A computer model for Russian grammatical description, and a method of English synthesis in machine translation

D.M. Yates (National Physical Laboratory, Teddington)

Introduction

This paper is the second of two from the NPL MT group at this conference. It describes a model designed to express the grammatical facts discovered by the Russian analysis algorithm in such a way that they can be used directly by the English synthesis algorithm. The general nature of this synthesis process is the subject of the second part of the paper.

The model: linguistic features

Russian and English have many important categories in common. For instance, both have subjects, verbs, objects, nominal groups, conditional clauses and so on. When it comes to finer details, though, the differences between the two languages become more noticeable than the similarities: the use of auxiliary verbs to represent tenses, for instance, is quite different (e.g. did not ask = не спросил)

The basic task of this model is to provide a means of representing in the computer any Russian grammatical structure which the analysis algorithm may have to express. As far as possible this representation must be independent of the particular conventions of either language. For example не спросил would not be ascribed any internal structure, but would be represented as "спрос-/ask, negative, past tense". The analysis would discover these facts, concerning itself only with Russian conventions, and the synthesis would express them in English, concerning itself only with the English conventions. "Negative" and "past tense" are examples of choices within closed sets of possibilities. Such sets are known as systems. Our model therefore has two main linguistic features, structure and system, which will both be needed to describe a Russian sentence. This terminology is taken from the work of Halliday (1961).

The structure is fundamentally a hierarchy of constituents, but there are four ways in which it differs from a conventional constituent structure:

(1) Each constituent may exemplify choices in systems, and, as illustrated above, this means that some units in the text (e.g. particles and auxiliaries) are not given places in the structure.

(2) One item may occupy more than one place in the structure. The only need for this in scientific Russian seems to be

the dual role of a relative word in linking a subordinate clause to some higher constituent and at the same time taking some role within its structure.

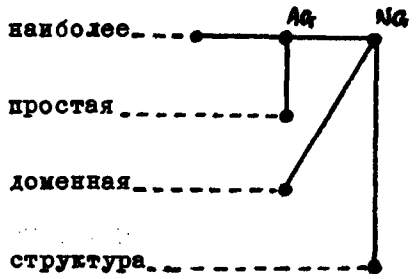
(3) There is no requirement for a constituent to be continuous in the text (although those found by the current analysis algorithm always are).

(4) If the systems are powerful enough there is no need for explicit ordering of subconstituents. This point will be taken up again later.

The model: computing features

A grammatical structure of word-groups is represented in the computer by a list structure, that is to say a collection of stored items called elements with the property that each element either (i) contains addresses of one or more other elements, or (ii) is marked as a terminal element. The terminal elements represent single items (words or idioms); the other elements each represent a larger word-group or constituent of the sentence. If an element A contains the address of an element B, this represents the fact that word-group A includes word-group B.

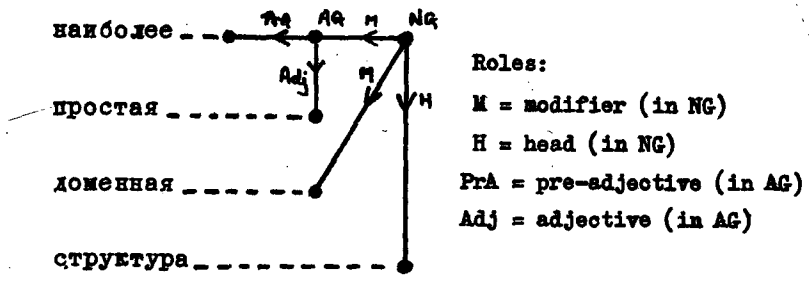
For example, the description of the structure of the group наиболее простая доменная структура includes four terminal elements (for the four words) and two other elements, linked as follows:



(AG = adjectival group,
NG = nominal group)

Each element is labelled with a code giving the constituent type (noun, verbal group, etc.), and each address referring to an included word-group is labelled with a code giving the role of the smaller group in the larger one (complement in prepositional group, for instance).

With roles included, the above description becomes:



Choices in systems are also represented in a label in the element concerned. This label is called the systems word. In the above example, the systems word in the nominal group element records the number, gender, and case of the group.

In theory, the observed order of items is either evidence for a particular structure (as in the order of prepositions and their complements), or evidence for a choice in a system (as in the order of auxiliary and subject in English interrogative sentences). Just the same is true of punctuation (some commas indicate structure, e.g. those marking clause boundaries; others indicate a choice in a system, e.g. those distinguishing 'descriptive' and 'restrictive' qualifiers in nominal groups). Ideally then the model would have no need to represent item order or punctuation explicitly: it would record the structures and systems, and the synthesis algorithm would have a free hand in determining the English order and punctuation according to English structural and systemic rules. But in practice the language features concerned are not yet understood in sufficient detail, so the synthesis keeps the original order and punctuation except where it has some reason to change them. This means that they need to be recorded in the model statement. The addresses in an element are therefore stored in the same order as the constituents to which they refer, and each element includes details of any punctuation surrounding the constituent.

The full list of constituent types and roles is as follows:

Constituent	Subconstituents' roles
Nominal group (NG)	Modifier (M̄) Head (H) Qualifier (Q) Appositive (Ap)
Adjectival group (AG)	Pre-adjective (PrA) Adjective (Adj) Post-adjective (PtA)
Prepositional group (PG)	Preposition (Pp) Complement (Ct)
Adverbial group (ADV)	Pre-adverb (PrA) Adverb (Adv) Post-adverb (PtA)
Verbal group (VG)	Verb (V) Complement (Ct) Adjunct (At)
Coordinate group (CG)	Conjunction (Cj) Member (Mb)
Clause (CL)	Subject (S) Predicate (Pd) Adjunct (At)
Subordinate clause (SC)	Conjunction (Cj) Clause (Cl)
Complex clause (CC)	Clause (Cl) Adjunct (At)
Comparative group (CPG) (e.g. как + noun)	Link (Lk) Comparison (Cp)
Prefix group (PFG) (e.g. вектор-функций)	Prefix (Pf) Stock (St)

Although most of the terminology in the table will be self-explanatory, it should be made clear that in a co-ordinate group the 'members' may be constituents of any type. Likewise the prefix group is a general one, the 'stock' being noun, adjective, or verb. (In practice, for reasons of programming convenience, the prefix group was not used, such groups being represented by the 'stock' alone, tagged with the reference number of the prefix).

The table attempts to provide an adequate set of constituent types and roles for the description of sentences in our texts. It should not be inferred that our analysis processes could recognise all these features; indeed the clauses and the comparative group were not used at all.

Associated with each type of constituent there are certain systems. For example, a clause may be either non-finite (если импульс подать ...) or finite. If finite, choices of mood (interrogative/imperative/declarative), conditionality, and personality will have been made; and if the clause is personal there will be selections of person and number. All these systemic choices would be recorded in the element representing the clause.

Below, an example is given of the structural description of a complete sentence; again it is not a structure which the current analysis could produce, but is intended simply to illustrate the use of the model.

Example of sentence structure description

(N.B. С помощью is treated as one item since it is included in the dictionary as an idiom).

The English synthesis algorithm

The synthesis algorithm has the task of taking a sentence expressed in terms of the model described above, and producing from it the string of characters which form the English output sentence.

The program uses the model statement to guide it in decisions on:

(1) re-ordering;

(2) insertion of English 'function' words (auxiliary verbs, etc.);

(3) selection of English equivalents from the short list in each dictionary entry;

(4) inflection of English equivalents.

These decisions are of course based on grammatical data only (both structural and systemic); in particular in the selection of equivalents no semantic or collocational techniques are used.

The particular tasks under these headings which are appropriate to a particular type of constituent will in general need to be carried out whatever the role of the constituent in some higher structure may be; and we are therefore led to the need for a separate routine for each constituent type. Such a routine will be called a constituent type procedure (CTP). The nominal group CTP, for example, will be called upon when and only when a nominal group has to be produced by the program.

Since constituents nest within one another freely, one CTP will need to call on others to deal with the parts of the constituent in turn. The CTPs must in fact be written as fully recursive subroutines; and the program consists basically of a control routine for exploring the list structure together with a set of CTPs, one for each constituent type.

As was pointed out by Yngve (1960), it is a linguistic fact (at least in the Indo-European family of languages to which Russian and English both belong) that in many constituents the final sub-constituent is a group of words, while other sub-constituents are more frequently single items. Thus multiple "nesting" of the CTPs usually involves final subconstituents. But in these cases all details of the higher constituent can be "forgotten" by the computer since that constituent will not need to be returned to; so even a long sentence needs no great depth of push-down store to handle the nested CTPs. (Language has presumably evolved in this way because of an analogous advantage

in the brain).

The first task of a CTP is to decide on any re-ordering needed. It implements such a decision simply by rearranging the addresses in the element concerned. Each CTP entered does this, so that the individual items are met in their new order and can be added to the output string at once.

The selection and inflection of equivalents are carried out at the time they are to be produced, when all relevant information is available to the CTP without excursions into other parts of the structure. The insertion of function words, on the other hand, may be done by any CTP.

The resulting English output string is then passed to a final program which is responsible for format control. The normal form of output is punched paper tape, from which the printed copy, as shown in McDaniel et al. (this conference), is produced on a 'Flexowriter'. There is an alternative form of output on punched cards, from which printed copy can be produced on a card-controlled typewriter. This earlier form gives the text in the two languages side by side, which was useful for research purposes, but the absence of lower-case Roman letters and pagination, and the restricted width of each language version, makes this form less well suited for general use.

This format control process, and the main control routine which deals with the exploration of the tree and the handover from one CTP to the next, need not be described further, but the tasks of the individual CTPs will be outlined below.

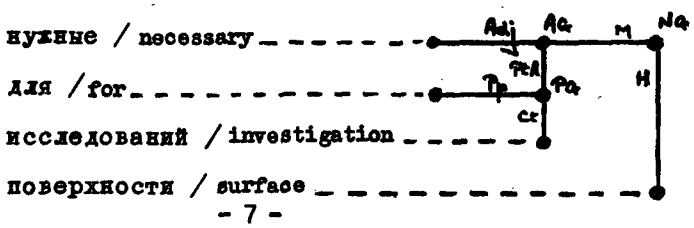
Tasks of nominal group CTP

(1) To insert before the group a preposition depending on the case and role of the group, e.g. of is inserted if case is genitive and role is qualifier in NG. Several instances occur in the sample output referred to above.

(2) To move modifiers containing items after the adjective or participle to the end of the group, with appropriate punctuation.

Example:

Structure as received from analysis:

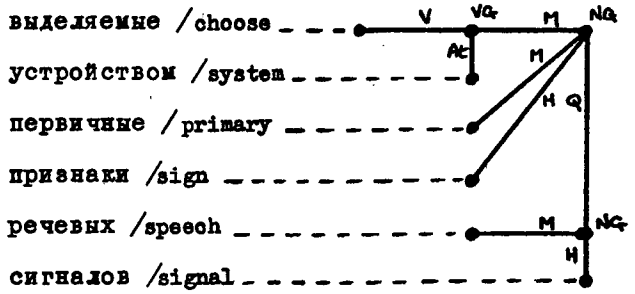


Result:

surfaces necessary for investigations

In a more complex case commas are inserted.

Example:



Result:

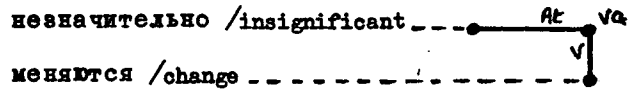
primary signs of speech signals, chosen by system.----

Tasks of verbal group CTP

(1) To insert auxiliary verbs and 'not' as necessary in finite verbal groups, for instance inserting does not for the present tense 3rd person singular negative. The precise rules for the position of the insertion are complex, but roughly these words are inserted immediately before the verb in negative verbal groups and before the verb and any immediately preceding adverbs in positive verbal groups.

Example:

Structure as received from analysis:



The VG has systems coding 3rd plur., present, ся passive positive. The VG CTP therefore outputs are and hands control to the adjective CTP (since the dictionary entry

for the first word is an adjectival one). As described below, this CTP will output the adjectival equivalent with an adverbial inflection -ly. The verb CTP then generates the verbal equivalent again with the appropriate inflection.

Result:

are insignificantly changed

(2) The VG CTP also inserts auxiliary verbs before "short form" predicative adjectives and participles, and inserts to before infinitives, in both cases with appropriate placing of not and any adverbs.

(3) Special measures are taken to allow for the non-standard behaviour (as regards English auxiliaries) when equivalents include be, should or can.

(4) The CTP is so arranged that a treatment of government phenomena could be added conveniently. The routine concerned was developed only as far as the flowchart stage.

Tasks of clause CTP

The principal task of this CTP is to determine the order of subject, verb and complements. For example, if in Russian a sentence begins with an intransitive verb, and the subject follows, the preferred translation depends on the length of the subject-short subjects can be put before the verb, but with long subjects this would not be acceptable in English and some expedient, such as the insertion of the dummy subject there, must be adopted (e.g. Then there arose the problem of ....).

Unlike the other CTPs described, this one was not implemented, being developed only as far as the flowchart stage. In its absence, certain pronominal subjects are inserted by ad hoc methods.

Tasks of noun, verb and adjective CTPs

Apart from certain insertions (such as having before past verbal adverbs), the main task of these CTPs is inflection. The decision to inflect is based on the systems coding and, in the case of adverb formation, on the role given to the item by the analysis. The actual type of inflection is chosen according to a code in the dictionary associated with each correspondent; thus boundary will be pluralised as boundaries , foot as feet , and so on. (Irregular forms such as feet are extracted by the program from a list, using an address given in the dictionary entry. Including both nouns and verbs, this list contains 212 forms). Provision is made for inflecting the right word in multiple word correspondents such as mode of life . All vagaries of English inflection called for by present

dictionary equivalents are covered.

Selection of equivalents

There are five CTPs which select equivalents on various grammatical criteria, usually the role of the item. A typical case is that dealing with 'noun/adjectives' such as другой This ensures that другими авторами is translated by other authors, while границ другой is translated as boundary(s) of another (assuming, of course, that the analysis has given them structures of modifier-head and head-qualifier respectively).

Conclusion

The model and synthesis algorithm described proved satisfactory in practical use. They have the advantage that translations can be produced when the algorithms are incomplete: provided the sub-trees produced by a partial analysis are linked arbitrarily to produce a single sentence structure, this can then be explored by a synthesis algorithm, even one in which several CTPs are replaced by dummies. As new packages (analysis passes or synthesis CTPs) become available they can be incorporated very simply.

The work described above has been carried out at the National Physical Laboratory.

References

1. Halliday, M.A.K. (1961) - Categories of the theory of grammar. Word, 17, (3), pp. 241-292.

2. McDaniel, J. et al. (1967) - An evaluation of the usefulness of machine translations produced at Teddington, and an account of the translation methods. (This conference).

3. Yngve, V.H. (1960) - A model and an hypothesis for language structure. Proc. Am. Phil. Soc., 104, (5), pp. 444-466.