic function of the agentive dependent of the noun *support*.

In much the same way the parser resolves syntactic ambiguities in the context of FUNC1, FUNC2, LABOR1-2 and so on. Needless to say that this extends to the compositions of functions such as INCEPOPER1, FINOPER1, INCEPOPER2, FINOPER2, INCEPFUNC1, FINFUNC1 and so on.

3.2 Lexical Ambiguity Resolution

LFs are also useful in resolving lexical ambiguity. For the sake of brevity, we shall give only one illustrative example. The Russian expression *provodit' razlichie* and its direct English equivalent *to draw a distinction* can be analyzed as composed of OPER1 + its keyword. Taken in isolation, the Russian and the English verbs are extremely polysemous, and choosing the right sense for the given sentence becomes a formidable problem. *Provodit'*, for example, has half a dozen senses ranging from 'spend' via 'perform' to 'see off', while *draw* is a polysemic verb for which dictionaries list 50 senses or more. However, in both expressions the mutual lexical attraction between the argument of the LF and its value is so strong that, once the fact of their co-occurrence is established by the parser, we can safely ignore all other meanings and keep for further processing only the one relevant here.

4 Finding Idiomatic Equivalents in MT with the Help of LF

Two important properties of LFs mentioned in Section 2, i.e. their semantic universality and cross-linguistic idiomaticity, make them an ideal tool for selecting idiomatic translations of set expressions in a MT system. As in the previous section, we will confine ourselves to a few clear examples of how this is done.

As is well known, locative prepositions used to form prepositional phrases denoting places, sites, directions, time points, periods, intervals etc. reveal great versatility within one language and incredibly fanciful matching across languages. If we were to account properly for the discrepancies existing between the uses of these prepositions, say, in English and Russian, we would have to write very intricate translation rules involving complicated semantic and pragmatic data. Incidentally, a large share of the task may be achieved with the help of LFs. Consider the following correspondences between English and Russian that may be easily found with the help of the LFs LOC (preposition denoting a typical location):

LOC (*institute*) = *at (the institute)*, LOC (*institut*) = *v (institute)*

LOC (*work*) = *at (work)*, LOC (*rabota*) = *na (rabote)*

LOC (*menu*) = *on (the menu)*, LOC (*menju*) = *v (menju)*

LOC (*North*) = *in (the North)*, LOC (*sever*) = *na (severe)*

In order to ensure the production of these equivalents in MT, we must only identify the arguments and the value of the LF during parsing and substitute the correct value from the target language dictionary during generation.

One of the assets of this approach is that information on the LFs for a given language becomes independent of the respective information for some other language. This is as much as to say that idiomatic translation of LF material among any number of languages can be effected without requiring any revision of the already existing rules. In an MT system which does not resort to the apparatus of LFs, idiomatic translations among a number of languages can also be achieved, but only at the expense of introducing new blocks of transfer rules for every pair of languages.

Another practical advantage of this approach is the fact that the lexicographers who code the data need only know one language (their native tongue) to be able to provide the values of the LFs.

5 An LF-Based Computer System of Paraphrasing Utterances

5.1 An Overview of the System of Paraphrasing

We presume some familiarity with the model of paraphrasing proposed by I. Mel'čuk (see the citations above) and therefore will not expound it here. What is important for us in the present context is that we attempted to formalize and computerize it within the environment of ETAP-3 which is somewhat different from the classical version of MTT framework. In the framework of ETAP-3 paraphrasing Russian sentences turns out to be just one more mode of operation – the mode of Russian-to-Russian machine translation.

Below we give a list of the differences between the MTT and ETAP-3 systems of paraphrasing.

1. As has been mentioned above, ETAP-3 makes use of a single system of tree transformation rules which effect all the necessary changes in the syntactic structure. Therefore the rules of paraphrasing are no longer divided into lexical and syntactic.

2. On the other hand, we have introduced a division of all rules into canonization rules and paraphrasing rules proper. Canonization rules reduce the input sentence to its syntactically and lexically simplest paraphrase comprising just those lexemes which are prototypical exponents of the respective concepts.

Paraphrasing rules proper are used right after canonization rules and yield clusters of paraphrases.

3. Various paraphrasing rules of MTT have incommensurate scopes. Cf. the rule X ⇔

OPER1(S0(X)) + S0(X), with a very broad, if not universal, scope, and the rule X ⇔ A0(X) + GENER(X), with a very narrow scope (cf. *linguistics – linguistic science,* but not *ox - *bovine animal*).

This situation can be quite adequately handled by ETAP-3. It will be remembered that it makes use of three basic types of rules – general rules (for more or less universal phenomena), class-specific rules (to handle compact classes of similarly organized phenomena), and dictionary rules (to handle word-specific phenomena). A rule like X ⇔ A0(X) + GENER(X) will be assigned to the lexeme *linguistics* but not the lexeme *ox*.

4. For a number of reasons, the majority of the original LFs were revised and redefined and some two dozen new LFs with a rich periphrastic potential were introduced.

The resulting model of paraphrasing makes use of all the information from the combinatorial dictionary and most of the rules of ETAP-3. It was implemented on the Russian material; however, it would be a purely technical question to extend it to English.

In the system of paraphrasing the following sets of rules are used: (a) morphological analysis; (b) parsing; (c) LF-interpretation of the syntactic structure; (d) normalization of the interpreted structure (for example, leaving out strongly governed prepositions); (e) canonization rules; (f) paraphrasing rules proper including a block of the simplest word-order rules which preserve the well-formedness of the processed syntactic structure when the introduction of certain LFs requires its profound recasting; (g) reinterpretation and correction rules which prepare the syntactic structure for the input to lexical transfer rules of ETAP-3; (h) introduction of lexical material such as specific prepositions required by the respective LFs; (i) syntactic synthesis; (j) morphological synthesis.

5.2 Some New LFs and New Rules of Paraphrasing

We take for granted the familiar paraphrasing rules of the classical "Meaning ⇔ Text" theory and do not quote them. The emphasis in this section will be on the new rules based on some new LFs.

We shall begin with two semantic types of antonyms based on negation and differing from one another by the location of negation in their semantic representations.

The first type of antonyms is based on inner negation. Consider the well-known pair *begin* and *stop*. *To stop doing P* can be explicated as 'to begin not to do P'. This type of antonymy is called ANTI1. The second type of antonyms is based on outer negation. Consider the pair *observe (the regulations) – violate (the regulations)*; *to violate* means 'not to observe'. This type of antonymy is called ANTI2. For more details see Apresjan 1974.

Two rules of paraphrasing can be naturally formulated on this basis:

(1) X ⇔ Neg + ANTI2(X): *He violated the regulations – He did not observe the regulations;*

(2) X + Y ⇔ ANTI1(X) + ANTI2(Y): *He began to observe the regulations – He stopped violating the regulations.*

One of the familiar LFs of the classical version of the "Meaning – Text" theory is S-RES. Two natural rules of paraphrasing can be formulated on the basis of this function:

(3) X = INCEPOPER1 + S-RES(X): *He learned physics – He acquired the knowledge of physics.*

(4) X = CAUSOPER1 + S-RES(X): *He taught me physics – He gave me the knowledge of physics*.

Note that *knowledge* is S-RES both from *to teach* and *to learn*.

Consideration of this example suggests one more substantive LF complementary to S-RES. S-RES denotes the state which is a result of a certain action or process denoted by the keyword; cf. *knowledge*, *teaching* and *learning* of the previous example. The complementary function we have in mind may be called S-INIT. It denotes a certain state preceding the action or process denoted by the keyword. For example, if we consider the process of *waking up*, the preceding state is apparently *sleep*: S-INIT(*wake up*) = *sleep.* Likewise, S-INIT(*calm down*) = *anxiety*, S-INIT(*stop*) = *movement* and so on. This allows us to formulate the following paraphrasing rules:

(5) X = FINOPER1 + S-INIT(X) = FINFUNC1 + S-INIT(X) etc.: *He came to dislike hiking – He lost his love of hiking – His love of hiking has gone*.

(6) X = LIQUOPER1 + S-INIT(X) = LIQUFUNC1 + S-INIT(X) etc.: *A sudden bell woke him up – A sudden bell interrupted his sleep*.

Now we shall take up a new family of LFs introduced in Apresjan 2001and called REALi-M – FACTi-M. The letter M in the names of these LFs stands for "modality" and is used to refer to keywords like *advice, demand, desire, hope, hypothesis, instruction, law, order, regulation, request, rule* etc. They include senses like 'want', 'may', 'must', 'must not' which account for some peculiarities of their syntactic and semantic behaviour and allow to formulate interesting paraphrasing rules for them. Considerations of space preclude a detailed discussion of these matters, and we shall proceed straight to the rules. We shall quote two rules out of a dozen based on these functions. They have an added value of illustrating one more type of paraphrasing rules – implicative rules that yield not strictly synonymous paraphrases but rather entailments.

(7) CAUSFACT0-M + X / CAUSFACT1-M + X / CAUSREAL1-M + X ≈ INCEPFACT0-M + X / INCEPREAL1-M + X etc. (*They sent him on leave for a few days – He went on leave for a few days*).

(8) LIQUFACT0-M + X / LIQUFACT1-M + X / LIQUREAL1-M + X ≈ FINFACT0-M + X / FINREAL1-M + X etc. (*He was deprived of his last chance to win in this event – He lost his last chance to win in this event*).

For a detailed description of the computer system of paraphrasing see Apresjan & Tsinman 1998, 2002.

5.3 Experimental Data

To give an idea of the paraphrasing experiments we have run on the computer we shall adduce three clusters of Russian paraphrases produced by the computer together with their English translations.

(1) *Pravitel'stvo kontroliruet ceny – Kontrol' cen osushchestvliaetsia pravitel'stvom – Pravitel'stvo osushchestvliaet kontrol' cen – Ceny naxodjatsja pod kontrolem pravitel'stva – Ceny derzhit pravitel'stvo pod kontrolem – Ceny kontrolirujutsja pravitel'stvom* [The government controls prices – Control over prices is exercised by the government – The government exercises control over prices – The prices are under the government's control – The government keeps prices under control – The prices are controlled by the government].

(2) *Pravitel'stvo ustanovilo kontrol' nad cenami – Ceny byli postavleny pravitel'stvom pod kontrol' – Pravitel'stvo postavilo ceny pod kontro ' – Kontrol' byl ustanovlen pravitel'stvom nad cenami* [The government established control over prices – The prices were put under the control of the government – The government put the prices under its control – The government's control over prices was established].

(3) *Pravitel'stvo otmenilo kontrol' nad cenami – Ceny byli osvobozhdeny pravitel'stvom ot kontrolja – Pravitel'stvo osvobodilo ceny ot kontroljia – Ceny byli vyvedeny pravitel'stvom iz-pod kontrolja – Kontrol' byl otmenen pravitel'stvom nad cenami* [The government cancelled control of the prices – The prices were freed of the government's control – The government freed the prices of its control – Control over prices was cancelled by the government].

All in all we have received several hundred paraphrase clusters on the computer. Some of them contained serious mistakes, and it is precisely those clusters that were of exceptional interest to us. A linguistic processor incorporating a serious linguistic theory becomes, by the very nature of things, a gigantic testing ground for this theory. The computer makes mistakes unimaginable for a human. The analysis of such mistakes can be extremely revealing in the sense that it is a shortcut to correcting lexicographic and grammatical descriptions of which its linguistic software is composed. Moreover, very often experimenting with formal models of language on the computer results in genuine linguistic discoveries. Unfortunately, for a number of technical reasons we shall not be able to discuss these matters in detail.

6 Computer-Aided Learning of Lexica

The idea of using lexical functions for a system of computer-aided learning of lexica was conceived at the very beginning of the 90s. The LFs are a tool of formal description of that part of vocabulary which is simultaneously systematic and idiomatic and is therefore most difficult and most important for language acquisition.

In 1993 we developed a prototype of the system using up to a hundred lexical functions and based on the vocabularies of Russian and English counting around five hundred lexical items each. We designed five basic linguistic games supplied with a system of automatic numerical assessment of the user's performance. The first results of this research were reported in Apresjan 1996. Since then work on computer-aided learning of lexica was carried on and now

we have dictionaries of English and Russian designed for the games, each counting up to 3000 items. We first describe the organization of the dictionaries and then the games themselves.

6.1 Dictionaries

6.1.1 Definitions

In addition to the types of information stored in the combinatory dictionaries of ETAP-3 and described above the dictionaries of computer games include one more important type of information – the analytical definitions formulated in a special semantic metalanguage – a reduced, simplified and standardized version of the object language.

The analytical definitions were written by Ju. D. Apresjan and are supposed to meet the familiar requirements of non-circularity, completeness, non-redundancy, reducibility to semantic primitives and systematicity: the set of definitions should be constructed in such a way as to bring out explicitly (by means of common semantic components) systematic semantic links of the given lexeme with other lexemes of the language.

Consider the following two definitions:

X demands of Y that Y should do P = 'Wanting Y to do P and thinking that Y is obliged to do P, X says to Y that he wants him to do P'.

X asks Y to do P = 'Wanting that P should be done and thinking that Y may do it, though he is not obliged to do it, X says to Y that he wants him to do P'.

Such analytical definitions were written for all the entries in the English and Russian dictionaries. Among the three thousand lexical items in each of the dictionaries the principal lexico-semantic classes of both vocabularies are quite well represented. The definitions for every item within the given class were constructed along the same lines. For example, for various types of fruit the definiton had to specify its size, form, inner structure, colour, taste, and the way it is used by humans. For emotions the definition had to specify the factor causing the emotion, the mental evaluation of this factor, the character, intensity and depth of emotion, the desires it calls forth, and its outer manifestations.

Fulfilling these requirements makes the definitions not only systematic but also identifiable by the user, which is an important asset in computer games.

6.1.2 Lexical Functions

The set of around a hundred LFs was defined anew with a view to providing a systematic description of the algebra of LFs, on the one hand, and making the definitions transparent for the user, on the other. We shall give sample definitions of a dozen LFs used in the linguistic games. In the definitions, X stands for the keyword (the argument of the respective LF), P1 for its first actant, P2 for its second actant and P0 for a non-participant of the situation (the causer of some situation).

OPER1 [to do X, to have X or to be in the state of X (a support verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to make (a choice), to take (an interview), to give / to take / to throw (a look), to have (the majority) / be in (the majority).*

OPER2 [to undergo the action of X or to be in the scope of X (a support verb taking P2 as its grammatical subject and X as its principal complement)]; cf. *to receive (a blessing), to be under (the influence), to give (an interview), to enjoy (smb's respect), to undergo (a test).*

INCEPOPER1 [to start to do X, to have X or to be in the state of X (a support verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to catch (the grippe), to acquire (importance), to fall in (love), to get / to receive / to win (the majority), to come to (power).*

FINOPER1 [to cease to do X, to have X or to be in the state of X (a support verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to break off (the acquaintance), to go out of (business), to lose (control), to part with (life), to cease (the struggle).*

REAL1-M [to do with regard to X that which is normally expected of P1 (a verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to do (one's duty), to fulfil (the obligation), to realize (one's plan), to follow (a principle).*

INCEPREAL1-M [to start to do with regard to X that which is normally expected of P1 (a verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to learn (the news), to accept (a principle).*

FINREAL1-M [to cease to do with regard to X that which is normally expected of P1 (a verb taking P1 as its grammatical subject and X as its principal complement)]; cf. *to give up (the attempt), to refute (the obligation), to abdicate from (the throne).*

CAUSREAL1-M [to cause P1 to do with regard to X that which is normally expected of P1 (a verb taking P0 as its grammatical subject, P1 as its principal complement and X as its secondary complement)]; cf. *to put (somebody) under (an obligation), to bind (somebody) with (a promise).*

LIQUREAL1-M [to cause P1 not to do with regard to X that which is normally expected of P1 (a verb taking P0 as its grammatical subject, P1 as its principal complement and X as its secondary complement)]; cf. *to release (somebody) from (a debt), to relieve (somebody) from (a duty), to free (somebody) of (an obligation)*, *to strip (somebody) of (a right).*

REAL2-M [to do with regard to X that which is normally expected of P2 (a verb taking P2 as its grammatical subject and X as its principal complement)]; cf. *to break (the blockade), to accept (a challenge), to pass (an examination), to avenge (an insult).*

6.2 Linguistic Games

The system offers the following four linguistic games based on the analytical definitions of lexemes and the LFs assigned to them: 1) supplying translations for a lexeme offered by the

computer; 2) supplying the values of all the LF's (offered by the computer) for a word chosen by the user; 3) supplying the values of a LF (chosen by the user) for all the lexemes offered by the computer; 4) guessing a particular lexeme from its analytical definition (decomposition) offered by the computer.

In the basic mode the user is supposed to play those games unilingually. However it is possible to duplicate the second game in the bilingual mode whereby the search for the right answer is guided by the user's knowledge of his/her mother tongue, irrespective of what the mother tongue in question is.

That forms the fifth game called "Word with a tip".

Depending on the semantic and syntactic properties of a particular LF the games are arranged into three levels of difficulty.

The first level of difficulty is composed of LF's whose meaning is transparent and whose syntactic (or actant) structure is simple. Cf. such LFs as SYN or MAGN.

The second level of difficulty is composed of the LFs with a straightforward meaning but a complicated syntactic (or actant) structure; cf. the LFs OPER-FUNC family.

The third level of difficulty is composed of the LF's with a complicated meaning and an involved syntactic structure. Cf. the REAL-FACT family.

Each level of difficulty is broken into two sublevels, depending on the foregroundedness of the value of a particular LF. The prototypical values of LFs are presumed to be in the foreground of the user's language competence and, if properly guessed, score less than more peripheral values.

The games are supplied with a system of quantitative assessment of the extent of lexical knowledge the user displays. The first level of difficulty scores the user one point for every correct answer. Every next level of difficulty scores him or her one point more. The user's performance is considered to be normal if he or she supplies at least one correct answer for every question. It is considered to be excellent if the user supplies more than one correct answer, in which case he or she is rewarded with ever greater scores for every answer beyond the “standard”.

At every stage in every game the user is shown the number of points he/she has scored for the current answer and his/her total score for the game.

A computer-driven dialogue system has been devised intended for specifying a) the language the user wishes to practise in, b) the linguistic game, and c) the level of difficulty.

All the games are supplied with exhaustive menus so that there is no need for further comments. Programming facilities are envisaged providing for constant updating and expansion of the dictionaries and all the other linguistic resources used in the games.

References

Apresjan Ju. (1974): *Leksicheskaja semantika. Sinonimicheskie sredstva jazyka* [Lexical Semantics. The Synonymic Devices of language]. Moscow, Nauka.

Apresjan Ju. (1996). Enseignement du lexique assisté par ordinateur. *Lexicomatique et dictionnairiques. IVes Journées scientifiques du réseau thématique "Lexicologie, Terminologie, Traduction".* Lyon, France, 28, 29, 30 septembre 1995. Montréal, 1-10.

Apresjan Ju. (2001): O leksicheskix funkcijax semejstva REAL – FACT [On the Lexical Functions of the REAL – FACT Family]. *Nie bez znaczenia … Prace ofiarowane Profesorowi Zygmuntowi Saloniemu z okazii jubileuszu 15000 dni praci naukowej*. Białystok, 23-40.

Apresjan Ju., Boguslavsky I., Iomdin L. *et al*. (1992). ETAP - 2: The Linguistics of a Machine Translation System. *META*, Vol. 37, No 1, pp. 97-112.

Apresjan Ju., Tsinman L. (1998). Perifrazirovanie na komp'jutere. [Paraphrasing on a Computer]. *Semiotika i informatika*, No. 36, pp. 177-202.

Apresjan Ju., Tsinman L. (2002). Formal'naja model' perifrazirovanija predlozhenij dlja sistem pererabotki tekstov na estestvennyx jazykax [A Formal Model of Sentence Paraphrasing for NLP Systems]. *Russkij jazyk v nauchnom osveshchenii*, No. 4, pp. 102-146.

Arsentyeva N. , Balandina, N., Krasovskaja A. (1969): O mashinnoj realizacii sistemy perifrazirovanija [On the Computer Implementation of a Paraphrasing System]. *Institute for Applied Mathematics, Academy of Sciences of the USSR.* Preprints 25, 26, 27. Moscow.

Boguslavsky I., Frid N. Grigoriev N., Grigorieva S., Kreidlin L. (2000a). Dependency Treebank for Russian: Concept, Tools, Types of Information. *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, 987-991.

Boguslavsky I., Frid N., Iomdin L., Kreidlin L., Sagalova I., Sizov V. (2000b). Creating a Universal Networking Language Module within an Advanced NLP System . *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, 83-89.

Boguslavsky I. (2001a). UNL from the linguistic point of view. *Proceedings of The First International Workshop on MultiMedia Annotation. Electrotechnical Laboratory SigMatics,* Tokyo, 1-6.

Boguslavsky I. (2001b). Some lexical issues of UNL. *Proceedings of the First International Workshop on UNL, other interlinguas and their applications*. Las Palmas, 19-22.

Boguslavsky, I. Chardin I., Grigorieva S, Frid N., Iomdin L., Kreidlin L. (2002). Development of a dependency treebank for Russian and its possible applications in NLP. *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, v. III, Las Palmas, 852 – 856.

Iomdin L., Tsinman L. (1997): Lexical Functions and Machine Translation. *Proceedings of Dialogue'97 Computational Linguistics and its Applications International Workshop.* Yasnaya Polyana.

Mel'čuk I. (1974). *Opyt teorii lingvisticheskix modelej "Smysl ⇔ Tekst"* [A Theory of Meaning ⇔ Text Linguistic Models"]. Moscow, Nauka, 314 p.

Mel'čuk I. (1996). Lexical Functions: A Tool for the Description of Lexical Relations in Lexicon. *L. Wanner (ed.), Lexical Functions in Lexicography and Natural Language Processing.* Amsterdam, 37-102.

Mel'čuk I. (1997). *Vers une linguistique Sens-Texte. Leçon inaugurale*. Paris.

Mel'čuk I., Zholkovskij A. (1984): *Tolkovo-kombinatornyj slovar' sovremennogo russkogo jazyka.* [An Explanatory Combinatorial Dictionary of the Contemporary Russian Language] Wiener Slawistischer Almanach, Sonderband 14, 992 p.

Mel'čuk I., Arbatchewsky-Jumarie N., Iordanskaja L., Lessard A. (1984). *Dictionnaire explicatif et combinatoire du français contemporain, Recherches lexico-sémantiques I.* Les Presses de l'Université de Montréal.

Mel'čuk I., Arbatchewsky-Jumarie N., Dagenais L., Elnitsky L., Iordanskaja L., Lefebvre M. N., Suzanne M. (1988). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques II.* Les Presses de l'Université de Montréal.

Mel'čuk I., Arbatchewsky-Jumarie N., Iordanskaja L., Suzanne M. (1992). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques III*. Les Presses de l'Université de Montréal.

Mel'čuk I., Arbatchewsky-Jumarie N., Iordanskaja L., Suzanne M. Mel'čuk, Polguère A. (1999). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques IV.* Les Presses de l'Université de Montréal.

Mel'čuk I., Wanner L. (2001). Towards a Lexicographic Approach to Lexical Transfer in Machine Translation (Illustrated by the German-Russian Language Pair). *Machine Translation*, No. 16, 21-87.

Polguère A. (1998). Pour un model stratifié de la lexicalisation en génération de texte: *t. a. l.*, 39 (2), 57-76.

Streiter O. (1996). *Linguistic Modeling for Multilingual Machine Translation*. Shaker Verlag. Aachen.