INFORMATICA, 1999, Vol. 10, No. 4, 367–376

Transcribing of the Lithuanian Text Using Formal Rules

Pijus KASPARAITIS
Department of Informatics, Faculty of Mathematics, Vilnius University
Naugarduko 24, 2006 Vilnius, Lithuania
e-mail: pkasparaitis@yahoo.com

Received: September 1999

Abstract. This paper deals with one of the components of text-to-speech synthesis of Lithuanian language namely – text transcription. Formal rules' method is used for text transcription. In this work the suitability of this method is grounded, an analysis of the form of rules to fit is made and the set of rules and interpreting algorithm is presented. Contextual information, features of stress, syllable boundaries and softness are used in the rules.

Key words: text-to-speech synthesis, text transcription, formal rules.

1. Introduction

One of recent areas of applying computer technique is text-to-speech synthesis. Text-to-speech synthesis can be divided in the following main stages:

1) dividing text into syllables;
2) stressing;
3) transcribing (transforming textual string into string of phonetic units);
4) sound generation.

This work deals with text transcription. Analysis is restricted to separate words; no syntactical analysis is done. In addition I'll suppose, that text is divided into syllables and stressed.

Before transcribing it is necessary to have a list of names of phonetic units. Phonetic units themselves can be segments cut out of natural speech of announcer or certain set of sound parameters, such as linear prediction coefficients together with length of phonetic unit, fundamental frequency and so on. There are some recent attempts to create the phonetic units' base of many languages (Dutoit *et al.*, 1996). This work is based on a set of phonetic units cut out of natural speech of announcer by prof. A. Girdenis specially for Lithuanian.

2. Short Characteristic of Lithuanian Language

There are 12 vowels and 20 consonants in Lithuanian language. Vowels can build diphthongs, vowels "a", "e", "i", "u" and consonants "l", "m", "n", "r" can build mixed diph-

thongs. Only letters belonging to one syllable can build diphthongs and mixed diphthongs. Vowels are short and long. Short vowels can be stressed and unstressed. Long vowels, diphthongs and mixed diphthongs can be unstressed, stressed with rising or stressed with falling accent (Girdenis, 1995).

So, choosing phonetic units corresponding to one or some letters we have to take into account stress position, accent, syllable boundaries ant context (neighbouring letters).

3. Methods of Transcribing

Most popular are following transcribing methods (Paulus, 1998):

1) inserting transcribed words;
2) inserting transcribed parts of words;
3) using formal rules having following form:

left context, current letter, right context $->$ phonetic unit.

The first method allows to transcribe only fixed set of words, but this method is always used for abbreviations, numbers, special characters and so on. The second method allows to transcribe unrestricted set of words. Problems can appear dividing word into parts. In addition it requires big amount (comparing with the third method) of transcribed parts of word. These both methods work well in languages where the same letters are used for different sounds, for example English. The third method is the best for Lithuanian, because there are very few situations, where the same letter is used for different sounds.

There are also other transcribing methods. For example, Ketlėrius *et al.* (1994) presents the method, how signs of stress, length, softness, syllable and phonetic unit boundaries, neighboring letters are inserted into text using high level programming language (C++ or other), and this way text string is transformed into the string of phonetic units. In this case phonetic unit names are built of the corresponding letter, its context and other features of the phonetic unit. For example, transcribed word "balsė" looks like this:

BA^AL"∼|S"^Ė^\Ė,

where

^ – end of the phonetic unit,
| – end of the syllable,
BA – sound B before A,
AL"∼ – mixed diphthong AL stressed with rising accent and having soft consonant L,
S" – soft consonant S,
Ė – unstressed vowel Ė,
\Ė – end of sound Ė with falling amplitude at the end of the word.

This method has following shortcomings: 1) programmed rules are difficult to understand; 2) if the set of phonetic units changes program has to be rewritten. Remark: the rules of stressing and dividing into syllables for Lithuanian are defined well enough, but the set of phonetic units not yet. The most comprehensive investigation was made by prof. A. Girdenis, but this is not the final and unchangeable variant of the set of phonetic units. By transcribing text I'll refer to phonetic units' set created by prof. A. Girdenis.

4. Short Characteristic of the Phonetic Units' Set

These are the main phonetic units' groups and short description how they are used to build sounds:

1) Unvoiced plosives ("c", "č", "k", "p", "t") are build of pause segment and consonant itself with beginning of vowel, hard consonant, soft consonant (marked with double quote) and consonant at the end of the word (for example, "ka", "ke", "ki", "ko", "kio", "k", "k"", "k ").

2) Voiced plosives ("b", "d", "dz", "dž", and "g") are built of voiced segment and consonant itself with beginning of vowel, hard consonant and soft consonant.

3) There is a group of nasals ("l", "m", "n") including variants with beginning of vowel and at the end of the word in stressed and unstressed syllable.

4) A group of other consonants ("ch", "f", "h", "s", "š", "v", "z", "ž") including variants with beginning of vowel, hard and soft consonant. In addition some consonants include variant at the end of word.

5) A group of mixed diphthongs, built of vowels "a", "e", "i", "io", "iu", "o", "u" and consonants "l", "m", "n", "n" before "g" or "k", "r" including unstressed, stressed with rising accent (marked with apostrophe), stressed with falling accent (marked with tilde) with hard and soft consonant (for example, "al", "a'l", "al∼", "al"", "a'l"", "al"∼").

6) A group of diphthongs including unstressed, stressed with rising accent, stressed with falling accent, in addition some of diphthongs include variants before hard and soft consonant.

7) A group of vowels including short and long stressed and unstressed variants before hard and soft consonant. Remark, long vowels are not divided into stressed with rising and falling accent. By building vowels (including vowels belonging to diphthongs and mixed diphthongs) in the beginning of syllable, beginnings of vowels with rising amplitude are added, on the other hand, in the end of syllable before vowel endings of vowels with falling amplitude are added.

8) A group of consonant "j" including variants with beginning of vowel and stressed and unstressed variant at the end of word.

9) A group of consonant "r" including variants with beginning of vowel and at the end of word. In addition by building "r" in the beginning of word special segments are added.

In total the set of phonetic units consists of 476 elements.

5. Text Transcribing Using Formal Rules

This work presents text-transcribing method using set of formal rules and computer program for interpreting these rules. In this case by changing set of phonetic units the program can be left unchanged, unlike Method 4 mentioned in Section 3. The most common form of rules is similar to given in Section 3:

left context	**[current] [letter]**	**right context**	−>	**list of phonetic units**	**actions**
	features of current letter				

where **left context** and **right context** are $1, 2, 3, \ldots$ letters to the left or to the right of current letter respectively. **Features of current letter** are stress, accent, syllable boundaries and so on. **List of phonetic units** can be empty (if the current letter doesn't correspond to any separate sound), consist of one phonetic unit (if one letter corresponds to one phonetic unit) or consist of several phonetic units (if certain letter corresponds to several phonetic units as described in Section 4 paragraphs 1, 2, 7, 9). The number of phonetic unit in the set can be used as its name, but it is more understandable to use names constructed of letter, its context and special symbols. An **Action** can be the number of letters to skip after applying this rule, pointer to the next rule to continue the search and so on.

REMARK. Examples of rules in this work are given in bold; names of phonetic units are written in double quotes using capital letters; current letter is given in square brackets; by specifying context, all letters, that can take the same place, are written in braces; only elements necessary for certain example are specified in the rules.

6. Requirements for the Set of Rules

Set of rules must meet following requirements:

1) minimum memory for one rule;
2) minimum operations to check one rule;
3) minimal number of rules;
4) maximal quick search in the set of rules;
5) it's easy to change set of rules if the set of phonetic units changes;
6) understandable rules.

Requirement to have minimal number of rules is not only necessary to have quick search, but also to create set of rules in real time, because the rules are to be created by man.

Requirement to have understandable rules is hard to formalize. One of criterions I keep to satisfy this requirement is that a phonetic unit can appear on the right side of the rule only if the current letter corresponds to this phonetic unit. For example, phonetic unit corresponding to mixed diphthong "AN" before "K" or "G" can appear only on the right of the rules with the current letter "a" or "n", but can not with the current letter "k", "g" or another.

7. Context vs Features of Current Letter

Taking into account, that boundaries of syllables depend on context and stress signs can be inserted into the text, it is possible to have rules without features of current letter. But

it is not worth to do, because dividing into syllables has to be done before stressing.

The opposite way is to increase the number of features of current letter by transforming contextual information into features of current letter. It is possible to have rules without context but with a lot of features of current letter, such as "is current letter before "a", "is current letter before "e" and so on. But in this case the rules are hard to understand. This is only the case if choosing of phonetic unit depends on very wide context with many variants, and transforming context into feature of current letter allows to reduce the number of rules. For example, the consonant "r" in the words "švirkštas" and "švirkštelis" is hard and soft respectively only because the fourth letter to the right is "a" or "e", respectively. So, it is convenient to interpret softness and hardness as one more feature of current letter. In Lithuanian consonants before "e", "ę", "ė", "i", "y", "į" are soft and before "o", "u", "ū", "ų" are hard. Consonant "j" is always soft and all other consonants before it are soft too.

8. Left and Right Context

Rules will require less memory if we use narrower context. How many letters from left and right to use it is convenient to decide by finding in the set of phonetic units the element requiring the widest context to choose it. Let denote this width with C. Now we have to choose, how many letters to the left (let denote L) and to the right (let denote R) to take. Here $L + R + 1 = C$. It is convenient to choose L and R so that the current letter corresponds to the phonetic unit indicated on the right side of the rule. In the current set of phonetic units the brightest context ($C = 5$) is used by choosing phonetic unit "CH" before "IO" in the context "chrio". So, it is possible to have the following left sides of rules: **[c] {h} {r} {i} {o}** and **{c} [h] {r} {i} {o}**. It is convenient, that rules with context shorter than C have left context narrower or equal L and right context narrower or equal R. In this case the set of fixed length rules require minimal amount of memory. In some cases it is necessary to take into account one letter to the left of the current, for example "a" after "j" become "e", so, set of rules with $L = 1$, $R = 3$ is used.

9. Features of Current Letter

There are three groups of features of current letter:

1) Softness and hardness. Feature can have two values. This feature relate not only to consonants, but also to vowels, because some vowels before soft consonants sound different than before hard ones;
2) End of syllable. It can have 3 values: a) end of syllable is one letter to the left of the current letter; b) end of syllable is the current letter; c) end of syllable is somewhere else. This feature guaranties, that end of syllable is not inside diphthong, mixed diphthong, fusion of consonants ("ch", "dz", "dž") and between vowel and standing before letter "i" or "j";

3) Stress. It can have following values: a) long, stressed with falling accent; b) long, stressed with rising accent; c) short stressed; d) standing immediately before stressed in the stressed syllable; e) not stressed and not standing immediately before stressed in the stressed syllable; f) standing in the unstressed syllable.

Features of current letter have to be calculated for each letter before applying the rules.

10. List of Phonetic Units

On the right side of rules one phonetic unit can be indicated or no one. The reason, why the possibility to indicate many phonetic units was refused, can be illustrated by following example: suppose, certain sound corresponding to one letter can be built taking one of I phonetic units corresponding to beginning of this sound and taking one of J phonetic units corresponding to ending of the sound. In total we have $I \times J$ combinations. It means, that we need $I \times J$ rules. If we use the rules with only one phonetic unit on the right side we need only $I + J$ rules. In this case sometimes we have to use some rules to transcribe one letter, while in the first case every single letter can be transcribed using one rule.

Now we can estimate the minimal number of rules. It is necessary to have as many rules as the number of phonetic units. In this case not less than 476.

Rules having no phonetic unit on the right side are used to skip letters and to make the set of rules smaller and more understandable. For example, instead of using rules:

{j} [a] {i} −> **“EI”**,
{j} [a] {u} −> **“EU”**,
{j} [a] −> **“E”**,
[i] {a} {i} −> **“EI”**,
[i] {a} {u} −> **“EU”**,
[i] {a} −> **“E”**,

it is more convenient and understandable to use:

[i] {a} −> **“ ”**,
{ij} [a] {i} −> **“EI”**,
{ij} [a] {u} −> **“EU”**,
{ij} [a] −> **“E”**.

11. Actions

Two actions are indicated in the rules: 1) how many letters to skip in the word we are transcribing after the rule is applied and 2) how many rules to skip to continue the search. If certain letter has to be transcribed into two phonetic units, the first rule indicates no skip to the other letter.

It is possible to have a set of rules without indicating the rule to continue the search (in this case the next rule is used to continue the search), but indicating the rules allows

to have quicker search and less complicated set of rules. Let examine following example, where three hypothetical rules are used to transcribe one letter into two phonetic units:

[x] {y} −> "FV11", 0;
[x] {all letters except y} −> "FV12", 0;
[x] −> "FV21", 1,

where number on the right side indicates how many letters to skip in the word we are transcribing.

Using method with indicating the rule to continue the search, rules looks like this:

[x] {y} −> "FV11", 0, 2;
[x] −> "FV12", 0, 1;
[x] −> "FV21", 1, 1,

where second number on the right side of the rule indicates how many rules to skip.

So, skipping the rules allows to use more general rules with shorter context (see second rule in both sets).

It is convenient to join the rules into groups so that a group consists of rules with the same current letter. If certain letter can be transcribed into many phonetic units, many groups are built with this current letter. The search would be quicker if on the right side the rule were indicated taking into account right context of applied rule. For example, it is a good idea after applying the rule **[a] {n} {gk} −> "AN", 2** to go to the rule with current letter "g" or "k". Disadvantage of this method is that we need to modify pointers in all rules to introduce a new rule. I chose another method, in which the pointer to the first rule in the next group is indicated. Now algorithm looks like this:

if the rule matches – apply the rule and skip to the next group of rules;
if the current letter does not match – skip to the next group of rules;
otherwise go to the next rule;
if the end of set is reached – go to the first rule.

Now it is enough to modify pointers in one group to introduce a new rule.

In Lithuanian in some cases there is an assimilation of sounds (Ambrazas *et al.*, 1996):

1. "c", "č", "s", "š", "p", "t", "k" before "dz", "dž", "z", "ž", "b", "d", "g" changes into "dz", "dž", "z", "ž", "b", "d", "g", respectively;
2. "dz", "dž", "z", "ž", "b", "d", "g" before "c", "č", "s", "š", "p", "t", "k" changes into "c", "č", "s", "š", "p", "t", "k", respectively;
3. "s", "z"before "č"changes into "š";
4. "s", "z"before "dž" changes into "ž".

Some rules are introduced to realize the sound assimilation.

12. Examples of Rules

The rules for transcribing letters "a" and "b" are presented. To make an example as short and understandable as possible, stressing and softness are not taking into account.

1. {_aąeęėiyįouūų} [a] −> "/A", 0, 1,

2. {ij} [a] {i} −> "EI", 1, 16,
3. {ij} [a] {u} −> "EU", 1, 15,
4. {ij} [a] {l} −> "EL", 2, 14,
5. {ij} [a] {m} −> "EM", 2, 13,
6. {ij} [a] {n} {gk} −> "En", 2, 12,
7. {ij} [a] {n} −> "EN", 2, 11,
8. {ij} [a] {r} −> "ER", 2, 10,
9. {ij} [a] −> "E", 0, 9,
10. [a] {i} −> "AI", 1, 8,
11. [a] {u} −> "AU", 1, 7,
12. [a] {l} −> "AL", 2, 6,
13. [a] {m} −> "AM", 2, 5,
14. [a] {n} {gk}−> "An", 2, 4,
15. [a] {n} −> "AN", 2, 3,
16. [a] {r} −> "AR", 2, 2,
17. [a] −> "A", 0, 1,

18. {ij} [a] {_ aąeęėiyįouūų} −> "E\", 1, 3,
19. [a] {_ aąeęėiyįouūų} −> "A\", 1, 2,
20. [a] −> "", 1, 1,

21. [b] {ptkcčsš} −> "_PTK", 0, 2,
22. [b] −> "_BDG", 0, 1,

23. [b] {aą} −> "BA", 1, 8,
24. [b] {eę} −> "BE", 1, 7,
25. [b] {i} {ouūų} −> "BIO", 1, 6,
26. [b] {i} {aą} −> "BE", 1, 5,
27. [b] {iyįė} −> "BI", 1, 4,
28. [b] {ouūų} −> "BO", 1, 3,
29. [b] { ptkcčsš } −> "P", 1, 2,
30. [b] −> "B", 1, 1.

The syllables' boundaries are also absent in the rules. The first rule can be applied only if the end of syllable is to the left of the current letter, rules 18–19 – to the right of the current letter. Rules 2–9 can not be applied if the end of syllable is to the left of the current letter and rules 2–8, 10–16 – to the right of the current letter.

The first rule builds separate group and is designated to insert a rising amplitude segment of sound "a" in the beginning of the syllable. The rules in the third group (18–20) are used to insert a falling amplitude segment of sound in the end of the syllable before vowel. The rules in the second group (2–17) are designated to transcribe letter "a", diphthongs and mixed diphthongs containing this letter. In the rules 2–3, 10–11 the skip to the second letter in the diphthong is indicated to check if it is the ending of syllable before vowel.

The rules in the fourth (21–22) and in the fifth (23–30) group are used to transcribe the beginning and the end of the plosive "b". Rules 21 and 29 represent assimilation of this letter before voiceless consonants.

13. Results

The set of rules allowing transcribing Lithuanian text into string of phonetic units was created. Text has to be divided into syllables and stressed. In total 740 rules were created. Rules are divided into 75 groups.

Another important result is, that an algorithm for interpreting these rules was created. This algorithm doesn't depend on the set of rules and set of phonetic units.

Shortcoming of this method: in Lithuanian there are some international words with sounds not present in Lithuanian but these sounds are denoted using Lithuanian letters, for example short "o". If this letter has an accent, the stressing algorithm can decide, if this sound is short or long, for example in the words "šõkis" and "šòkas", but if this letter is unstressed, it is impossible to make a decision and long sound "o"is used.

When we talk about reliability of the method of transcribing Lithuanian text presented in this work two aspects have to be distinguished:

1) does this method enable to transcribe Lithuanian text without errors,

2) is the set of rules built correctly.

If we want to find correspondence between letters and sounds in Lithuanian we have to know stress, syllable's boundaries and context of each letter. Context enables to calculate all other characteristics (softness, assimilation and so on). The other factors defining correspondence between letters and sounds don't exist. All necessary factors (stress, syllable's boundaries and context) are covered by the rules presented in this work. So, transcribing method itself enables to transcribe Lithuanian words without errors. Errors can only appear in the international words as mentioned above and because of errors by stressing and dividing into syllables.

The second aspect requires to make a test using all possible contexts, accents, syllable's boundaries and features of softness. Such a complicated test was not done. We made the test using a dictionary consisting of some hundreds words. This dictionary guaranties that all rules and all phonetic units are used at least once. No errors were detected in the set of rules.

In the future the main attention should be addressed to create an automatically stressing algorithm for Lithuanian.

References

Ambrazas, V., K. Garšva, A. Girdenis *et al.* (1996). *A Grammar of Modern Lithuanian*. 2nd. ed. Mokslo ir enciklopedijų leidykla, Vilnius (in Lithuanian).

Dutoit, T., V. Pagel, N. Pieret, O. Van der Vreken, F. Bataille (1996). The MBROLA project: towards a set of high-quality speech synthesizers free of use for non-commercial purposes. In: *Proc. ICSLP 96*. pp. 1393–1396.

Girdenis, A. (1995). *Theoretical Fundamentals of Phonology*. Vilniaus Universitetas, Vilnius (in Lithuanian).

Ketlėrius, A., V. Undzėnas, V. Vaitukaitis (1994). Lithuanian text-to-speech synthesizer, *Information Technologies*, Kaunas.

Paulus, E. (1998). *Speech Signal Processing: Analysis, Recognition, Synthesis*, Spektrum Akademischer Verlag, Heidelberg (in German).

P. Kasparaitis was born in 1967. In 1991 he graduated from Vilnius University (Faculty of Mathematics). In 1996 he has been admitted as a PhD student in Vilnius University. Current research interests include text-to-speech synthesis and image processing.

Lietuvių kalbos teksto transkribavimas naudojant formalias taisykles

Pijus KASPARAITIS

Šiame darbe nagrinėjama viena iš lietuvių kalbos sintezės pagal tekstą sudedamųjų dalių – teksto transkribavimas. Teksto transkribavimui taikomas formalių taisyklių metodas. Darbe pagrįstas šio metodo tinkamumas lietuvių kalbai, išanalizuota, kokį pavidalą turinčias taisykles patogiausia naudoti bei pateiktas tokių taisyklių rinkinys ir jas interpretuojantis algoritmas. Taisyklėse naudojama kontekstinė informacija, kirčiavimo, skiemenų ribų ir minkštumo požymiai.