Cross-Language Transcription of Proper Names

Edward Klyshinsky, Vadim Maximov, Sergey Yolkeen

Computational Linguistic Laboratory
M.V.Keldysh Institute of Applied Mathematics, Russian Academy of Sciences
Miusskaya sq. 4., Moscow, 125047 Russia
ipm4info@gmail.com
http://www.keldysh.ru/pages/cgraph/ASOLI/asoli.html

Abstract. The paper is dedicated to the problem of automatic cross-language transcription of proper names. The correct written transcription of foreign proper names is a serious communication problem. It is especially important for legal translation of documents, data retrieval, postal processing and, in general, in all fields, where the accurate identification of places, persons and organizations is required. In order to formalize the process of transcription and reduce the number of errors, the automatic rule-based system of transcription has been developed. The system transcribes proper names between more than 20 languages, including non-European ones. The phonetic approach provides the easy integration of new languages in the system. The results of long-term collaboration of linguists and programmers had been generalized in the monograph.

1 Introduction

The accurate transfer of foreign proper names pronunciation by means of written transcription is an actual communicational problem, as it provides for the proper identification of places, people and organizations. The automatic solution of this problem will help to avoid certain mistakes in data retrieval in multi-language environment; then, it will facilitate the work of those specialists whose activity is connected with the transcription of foreign proper names: translators, postal operators and others who must process the foreign names in written form. The formalization of transcription rules will make it possible to enter the new quality level of cross-language transcription. Furthermore, the automatization of transcription will make it possible to fix once and for all some variants in spelling of proper names, which are the subject of dispute for linguists until now.

The practical transcription is such a transfer of foreign words sounding from source language into the target one, which keeps the phonetic image of word with maximal similarity. If one and the same sounding could be expressed by various letters or combination of letters, then the variant, which is maximally similar from graphical point of view, shall be chosen. The practical transcription is used in processing and registration of documents and machine-readable texts. Now it is possible to find three methods of proper names transfer:

- *Translation*. According to this method each name of frequent occurrence corresponds to some equivalent in the target language, which had become historically accepted for the current moment;
- *Transcription.* (i.e. practical transcription). This is the method, according to which the foreign proper name corresponds to such word in the target language, that reproduces its sounding with maximal accuracy, that only could be achieved in the target language.
- *Transliteration*. This method provides "letter-by-letter" transfer of proper names, recorded by means of graphical system of original language, to another form of record by means of graphical system of target language.

The existing methods of cross-language proper name transfer and the respective software products provide no acceptable solution of this problem [2]. Then, obviously, it is the practical transcription method, that shall be chosen to transfer proper names from foreign language into the target one, for it shows the most adequate results for this purpose. In the context of automatic solutions, this method shall be recognized as dominating too, because *translation* and *transliteration* reveals considerable disadvantages. Translation method could work only if huge and constantly updated base of proper name are provided, while *transliteration* often provides no phonetically adequate equivalents for the original language.

2 Basic Problems

While creating the experimental machine transcription system, we had met the following problems.

1. Some countries have several national transcription and transliteration systems of transfer into Latin alphabet. Such systems often enter into competition. As an example, we could take the China Pinyin and Wade-Giles systems, Japan Romaji systems, Hepburn, Nihonsiki and Kunreisiki, Russian conversion systems: GOST 16876-71, ISO 9, the system US Congress Library, Russian Academy of Science Standard etc.

2. Countries usually possess system of transcription into Latin, but if we will take some other target system, such as Cyrillic, some countries have no developed transcription systems for it at all. Some most widespread variants usually are determined unambiguously, while other more rare combinations of letters could stay unclear and undefined at all. In Arabian and Turkish languages one may find many examples of such state of affairs.

But even in English, which is one of the most widespread languages in the world, there are many complexities and ambiguities. The rules of practical transcription for English language for the present moment are based on phonetic transcription, but the historical evolution of English spelling had resulted in some significant discrepancies with the pronunciation. This is the reason why sometime it is impossible to determine the proper variant among the possible ones. So, English pairs of letters *ou* and *ow* could correspond to the diphthong [ou] such as in examples: *Barrow ['bærou], Boulder ['bouldə]*. But they as well may correspond to the diphthong [au], such as in the example: *Founder → ['faunldə]*. There are no methods to point out the rule, which will eliminate the ambiguity. No one can be sure that in some unknown word the pairs of letters *ou* or *ow* must be read in a certain way. So, there are two potentially

possible variants of phonetic interpretation of proper name with no preferences of one variant over another.

3. In addition to the phonetic ambiguity in construction of word sound image, there are certain difficulties in adequate labeling of the sounds that do not exist in target language. This fact leads to the situation, where in the transcription system the sounds of original language are placed only in approximate correspondence with the sounds of target language. As the result, the urgent phonetic information such as *continuance, palatalization, tone height* and other could be lost. For example, in case the target language is French, there is no difference between such English sounds as [t] and [θ], and both sounds are labeled through the letter '*t*'. But there could arise some additional ambiguity if the sound of original language does not exist in the target language and can be expressed by various sounds. Such sound as [w] can be presented by Russian [u], [v] or [æ] – even by three different letters. This situation causes different variants of transcription and a lot of disputes among linguists which variant must be preferable. Sometimes it is not easy to give the confident answer, because each variant could have its benefits and implications.

4. The lack of unambiguous correspondence in the process of transcription give rise to just one more problem. If we transcribe a word from original language into the target one and then back to original language, the result of such double transcription in many cases will not coincide with the original. The discrepancy results from the loss of phonetic information in the process of transcription. This problem is connected with the lack of sounds in the target language, which are present in the phonemes of the original language.

According to the international requirements, the machine-readable documents must be written by Latin letters. In this connection, in the process of transcription the original word loses all its specific letters which are usually labeled with the help of diacritical marks. It causes the additional loss of information, and due to this fact such wide-spread Korean surname as Choi will be transcribed as "Chkhve" in Russian, and then it will appear in international documents as Chhwe. Chinese Zhongzhou after such double transcription will appear as Chzhunchzhou.

5. Just one more problem arises, when we transcribe a proper name from some language and this proper name is not native for this language. For example, if we transcribe Mexican name from English, we must follow the rules of "original", i.e. English language, and *Jose Enrique Martinez* will appear as [jōus in'raik].

6. And in the end, we must mention the "struggle" between the modern and historical rules of transcription. It creates additional discords in this field.

Here we have listed only the major problems that arise in practical transcription of machine-readable documents. Such situations as substitutions of discordant phonemes and replacements in case of objectionable associations are not discussed in the present paper.

The development of practical transcription method will allow to formalize and fix some mechanism and rules of transcription. It will help to eliminate partially the problems listed above. I this context the thorough development of formal methods of transcription and the respective software system seems to be very actual.

3 Creation of Unified Phonetic Table

Traditionally in the practical transcription each pair of languages (original and target) requires its own set of rules. The linguist, who creates such rules, must know both languages (at least such job may require the collaboration of two linguists). But if we want to create the system that will provide cross-language transcription for more than 10 languages, the coordination of amendments between all specialists is a very complicated task. So, it was suggested to create the common unified phonetic table for all languages. This feature distinguishes the present system from the others. Such unified phonetic table had made it possible to reduce the amount of transcription rules without any reduction of transcription quality.

The existing systems of direct transcription from one language to another require transcription rules for each pair of languages:

	Language 1	Language 2	…	Language n
Language 1	-	+		+
Language 2	+	-		+
…			-	
Language n	+	+		-

As the result, it is necessary to develop $n \cdot (n-1)$ bases of transcription. The introduction of common unified phonetic table allows us to write for each language only 2 set of rules, which provide transcription from this language into common unified phonetic representation (PhR) and in the opposite direction. Phonetic representation is a phonetic image of the word written in terms of some phonetic table.

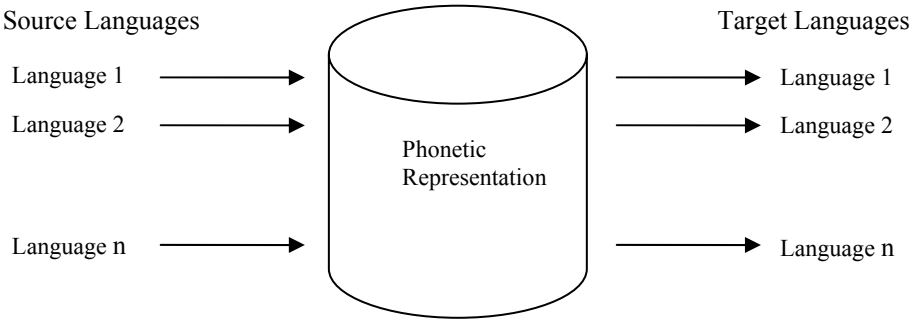

Fig. 1 Scheme of Transcription with Phonetic Representation

This scheme (Fig.1) requires only 2n bases of rules.

Transcription is fulfilled in two steps: at the first step the proper name is transcribed from the original language into the PhR in accordance with the table, which will be

described below. At the second step the PhR is transferred into the target language. The transcription is carried out with the help of the software engine which remains unchangeable for all databases of new languages and rule updates. So, the system allows to attach new languages quite easily with no changes in the engine itself.

The creation of unified phonetic table is connected with certain problems, for it is not enough to represent such table by simple merging of sounds, that belong to different languages. We must observe the condition that the table must contain all sounds of original languages, and, at the same time, each sound shall be presented by only one symbol. So, it was impossible to use the existing tables, because the same sounds are differently labeled in various phonetic systems for different languages, and, vice versa, the different labeled sounds may express one and the same phoneme. Then, each matter requires the specific decision, whether two similar but different sounds shall be labeled by one or two different symbols in PhR. One symbol in the table may possess different parameters.

Below some examples from the phonetic table are listed.

1. Sounds [n] and [ŋ] correspond to different symbols of phonetic table,
2. English 'l' and German 'l' are both labeled by one and the same symbol 'l', but they have different values of the parameter "softness/hardness";
3. French guttural 'r' and Japan (or Russian) alveolar 'r' are represented by one symbol 'r' in the table.

Before starting to develop the table, the structure of various languages had been studied in details. Basing on the phonetic form of word, it was also necessary to take spelling into account in order to preserve the graphical form of the word as far as possible. So, for instance, there was a question, whether English [θ] (recorded as 'th') and Spanish [ȼ] (recorded by letter 'c'), which are phonetically similar, shall be labeled by the same symbol. In this very case it was decided to differentiate the symbols: the Spanish [ȼ] is represented in English by letter 's' because in American dialects of Spanish language this letter is read as [s], and the graphical form of representation dominates in this situation.

It was also necessary to determine whether it is necessary to take into account the tradition of proper name transcribing, or it was better to base only on the phonetic image of the word. Many surnames and names had been transcribed a long time ago, especially it pertains to certain people that had entered the history. But at the same time the surname of the people from the same family had been transcribed variously in different historical periods. Even the name of the same person could differ significantly with the lapse of time. That is why there are many arguments to make it a rule to transcribe the names of modern namesakes of famous historical persons according to common tradition.

4 Mathematical description of machine transcription

Let us look at the problem of machine transcription from the mathematical point of view.

Here it will be considered that the letter itself, and not only the sound which is labeled by such letter, have some parameters (for instance, vowel/consonant, raw etc.). It is necessary to determine what sound corresponds to this symbol in the certain context and what set of parameters shall correspond to this sound.

Let us determine the parameter as the pair P=<N,V>, where N is the name of parameter and V is its meaning. The parameter shall represent some parameters of the letter that are urgent for transcription or that will help to classify the letters by groups. For instance: <raw, front>, <type, vowel>, <stress, unstressed>. Two parameters are considered equal, if their names and values concur.

It is necessary to define the letter in such manner, that such definition will be suitable for the further consideration. The letter consists of grapheme, which unambiguously identifies the letter and the set of parameters, that are originally inherent to such letter or reflect the position of the letter in the word. In this context we must determine each letter as the pair S=<C,{P}>, where C is a fixed symbol (grapheme), which is labeling this letter, and P is the set of parameters for the given letter. At the same time we must consider the different ways of writing (for instance, lower-case or upper-case, in the beginning, in the middle, in the end of the word or isolated), but they may have different values of certain parameters. The set of parameters is determined depending on whether it is really necessary to distinguish such writings in the process of transcription and the particular specific features of the language.

As an example we may take the pair <'A',{<"type", "vowel">, <"case", "upper case">, <"raw", "back">}>, where 'A' is a grapheme, which identifies the given letter, and the ensemble in the curly brackets is the ensemble of parameters for this letter. Here and below we will enclose in apostrophes the graphemes that correspond to the symbols of some language. The accessory graphemes intended for the provision of transcription process, will be labeled by several symbols and will not be enclosed in apostrophes.

Then it is necessary to determine the following operators for comparison of letters.

The operator "=" provides the comparison both of letters graphemes and their sets of parameters. Two letters S_1 and S_2 are equal in the terms of operator "=" ($S_1=S_2$), if their graphemes coincide and the set of parameters S_2 is a subset for the parameters of S_1.

The operator "≈" provides the comparison of sets of letter parameters. Two letters S_1 and S_2 are equal in the terms of operator "≈"($S_1 \approx S_2$) if the set of parameters of S_2 is a subset for the parameters of S_1. In other words, we may say that if $S_1=<C_1,\{P_1\}>$, $S_2=<C_2,\{P_2\}>$, $S_1 \approx S_2$ and $C_1=C_2$, then $S_1=S_2$.

The transcription will consist of two parts – from the original language to the intermediate language (intermediate phonetic representation table) and further transcription from intermediate language to the target one. The advantage of such approach is the reduction of the number of rules of transcription, which are necessary in case of a large number of languages. As it was shown above, the absence of intermediate language will require creating the rule bases for transcription from each language to every other one, that will make up $N_L \cdot (N_L - 1)$ bases, where N_L is the general number of languages. The introduction of intermediate language will reduce this number to $2 \cdot N_L$.

But this method imposes the additional requirements for the intermediate language. Its alphabet shall contain the sounds of all languages, from which the transcription is carried out. Besides the alphabet, there shall be determined a set of parameters for the letters of such language. If we want to transcribe correctly, the rules of transcription

from the intermediate language shall cover all letters of this language, and that will slightly increase the volume of rules. At the same time it increases the time of processing because there appear additional data to be processed.

Then, it is necessary to determine the alphabet of each language in order to find correspondence between all symbols one may come across in this language and the letters of this alphabet (the grapheme and the set of parameters).

The process of transcription shall be presented in five stages:

1. Transformation of the spelling of word in original language into internal representation;
2. Marking the syllables and stresses;
3. Transformation from internal representation to a PhR word;
4. Transformation of the PhR word into internal representation of the word in the target language;
5. Transformation of internal representation of the word in the target language into the written word in the target language.

The intermediary internal representation here is the term of programming, which means the format of the internal information record in the memory.

Now, each of five stages will be considered in details.

1. Transformation of Word Spelling in Original Language into Internal Representation. At this stage the word written as the set of symbols W={G}, shall be transformed into the set of letters W'={S}. Here G is some symbol (sign), and in the case of machine transcription - the informational code of sign in computer code tables (ASCII, ANSI or any other). For such transformation the set of rule is introduced. These so called alphabet rules set the correspondence between the symbol (informational code of symbol) G and the letter S. $\Re_a=\{R_a\}$, where $\Re_a$ is an ensemble of alphabet rules, and $R_a=<G,S>$ is a rule. In the machine transcription all sets of rules are kept in some bases which are called rule bases.

The example of alphabet rules could be represented by the following ensemble:

<'A',<*'A'*,***{<«type», «vowel»>, <«case», "lower">, <"raw", "back">}***>>
<'a',<*'A'*,***{<«type», «vowel»>, <«case», «lower»>, <"raw", "back">}***>>
<'B',<*'B'*,***{<«type»,»consonant»>,<«case»,»upper»>,<"sonority","vocalized">}***>>
<'b',<*'B'*,***{<«type»,»consonant»>,<«case»,»lower»>,<"sonority","vocalized">}***>>

The part pertaining to the letter (S) is marked with italic type, and bold type is for the parameters of the letter. For all graphemes of input word there shall be found such rules, that each grapheme of the input word W shall coincide with the grapheme from the rule found. The internal representation of word W' is obtained in the result of consequent concatenation of letters, which belong to the rules obtained. Besides that, the beginning and the end of each word is marked with special letters, which denote the beginning and the end of the word. All graphemes, for which the correspondence had not been found in the alphabet rules, are considered to be the punctuation marks, and then they are passing to the next stages without changes. The letter, which means the end of the word, is added each time before the beginning of the punctuation mark group. In the end of punctuation mark group the letter, which means the beginning of the word, is added. Such approach allows to separate not only punctuation marks, but

also the symbols from other alphabets, which shall not be transcribed within the frames of given alphabet.

So, $W \Rightarrow W' = \bigcup_{m=1}^{N_{W'}} S_m$, and at the same time

a) $S_1 = <BEG, \{\}>$,

b) $S_N = <END, \{\}>$, here BEG and END are the graphemes, which mean the beginning and the end of the word,

c) $S_m = S$, if $\exists (R_a = <G, S> \in \Re_a : G = G_j)$, here j=1..M, where M – is the general number of symbols in the input word, where j remains nondecreasing with the rise of m,

d) $S_m = <G_j, \{\}>$, if not $\exists R_a = <G, S> \in \Re_a : G = G_j$,

e) $S_m = <BEG, \{\}>$, if S_{m-1} is obtained by means of the rule d), and S_{m+1} is obtained by means of the rule c),

f) $S_m = <END, \{\}>$, if S_{m-1} is obtained by means of the rule c), and S_{m+1} is obtained by means of the rule d),

Here we have $m \in (1, N_{W'})$, where $N_{W'}$ is the general number of letters in the output word (in the internal format).

2. Marking the Syllables and Stresses. This operation must be carried out to determine closed and open syllables and stressed/unstressed letters. Any letter in the end of the syllable gets the additional parameter "Letter in Syllable" with the value “open”. For all other letter the value of this parameter is “closed”.

The determination of syllables is made according to the following algorithm. For the alphabet of each language we may fix the set of syllable-forming letters. The half of letters between two syllable-forming letters is taken as a part of the syllable, which is attached to the corresponding syllable-forming letter. If the number of letters is odd, the central letter is attached to the next syllable. The exception is made for the prefixes, suffixes and inflexions, for which the division into syllables is fixed. These parts of word are attached to the rest part of the word as a separate syllable or several syllables fixed with the help of special rule base.

The marking of stresses and the determination of syllables is not an obligatory operation. Such operations are required for those languages where the phoneme of the corresponding letter is changed subject to the position of letter (stressed or unstressed, in the end of the syllable or not).

For those languages where the marking of stresses is determinative, the sequential number of syllable and the direction of syllable counting (from the beginning or from the end of the word) shall be fixed. In case the word contains less syllables than the number which is obtained according to the rules, the last actual syllable is got stressed.

3. ***Transformation from Internal Representation to a PhR Word.*** This stage is necessary to unify the representation of words, belonging to different languages, to one common form of record with the help of the phonetic table alphabet. The succession of letters of original language is coming to the input of this stage. The output is the set of phonemes of the phonetic table.

The string (word) is understood here as an ordered set of letters. The substring of the word shall be determined as the subset of successive letters of this word. Let us determine the substring of the word W with the length l as W_l^i, where i is the number of position of the first letter of the word W. Here and in the following the superscript of substring shall correspond to the initial position of the substring in the word, and the subscript shall be the length of the substring. The symbol * is for the arbitrary value of the position.

Then, let us determine the Transformation Rule as the pair $R_t=<W_{l1}^*, \overline{W}_{l2}>$, where W_{l1}^* is a sample string and $\overline{W}_{l2}$ is a result string. The rule R may be applied to the substring W_{l1}^i, if the sample string is comparable to W_{l1}^i. The comparability shall be understand as the equality of letters from W_{l1}^* и W_{l1}^i strictly in the same positions of the string W_{l1}^* and the substring W_{l1}^i. At the same time here the letter S_1 is equal to the letter S_2 if $S_1=S_2$ or $S_1 \approx S_2$. The detailed algorithm of application of the rule to the string is given lower:

The transformation of the substring W_{l1}^i looks like the function $\overline{W}_{l2} = F^t(W_{l1}^i)$, for which there must $\exists R_t=<W_{l1}^*, \overline{W}_{l2}> \in \Re_t$ applicable to W_{l1}^i. Here $\Re_t=\{R_t\}$ is the transformation rule base.

The task of transformation into the intermediate phonetic representation in this case could be represented as follows:

Let some word $W=<S_1, S_2, ..., S_a>$ be entered at the input of the given stage. The set of transformation rules is $\Re_t$. The transformation of the internal representation into the intermediate phonetic written form shall be fulfilled, if we found and apply the ordered subset of such rules $\Re=<W_l^*, \overline{W}_{l2}>$, that the following conditions are observed:

1. $i=<i_1, i_2, ..., i_n>$, where n is the number of rules in the subset $\Re$;
2. $l=<l_1, l_2, ..., l_n>$;
3. $\sum_{j=1}^{n} l_j = a$;
4. $i_1=1$;
5. $i_{k+1}=i_k+l_k$ for $k<n$ and $i_n+l_n=a+1$;
6. $\forall k,m \; \exists R_t=<W_k^*, \overline{W}_{k2}> : \exists \; \overline{W}_{k2} = F^t(W_k^m)$.

Here the set i is the number of positions, from which the rules could be applied, and the set l is the set of substring lengths.

In case such set of rules does not exist, the transcription is considered to be failed. In this case it is possible to try to find such set of rules which will provide the minimal number of breaks (untranscribed letters) in the input word.

The result of the translation shall be the concatenation of the results of successive application of transformation rule.

$$\overline{W} = \bigcup_{i,l} F^t(W_l^i)$$

The check of applicability of the rule to the string is carrying up in the following manner. The rules may contain letters with special grapheme EMPTY. The comparison of letter from the rule and the letter from the string is carried out with the help of operator "≈" if the grapheme of the rule is equal to EMPTY, and if not, the letters are compared with the operator "=".

In the beginning of transformation of word internal representation into intermediate phonetic representation, the current position in the input string is set to 1. Then, until the end of word is reached, the following algorithm is applied.

The current position shall be saved. Then, it is necessary to try to find all the rules applicable for the string, which begins from the current position. If several initial sequential letters in the rule have the grapheme equal to EMPTY, we decrease the current position by the number of such letters. If the current position is getting less than 1, we consider that the rule is not applicable, restore the current position and go to the next rule.

Starting from the current position, we successively compare the letters from the string and rule. If even one letter of the string is not equal to the corresponding letter of the rule, we consider the rule as not applicable and go the next rule. If the comparison of all letters is successfully, we consider the rule as applicable. In this case we put the current position to the set *i*. The number of letters in the rule with the deduction of successive letters in the beginning and the in end of the rule with the grapheme EMPTY, is put to the set *l*. If more than one rule is applicable to the same position in the word, for each rule the sets *i* and *l* are formed on the base of the existing rules. Then the current position and number of letters are entered into these sets. Then the saved current position is restored and we pass to the next rule.

When the search for all the rules is completed, the current position is increased by the value that is saved in the set *l*.

4. *Transformation of PhR Word into Internal Representation of the Word in the Target Language.* This stage is analogous to the stage 3 but has quite the opposite tasks. Its function is to form the chain of letters which reflect the obtained phonetic sounding of the word in the target language. This work is carried out in accordance with the same principles as in the stage 3. But the rules here are not so multivalued as in the stage 3, because in the process of creation of rules $\Re_t$ there is an opportunity to set only one fixed rule for the reproduction of the given set of sounds in the case the alternative is present.

5. ***Transformation of Internal Representation of the Word in the Target Language into the Written Word in the Target Language***. This stage is opposite to the stage 1. Here the same rules as in the stage 1could be applied, as in the most cases there must exist the unambiguous correspondence between the grapheme and the set of parame-

ters. The letters with the graphemes BEG and END are to be deleted, the punctuation marks are recorded with the corresponding symbols.

The method suggested allows to formalize the machine transcription in multi-language systems. It provides the opportunity to formulate strict requirements to the intermediary language and the languages participating in the transcription and to study their specific features. The formalization of transcription makes the machine transcription easy.

5 The Software for Machine Translation "TransCriba"

The software "TransCriba" has been created on the base of the described model. This program is intended for the transcription of names from original language to target one. For the present moment the rule databases have been created for the following languages: English, German, French, Spanish, Italian, Swedish, Polish, Chinese (Pinyin and Wade-Giles systems), Japanese (Romaji system, Hepburn, and our own system which allows the presence of Latin symbols forbidden in the two traditional ones), Korean (Northern, Southern and old variant of record), Vietnamese, Arabian and Turkish. But the system developed have significant restrictions for the transcription into syllable languages, such as Japanese and Chinese, as the word must be recorded with the symbols of original language. But for the present moment there exist only approximate recommendations for the recording of arbitrary word with the fixed set of syllables, and there are no methods for obtaining such a result.

The software system had been tested for the sampling containing from 1 to 5 thousand words. The discrepancy between the manual and automatic transcription doesn't exceed 1%. At the present moment the system is being extended and tested for the other languages.

References

1. Aminieva S.M. et al.: Practical Transcription of Surname and Name Groups. Moscow (2006)
2. Bondarenko A.V., Galaktionov V.A. et al.: Automatic transcription of surnames and names with the usage of unified intermediary phonetic representation. In: Science and technical information (STI) Series 2,. No. 4, (2004) 35-39
3. Reformatsky A.A.: Introduction into Linguistics Ch.3, Phonetics. Moscow, Aspect-Press (1996)
4. Trubetskoy, N.S.: The Foundations of Phonology. Moscow (1960)
5. Knight K. and Graehl J.: Machine Transliteration. In: Computational Linguistics, 24(4), (1998)
6. Choi J. Oh, K. and Isahara H.: A Comparison of Different Machine Transliteration Models: Journal of Arti¯cial Intelligence Research 27 (2006) 119-151
7. Hsin-Hsi Chen, Sheng-Jie Huang, Yung-Wei Ding, and Shih-Chung Tsai: Proper name translation in cross-language information retrieval. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics, (1998) 232-236.