[*Mechanical Translation*, vol.4, no.3, December 1957; pp. 76-78]

A Refinement in Coding the Russian Cyrillic Alphabet

B. Zacharov, London University, London, England

By reducing the number of characters to be coded the problem of devising a numerical code for the Cyrillic alphabet can be simplified. This reduction can be achieved by providing code-words for only the lower-case forms of characters that do not occur initially; by disregarding the diacritic of the character ë, and by disregarding the character ë entirely. Ambiguities that arise in the latter cases can be resolved by an examination of the context.

THE PROBLEM of coding the Russian Cyrillic alphabet in numerical form has been considered previously in several papers [1] and it is clear that it would be desirable if each character of the Russian alphabet (together with any required numbers, punctuation marks and capitals) could be coded in such a way that a separate unique numerical code-word existed for each lower-case character, capital, etc. Unfortunately, the speed of modern digital computers and the size of their memories are such that a code of this form would result in considerable time being spent in the memory search for the appropriate target language equivalent.

It is clear, then, that ways must be found, apart from engineering advances, to speed up the memory search time. One way of doing this would be to decrease the amount of linguistic data stored in the memory, and this has been considered. [2] Another method would be to decrease the amount of numerical data (i.e., the number of bits) in the memory for a given number of source language characters. This last approach has been considered in a recent paper on mechanical translation[3] where all the lower-case characters, except ë, и, ъ and ь are represented by a five binary-digit code, while all the capitals and decimal numbers use a ten bit code; in the code proposed in that paper simplification is obtained on the basis of the statement that "... five of the 33 Russian letters never start a word and will not need to be capitalized ... ". The five Russian letters referred to are ë, и, ъ, ь, ы.

All the other Russian characters occur frequently in both upper and lower case and require to be coded separately in both these forms or by the same numerical code, except that the upper case is always preceded by some number which denotes an 'upper-case shift'.

Inspection of the statement quoted above reveals that it is formally incorrect with respect to ë although it is quite correct to state that none of the four characters й, ъ, ь, and ы ever begin a word in the Russian language so that clearly, it will never be necessary for them to be coded in upper-case form. (A rigorously phonetic transliteration of some other alphabet into Russian may create a trivial exception in the cases of й and ы This will not be considered here.)

---

1. Harper, K.E., "The Mechanical Translation of Russian: Preliminary Report", Modern Language Forum, vol.38, no. 3-4, pp. 12-29, Sept. - Dec. 1953.

2. Oettinger, A. G., "The Design of an Automatic Russian-English Dictionary", Machine Translation of Languages, John Wiley and Sons, New York (1955), pp. 47-65.

3. Wall, R. E., "Some of the Engineering Aspects of the Machine Translation of Languages", AIEE Transactions, I, vol.75, 580 (1956).

The Problem of ë

Reference to a Russian-English dictionary[4] shows us that many words of the Russian language begin with ë Notable examples are ёлка 'fir tree' and ёмкость 'capacity'; the latter is of especial importance in scientific texts.

Superficially, therefore, it would appear that ë should be treated in the same way as the other word-initial characters and that it should be coded in upper and lower case. However, the following points must be considered,

i) In practice, ë is never written in script form with the diacritic, either in lower or upper case — e and E are used.

ii) A modern standard Russian typewriter keyboard does not contain Ë or ë — the upper and lower case forms of e are used, as in (i).

iii) Both ë and Ë frequently appear in print, especially in the texts of scientific periodicals .

Thus, from (i), (ii) and (iii) above, it can be seen that the problem of encoding ë and Ë is complicated by the source of the Russian language text. If e and ë are coded separately, it would appear that words containing ë would have to be stored in the memory in two separate locations, with both e and ë in the corresponding positions of each word.

a) ë at the beginning of a word

For words with ë at the beginning, any coding difficulty can be overcome if it is noted that, if the diacritic is ignored, no ambiguity can arise. This is because no two words in the Russian language exist with different meaning such that corresponding letters of both words are the same except that ë at the beginning of the first word is replaced by e in the second word. As a result of this consideration it will clearly never be necessary to encode ë in capitalized form — the upper-case form of e will be sufficient.

b) ë in any letter position

If ë occurs in some letter position other than at the beginning of some word (x), ambiguity can arise only if another word (y) exists such that all the letters of the (y)-word are the same as the corresponding letters of the (x)-word except that ë in (x) is replaced by e in (y).

Examination of a Russian-English dictionary reveals that this does not occur often in the stem of a word. Similarly, experience tells us that ambiguity seldom arises as a result of word endings together with stem.

Examples of words where ambiguity may occur are:

все	all (plural)
всё	all (singular, neuter)
села	of the village (genitive, singular) / she sat
сёла	villages (nominative/accusative, pl.)

Whereas discrepancy need not necessarily occur in the first example, considerable ambiguity can arise in the second case since the words are different grammatical forms of widely different words (сёла is a plural noun while села may be a verb form or a singular noun).

However, we note that if the contexts of these words are examined, most cases of ambiguity disappear (this is especially true for Russian where strict grammatical rules concerning case endings and conjugation must be observed). Indeed, such an examination is essential for certain words in Russian and, more especially, in English. [5]

Certain Russian words are such that their spelling is associated with multiple meaning and, here, it is often the case that an examination of the context will not reveal which alternative is meant. In this event it becomes necessary to print out all the alternatives stored in the computer memory which correspond to the source word. At this stage a simplification may be effected if the computer dictionary is concerned only with a certain field (e.g., nuclear physics), in which case only those terms which may reasonably be expected to relate to that field will be printed out.

Examples of Russian words in such a category are:

замок	castle / lock
замотать	twist / shake

4. Smirinskii, A.I., Russian-English Dictionary, State Publishing House for Foreign and National Dictionaries, Moscow, (1952).

5. Yngve, V.H., "Syntax and the Problem of Multiple Meaning", Machine Translation of Languages, John Wiley and Sons, New York (1955), pp.208-226.

In the two examples above, ambiguity will disappear if the words are used in idiomatic context (e.g. padlock = висячий замок).

In the case of words containing e or ë, however, difficulties of multiple meaning that cannot be resolved by simple context (i. e., syntax) examination are very rare. In fact, in the author's experience, no example can readily be quoted.

Suggested Encoding Rules

From the above considerations, a set of rules can be formulated to include words containing ë and Ë. They are:

i) Source language words containing ë or Ë are stored in the dictionary in numerical form as if they contained e or E in the corresponding letter positions,

ii) Incoming source language words are coded with a unique number code for every lower-case character except ë which is treated as if it were e. All upper-case characters will have unique number codes corresponding to them (or they will be preceded by a coded upper-case symbol), except Ë, where the diacritic is ignored and the character is treated as if it were E; й, ъ, ь, and ы will have no upper-case code,

iii) If more than one target language alternative is found, the context of the Russian language word must be examined; this will also be required for any other word (not containing e or ë) where ambiguity may exist — as in the examples above.

The Problem of ъ

It may be noted that ъ could also be ignored completely since it occurs so very rarely in the Russian language. This may be of some importance since the character can be represented in several different ways, namely:

i) as ъ.
ii) as '
iii) as a gap in a word
iv) it is ignored completely.

As in the above encoding rules, if ambiguity occurs because ъ is ignored, the context of the word must be examined. An example of words where this kind of difficulty can arise is

сесть	=	sit down
съесть	=	eat

In these cases, if a unique meaning cannot be found simply from the program, all the target-language equivalents will have to be printed out and the required meaning determined by post-editing.

From an examination of the occurrence of e in the Russian language it seems that, if the diacritic is ignored the chances of ambiguity occurring in MT, with the rules formulated above, are very slight. Indeed, for a specific subject, where all the source language words in the dictionary are known, most cases of ambiguity and difficulties of multiple meaning could be overcome by sufficiently sophisticated programming techniques (i.e., syntactical and idiomatic context examination for all the cases of expected ambiguity).

As to ъ, it may be ignored in the encoding. The few cases of ambiguity will be resolved from a study of context.