VILNIUS UNIVERSITY

Tomas Anbinderis

MATHEMATICAL MODELLING OF SOME ASPECTS OF STRESSING A LITHUANIAN TEXT

Summary of the Doctoral Dissertation

Physical Sciences, Informatics (09P)

Vilnius, 2010

Doctoral dissertation was prepared at Vilnius University in 2005-2009.

From 1 October 2005 till 17 April 2008: Scientific Supervisor Assoc. Prof. Dr. Algirdas Bastys (Vilnius University, Physical Sciences, Informatics – 09P).
Scientific Supervisor:
Dr. Pijus Kasparaitis (Vilnius University, Physical Sciences, Informatics – 09P).

The dissertation is being defended at the Council of Scientific Field of Informatics at Vilnius University:
Chairman:
Prof. Dr. Habil. Feliksas Ivanauskas (Vilnius University, Physical Sciences, Informatics – 09P).
Members:
Prof. Dr. Romas Baronas (Vilnius University, Physical Sciences, Informatics – 09P),
Prof. Dr. Habil. Aleksas Girdenis (Vilnius University, Humanitarian Sciences, Philology – 04H),
Prof. Dr. Vytautas Kleiza (Institute of Mathematics and Informatics, Physical Sciences, Informatics – 09P),
Prof. Dr. Habil. Mifodijus Sapagovas (Institute of Mathematics and Informatics, Physical Sciences, Informatics – 09P).
Opponents:
Prof. Dr. Eduardas Bareiša (Kaunas University of Technology, Technological Sciences, Informatics Engineering – 07T),
Assoc. Prof. Dr. Rimantas Vaicekauskas (Vilnius University, Physical Sciences, Informatics – 09P).

The dissertation will be defended at the public meeting of the Council of Scientific Field of Informatics at Vilnius University Information Technology Center, at 2 p. m. on 28 June 2010. Address: Šaltinių g. 1A, LT-03214, Vilnius, Lithuania. The summary of the doctoral dissertation was distributed on 27 May 2010. A copy of doctoral dissertation is available for review at the library of Vilnius University.

VILNIAUS UNIVERSITETAS

Tomas Anbinderis

KAI KURIŲ LIETUVIŲ KALBOS TEKSTO KIRČIAVIMO ASPEKTŲ MATEMATINIS MODELIAVIMAS

Daktaro disertacijos santrauka

Fiziniai mokslai, informatika (09P)

Vilnius, 2010

Disertacija rengta 2005-2009 metais Vilniaus universitete.

Nuo 2005.10.01 iki 2008.04.17: mokslinis vadovas doc. dr. Algirdas Bastys (Vilniaus universitetas, fiziniai mokslai, informatika – 09P).

Mokslinis vadovas:

dr. Pijus Kasparaitis (Vilniaus universitetas, fiziniai mokslai, informatika – 09P).

Disertacija ginama Vilniaus universiteto Informatikos mokslo krypties taryboje:

Pirmininkas:

prof. habil. dr. Feliksas Ivanauskas (Vilniaus universitetas, fiziniai mokslai, informatika – 09P).

Nariai:

prof. dr. Romas Baronas (Vilniaus universitetas, fiziniai mokslai, informatika – 09P),

prof. habil. dr. Aleksas Girdenis (Vilniaus universitetas, humanitariniai mokslai, filologija – 04H),

prof. dr. Vytautas Kleiza (Matematikos ir informatikos institutas, fiziniai mokslai, informatika – 09P),

prof. habil. dr. Mifodijus Sapagovas (Matematikos ir informatikos institutas, fiziniai mokslai, informatika – 09P).

Oponentai:

prof. dr. Eduardas Bareiša (Kauno technologijos universitetas, technologijos mokslai, informatikos inžinerija – 07T),

doc. dr. Rimantas Vaicekauskas (Vilniaus universitetas, fiziniai mokslai, informatika – 09P).

Disertacija bus ginama viešame Informatikos mokslo krypties tarybos posėdyje 2010 m. birželio mėn. 28 d. 14 val. Vilniaus universiteto Informacinių technologijų centre.

Adresas: Šaltinių g. 1A, LT-03214, Vilnius, Lietuva.

Disertacijos santrauka išsiuntinėta 2010 m. gegužės mėn. 27 d.

Disertaciją galima peržiūrėti Vilniaus universiteto bibliotekoje.

Introduction

Problem under Investigation

The present work deals with automatic stressing of a Lithuanian text and two other goals relating to it – homographs disambiguation and a search for clitics.

Relevance of the Theme

The spoken language is one of the main means of communication between people (i.e. means of exchanging information). With the appearance of computers interest in the way of realising communication between an individual and a computer by means of the spoken language developed. Unfortunately, this task has been unfulfilled thus far.

The spoken language processing systems are often divided into the systems of speech synthesis, recognition and interpretation. The speech synthesis systems are systems, which automatically generate the human speech from any textual input. Speech synthesis can be used in telecommunications, for teaching languages, the disabled people, etc. The present work investigates one of the constituent parts of speech syntheses – automatic stressing of a text.

For the synthesised speech to sound understandable and natural it is necessary to determine the stress of words in a text. Additional problems arise when stressing the words, which are written in the same way but which are pronounced differently (homographs). Furthermore, the rhythm is characteristic of the spoken language. Seeking to preserve it some words remain unstressed (become clitics).

The Lithuanian language is a highly inflected language with a free stress; therefore, the task of automatic stressing is a non-trivial one. Solving the problems of stressing of a Lithuanian text, disambiguation of homographs and a search for clitics is important when seeking to create a synthesiser of the Lithuanian voice of a higher quality.

Objectives and Tasks of the Research

The major objective of this work is to create algorithms that automatically stress the Lithuanian words, perform disambiguation of homographs and search for clitics, and implement them in computer programs. Algorithms should in no way be inferior to the already existing algorithms (if any). Seeking to achieve this objective, the following tasks were carried out:

1) To define the place of automatic stressing, disambiguation of homographs in the total scheme of speech synthesis, their interaction with other modules, the data received and conveyed. To investigate methods that have been employed to carry out these tasks in other languages.
2) To prepare a large stressed corpus of the Lithuanian language of various genres. To create software necessary for the preparation of the corpus. This corpus will be used for experiments and to evaluate the accuracy of algorithms.
3) To propose a new algorithm of disambiguation of the Lithuanian homographs.
4) To propose a new algorithm for stressing the Lithuanian words.
5) To propose the algorithm for detecting clitics in the Lithuanian text.
6) To implement the proposed algorithms and evaluate their accuracy experimentally.

Research Methods

Theoretical investigations were carried out by means of methods, concepts and other knowledge in the sphere of linguistics, computer linguistics, pattern recognition theory, graph theory, mathematical statistics and programming.

Experimental investigations and algorithms were carried out by means of software developed specially for that purpose and written in *C++* programming language, using *Microsoft* development environment *Visual Studio 6.0*. In addition, software of the accentuation of the Lithuanian words and the morphological analysis designed by Pijus Kasparaitis was used.

Scientific Innovation of the Work

Thus far, methods based on the morphological analysis only have been exclusively applied to automatic stressing of the Lithuanian language. Such algorithms are complicated therefore it is difficult to port them into other programming languages or operational systems; they are difficult to be modified or optimised. The method proposed in the present work is based on sequences of letters, it does not require any knowledge of the language and therefore it is especially simple, fast and easily applied to other languages. Stressing rules are created automatically from a large number of stressed texts. Such methods are usually applied to the non-inflectional languages.

HMM, ID3 and methods based on the syntactical analysis have been applied to disambiguation of homographs so far. They all are based of the word context. The method proposed in the present work is based on frequencies of lexemes and morphological features, and it does not make use of contextual information at all.

Besides, the algorithm of a search for clitics of the Lithuanian language is proposed. Only general tendencies for the words to turn into clitics can be found in the linguistic works; no algorithms of automatic stressing of clitics have been investigated yet.

Defended Propositions

1) The algorithm of disambiguation of homographs of the Lithuanian language based on usage frequencies of lexemes and morphological features.
2) The algorithm of the accentuation of a Lithuanian text based on sequences of letters in words. The algorithm of the automatic creation of stressing rules from a large number of stressed texts. The algorithm of reducing the number of the stressing rules.
3) The algorithm of searching for clitics of the Lithuanian language in a text: based on the recognition of combinational forms, the statistical stressed/unstressed frequency of a word, grammar rules and stressing of the adjacent words.

Practical Application

1) The algorithms of disambiguation of homographs and search for unstressed worlds (clitics) are used in the internet-based Lithuanian language synthesiser [http://www.studijos.lt/sintezatorius, viewed on 13 April 2010].
2) The word-stressing algorithm based on decision trees is used in the Lithuanian language synthesiser of *"Etalinkas" UAB*: [http://www.etalink.lt/lietuviu-kalbos-sintezatorius, viewed on 5 January 2010].
3) The algorithm of a search for unstressed worlds (clitics) is also used in program of the accentuation of the Lithuanian language *AccentTools* (see Chapter 3).

Structure and Contents of the Thesis

The thesis consists of the introduction, six chapters, the conclusions, a list of references, two annexes and lists of terms and abbreviations. The main part is 139 pages long and includes 22 figures and 25 tables. The list of references contains 160 items. The thesis is written in Lithuanian.

1 Architecture of Speech Synthesis Systems

Modern Spoken Language Processing (SLP) systems have at least one of the below-listed subsystems:

- **Speech synthesis** or **Text-to-Speech (TTS)**.
- **Automatic Speech Recognition (ASR)**.
- **Spoken Language Understanding (SLU)**.

The present work investigates the TTS systems more extensively. The standard TTS system consists of four modules (see Fig. 1).

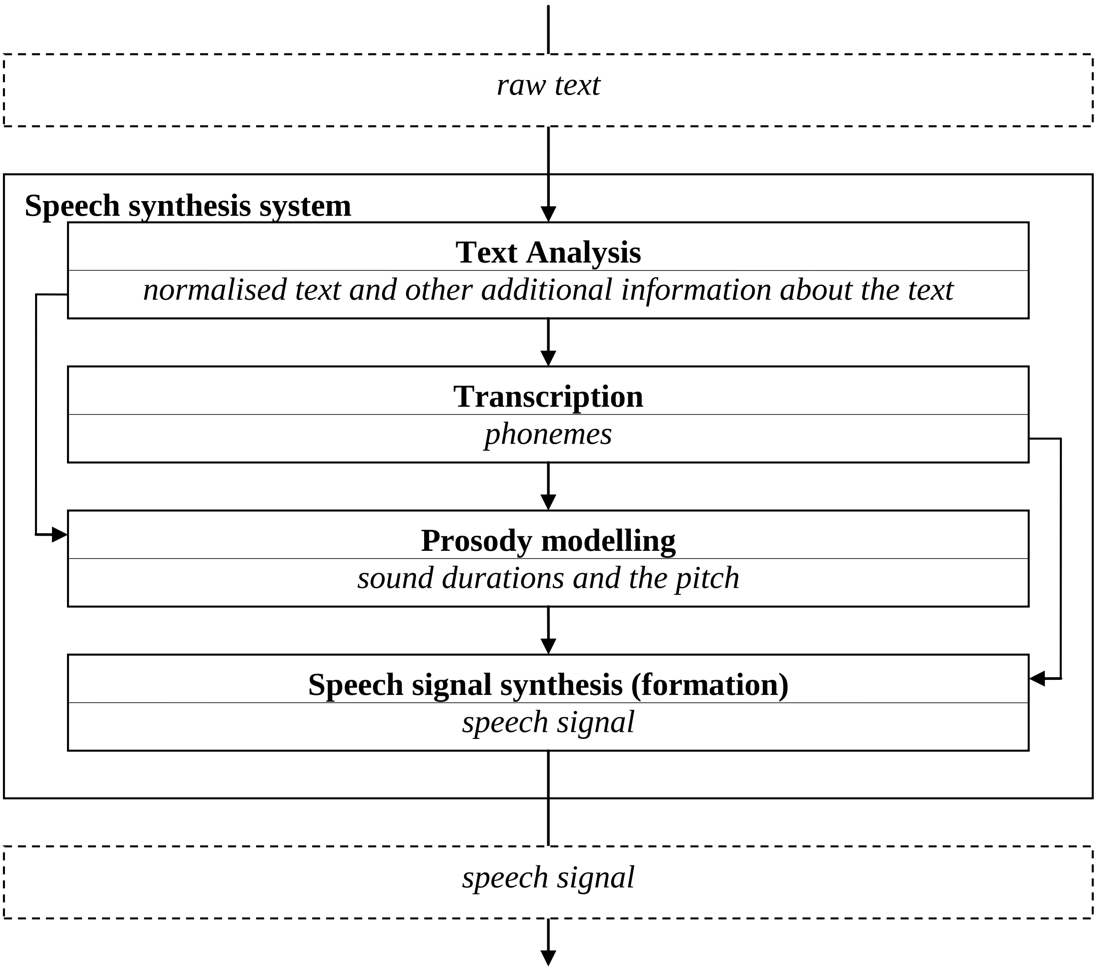

raw text
Speech synthesis system
Text Analysis
normalised text and other additional information about the text
Transcription
phonemes
Prosody modelling
sound durations and the pitch
Speech signal synthesis (formation)
speech signal
speech signal

Fig. 1. General architecture of TTS system

The Text Analysis (**TA**) module should perform the following tasks: 1) Text normalization; 2) Document Structure Analysis; 3) Morphological Analysis; 4) Syntactical Analysis; 5) Semantic Analysis; 6) **Homographs Disambiguation**; 7) **Stressing of Words**; 8) **Speech Rhythm Determination**; 9) Stressing of phrases and sentences. The functions extensively investigated in this work are emphasized.

Homographs disambiguation is made on the basis of the results of the morphological, syntactical and semantic analysis, the results are used for stressing and transcribing words.

Stressing of words is based on the morphological analysis and disambiguation of homographs, and stressing information is used for transcription, finding of phrasal stresses, prosody and rhythm modelling.

Rhythm modelling (a search for clitics) is based on the morphological analysis and word stressing, and the results are used for transcription and prosody modelling.

2 Automatic Stressing Algorithms

Text stressing depends on the language. According to the stressing paradigm, languages can have **free** stressing, or **fixed** stressing. In the case of a fixed stress, the stressing algorithm is usually defined by simple stressing rules and exceptions. In the case of a free stress, stressing methods and their complexity depend on the fact whether the language is inflectional or non-inflectional.

Words of **non-inflectional** languages (e.g., English) have few grammatical forms. Meanwhile words of **inflectional** languages (e.g., Lithuanian, Russian) have different forms depending on the gender, number, case, degree, mood, tense, person, etc., and each form of the same word can have a different stress location.

Automatic stressing methods of the languages that have free stressing are usually based on 1) vocabularies of full words or morphemes; 2) rules created by linguistic experts or generated automatically from the data.

If the language is non-inflectional, it is simply possible to build a vocabulary of all words with stresses. Methods based on the morphological word inflection rules are most often used for inflectional languages. Such methods usually have 1) vocabularies of stressed morphemes, 2) word building rules, 3) word stressing rules.

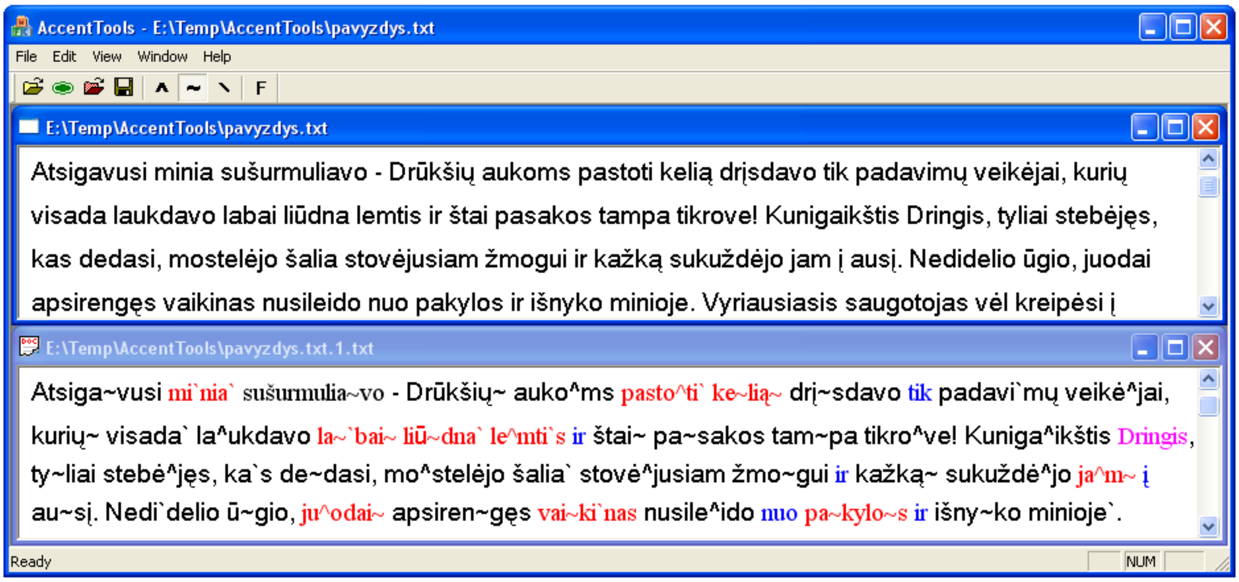
Methods based on the rules most often are based on the sequence of letters in a word, the quality and structure of syllables (closed/open), the number of syllables, information of the part of speech, etc.

The Lithuanian language: a) has free stressing, b) belongs to the group of inflectional languages, c) has additional stress elements called accents (circumflex and acute). Though the accentuation of the Lithuanian language is described in detail in linguistics textbooks, it is rather complicated to adapt these rules for a computer use.

The issue of the Lithuanian language stress approached by means of morphological rules has been dealt with in several works already [Kasparaitis, 2000], [Kasparaitis, 2001], [Kazlauskienė et al., 2004], [Norkevičius et al., 2004].

3 Stressed Texts Preparation

A significant quantity of stressed texts is needed for experiments described in Chapter 4 and 5. Automatic stressing algorithm based on morphological analysis [Kasparaitis, 2000] was used for texts stressing. This algorithm was implemented in a special program (*AccentTools*) that stresses text, marks out with a different colour unstressed words and words that can be stressed in several ways, and allows the user to choose one stressing option or correct (add) the stress mark (see Fig. 2). Using this program, a professional philologist stressed and reviewed a set of texts containing about one million words (985967 words).


AccentTools - E:\Temp\AccentTools\pavyzdys.txt
File
Edit
View
Window
Help
F
E:\Temp\AccentTools\pavyzdys.txt
E:\Temp\AccentTools\pavyzdys.txt.1.txt
Ready
NUM

Atsigavusi minia sušurmuliavo - Drūkšių aukoms pastoti kelią drįsdavo tik padavimų veikėjai, kurių visada laukdavo labai liūdna lemtis ir štai pasakos tampa tikrove! Kunigaikštis Dringis, tyliai stebėjęs, kas dedasi, mostelėjo šalia stovėjusiam žmogui ir kažką sukuždėjo jam į ausį. Nedidelio ūgio, juodai apsirengęs vaikinas nusileido nuo pakylos ir išnyko minioje. Vyriausiasis saugotojas vėl kreipėsi į

Atsiga~vusi mi`nia` sušurmulia~vo - Drūkšių~ auko^ms pasto^ti` ke~lią~ drį~sdavo tik padavi`mų veikė^jai, kurių~ visada` la^ukdavo la~`bai~ liū~dna` le^mti`s ir štai~ pa~sakos tam~pa tikro^ve! Kuniga^ikštis Dringis, ty~liai stebė^jęs, ka`s de~dasi, mo^stelėjo šalia` stovė^jusiam žmo~gui ir kažką~ sukuždė^jo ja^m~ į au~sį. Nedi`delio ū~gio, ju^odai~ apsiren~gęs vai~ki`nas nusile^ido nuo pa~kylo~s ir išny~ko minioje`.

Fig. 2. The main window of the program *AccentTools*.

The red colour marks the words to which several stressing hypotheses were returned, the blue colour marks the clitics, the violet colour denotes the unrecognised words and the black colour signifies correctly stressed words.

4 Disambiguation of Lithuanian Homographs

4.1 Concept of the Homograph

If methods based on the morphological analysis are used for word stressing, it is convenient to divide stressing of a word into the following three steps: finding the title form (lemma), finding a grammatical description (gender, number, case, etc.) and determining the stress location and the accent on the basis of the lemma and the grammatical description. A large number of words in the Lithuanian can have several lemmas and several grammatical descriptions. E.g., the Lithuanian word *galvos* can mean the following: a) the genitive case of the singular noun *galva* (head); b) the plural nominative case; c) the third person future tense of the verb *galvoti*. Such words are called homoforms. Homoforms that are pronounced in a different way are called **homographs**.

This chapter is concerned with disambiguation of homographs (selection of one variant of stressing). Disambiguation of Lithuanian homoforms using HMM, ID3 algorithm, and the syntactical analysis have been considered in [Rimkutė, Grybinaitė, 2004], [Rimkutė, Grigonytė, 2006].

4.2 Selection and Preparation of Data

A somewhat modified stressing algorithm described in Chapters 2 and 3 of the thesis was used in experiments described in the present Chapter. The stressing algorithm for each word generates all possible hypotheses about what grammatical form of what word this could be and how it is stressed. For short, hereinafter they will simply be referred to as **hypotheses.** When comparing stressing of the hypotheses generated by the stressing algorithm with the stressed text (described in Chapter 3), we find both hypotheses whose stressing coincides with that of the stressed text (referred to as a **correct hypothesis**) and whose stressing is at variance with it (referred to as an **incorrect hypotheses**). For example, having come across the word *galvõs* in the stressed text, a and c will be correct hypotheses (see the first paragraph of section 4.1), and b shall be an incorrect hypothesis.

Since one word can correspond to two, three or more grammatical forms (hypotheses), only those pairs of grammatical forms (hypotheses) were considered where one hypothesis is correct and the other is incorrect. For example, there are two pairs of such hypotheses for the word *galvos* (a-b and b-c). Using the stressed text for each pair of hypotheses we can count how many times the first hypothesis was correct and how many times the second hypothesis was correct. We shall call the hypothesis, which was correct more times, a **more frequent hypothesis**, and the other hypothesis of the pair shall be called a **rare hypothesis.** The more frequent hypothesis is written first. When disambiguating homographs pairs of hypotheses are simply taken and rare hypotheses are rejected.

The stressing algorithm uses three vocabularies: nouns-adjectives (hereinafter referred to as NA), verbs (hereinafter referred to as Vbs), that of the uninflected words (hereinafter referred to as Uninf). Entries of these vocabularies are called **lexemes**.

Each hypothesis contains the following information: 1) a vocabulary (NA, Vbs, and Uninf); 2) a lexeme; 3) a grammatical form. Thus, it can be regarded that the stressing algorithm fills the table of pairs of hypotheses. For example, if all available stressed texts consist of only two words *Mamà galvõs*, the result of work of the stressing algorithm will look like as it is represented in Table 1. Logical meanings in columns Is_corr1 and Is_corr2 show the correct and incorrect hypothesis, respectively.

Table 1. Table of pairs of hypotheses

Voc.1	Gr_f_11	Gr_f_12	Lexeme1	Is_corr1	Voc.2	Gr_f_21	Gr_f_22	Lexeme2	Is_corr2
NA	sng. Nom.	3 declens.	Mama	TRUE	NA	sng. Voc.	3 declens.	mama	FALSE
NA	sng. Inst.	3 declens.	Mama	TRUE	NA	sng. Voc.	3 declens.	mama	FALSE
NA	sng. Gen.	3 declens.	Galva	TRUE	NA	plr. Nom.	3 declens.	galva	FALSE
Vbs	future	–	Galvoti	TRUE	NA	plr. Nom	3 declens.	galva	FALSE

Experiments have been carried out at different cross-sections when grouping the data that are analogous to those presented in Table 1.

4.3 Rejections of Lexemes

Existence of some lexemes in a vocabulary sooner interferes with stressing than helps it. Having thrown them away from the vocabulary, it would be possible to stress more words unambiguously. Two ways of throwing away lexemes were considered:

1) Pairs of lexemes of the noun-adjective vocabulary are found whose stems and declinations coincide. This can be written down in the following query:

```
SELECT Lexeme1, count(Is_corr1 = TRUE),
Lexeme2, count(Is_corr2 = TRUE),
GROUP BY Lexeme1, Lexeme2,
WHERE (Vocabul1 = „NA“) && (Vocabul2 = „NA“)
&& (Gr_f_11 = Gr_f_21) && (Gr_f_12 = Gr_f_22).
```

We shall call a **more frequent lexeme** the lexeme from which correct hypotheses are more often generated. Rare lexemes are rejected. In the below presented examples we shall write the lemma in place of the lexeme and shall present the number of repetitions next to the lemma, and for the sake of shortness we shall separate the more frequent lexeme from the rare one by the sign >. For example, *kláusimas* (699) > *klausìmas* (0), *Jõnas* (172) > *jònas* (17).

2) The number of correct (incorrect) hypotheses is simply counted for each lexeme (and analogically rare lexemes are rejected). This is defined by the following query:

```
SELECT Lexeme1, count(Is_corr1 = TRUE),
Lexeme2, count(Is_corr2 = TRUE),
GROUP BY Lexeme1, Lexeme2.
```

The first method enables only a small number of lexemes to be rejected; however, it guarantees that the number of unstressed or incorrectly stressed words will not increase. The second method allows more lexemes to be rejected; however, it can also have side effects. Having applied both methods, the share of correct hypotheses among all the hypotheses has increased by as much as 6.1%.

4.4 Rules Based on Frequencies of Grammatical Forms

This Chapter deals with grouping of the data presented in Table 1 according to the grammatical forms; the query of the following type is used for this purpose:

```
SELECT Vocabul1, Gr_f_11, Gr_f_12, count(Is_corr1 = TRUE),
Vocabul2, Gr_f_21, Gr_f_22, count(Is_corr2 = TRUE),
GROUP BY Vocabul1, Gr_f_11, Gr_f_12, Vocabul2, Gr_f_21, Gr_f_22.
```

We obtain a new table of pairs of hypotheses (more frequent and rare ones) whose entries can be treated as certain rules of selecting one variant of stressing on the basis of frequencies of morphological features, which have the below-presented form:

```
(Vocabul1, Gr_f_11, Gr_f_12) > (Vocabul2, Gr_f_21, Gr_f_22).
```

The rules obtained are divided into four groups (A, B, C and D) adding additional corresponding conditions to the above-mentioned query:

```
(A) WHERE ((Vocabul1 = „Nek‟) || (Vocabul2 = „Nek‟).
(B) WHERE ((Vocabul1 = Vocabul2) && (Lexeme1 = Lexeme2)).
(C) WHERE (Vocabul1 <> Vocabul2).
(D) All remaining rules.
```

Having done several reductions of the rules, 1155 rules were obtained. Furthermore, it was investigated how the accuracy of stressing changes when refusing the most unreliable rules. When all the rules are used, the accuracy is as much as 84.3% for training data. If as much as 40% of the rules are left, the accuracy will decrease by as little as 0.9%, and having left only 7% of the rules, the accuracy will decrease by 10%.

4.5 Comparison of the Results

The accuracy of disambiguation of homographs is compared with the results presented in the work [Rimkutė, Grybinaitė, 2004]. It is true, experiments were carried out with other data in that work and all rather than only differently stressed grammatical forms were disambiguated. The ID3 algorithm, which is based on the morphological features of the adjacent words, was used for disambiguation. The comparison of the results is presented in Table 2. The results are rather similar; perhaps they are only insignificantly worse.

Table 2. Comparison of the ID3 algorithm and the frequency-based rules

Pair of coinciding grammatical forms	Accuracy when applying ID3 algorithm	Accuracy when applying frequency-based rules
Singular possessive case of the feminine gender and the plural nominative case	73.65%	77.47%
Singular nominative and instrumental case of the feminine gender	81.65%	74.73%
The infinitive and the plural nominative case of the passive past non-pronominal mascular participles	92.15%	91.98%

4.6 Experiments of Text Stressing

The greater the number of correct hypotheses of a certain rule (in per cent) in the training data the more accurate stressing of the testing text can be expected. Dependence of the percentage of correct hypotheses on the text stressing accuracy is not direct. It is possible to find out how the rules interact and what accuracy of stressing they actually produce only when using them to stress the text. Experiments with three different sets of rules, 3 different vocabularies of lexemes and with/without a random choice of one stressing variant were carried out (18 experiments in total).

Having the aim to select one variant for all the words that have many variants of stressing, the best results were achieved by means of the sequence of algorithms presented in Table 3.

Table 3. The sequence of disambiguation algorithms that produced the best results

Algorithm	Number of disambiguated words	Correct
Lexeme rejection algorithm 1	680	91.18%
Rule groups A, B, C and D	29138	85.09%
Random selection of variant	379	67.81%
Total	30197	85.01%

Thus, in total testing texts contained 30197 homographs (15.30% of all the words); the correct variant was established within the accuracy of 85.01%.

5 Automatic Stressing of Lithuanian Text Using Decision Trees

5.1 The Main Idea of the Proposed Method

Methods based on stressed words lists are usually used for free stressed non-inflectional languages. Free stressed inflectional languages usually use morphology-based methods. So far, exclusively such methods were used for Lithuanian language. However, morphological rules for Lithuanian language are rather complicated.

In this chapter, the method that does not use any information about word forming morphemes, inflection, part of speech tags, boundaries of syllables etc. was applied for free stressed inflectional Lithuanian language. The proposed method uses a decision tree to find the sequence of letters, which unambiguously defines the word stressing. In the decision tree method the stressing rules are created automatically provided a sufficient amount of stressed texts is available. The stressing algorithm is extremely simple, fast; it can be easily adapted to other world languages, and easily ported to other programming languages and operating systems.

5.2 Lists of Words

Two lists of words were used: **The list of unstressed words** is mainly made of clitics (words that tend to be unstressed), foreign words and abbreviations. Then, the list of unstressed words (clitics) is always used before applying stressing rules.

The list of stressed words consists of the stressed words of the text. If the same words have different stressing (homographs), the stressing variant, which statistically occurs more frequently, is included in the list. The list of stressed words is further used to make decision trees and stressing rules.

5.3 Algorithm of Word Beginnings and Endings

Methods for drawing up stressing rules presented in this chapter use classification (or decision) trees. Decision trees are used to forecast the variable y value, which corresponds to the parameter vector $\mathbf{f}$. In this chapter, the parameter vector $\mathbf{f}$ corresponds to the sequence of letters in a word, and the variable y corresponds to the index of stressed letter and the accent type. The essence of the method is to single out such letter sequences that define a unique word stressing. Three different methods were tested: letter sequences at the beginning, the end of a word and in any part of the word.

Let us first consider making of the tree taking letters from the beginning of a word (left to right). The tree nodes store the possible stressing and the edges store letters (Fig. 3). When adding a word to the tree, all nodes that are on the path are complemented with the same information about stressing. After all words from the list are added to the tree, the root node stores all possible stressing variations. Below the algorithm is given in pseudo-code:

```
For each word from the list of stressed words
         The root becomes the current node
         For each letter starting with the beginning of the word
                  Add a letter to the tree
                  Change the current node and supplement it with
                  information about stressing
```

If one word is part of another word, only stressing rule corresponding with a longer word will be made. To solve this problem special word ending symbols (“#”) were added.

The nodes with a unique stressing, and where the index of the stressed letter is not greater than the level of the current node are referred to as **decision nodes**. The stressing rule is formulated by collecting all letters starting from the root to the decision node into a sequence. Fig. 3 and Table 4 show an example of a tree and rules created from five words. A textual representation of the rule (“ORK”), the index of the stressed letter (1) and the accent (‘/’) are combined into single form (“ÓRK”).

Table 4. List of words and rules of word beginnings formed from them

Input words	Beginning rule formed
OKEÃNAS# OKEÃNO# ORAĨ# ÓRKAITĖ# ÓRLAIDĖ#	OKEÃ ORAĨ ÓRK ÓRL

To stress a word with the word beginning rules all that is necessary to do is to find the rule that corresponds with the beginning of the word. If a suitable rule is not found the word is left unstressed. The tree of word endings is formed in the same way as the tree of word beginnings; only each word shall be reversed.

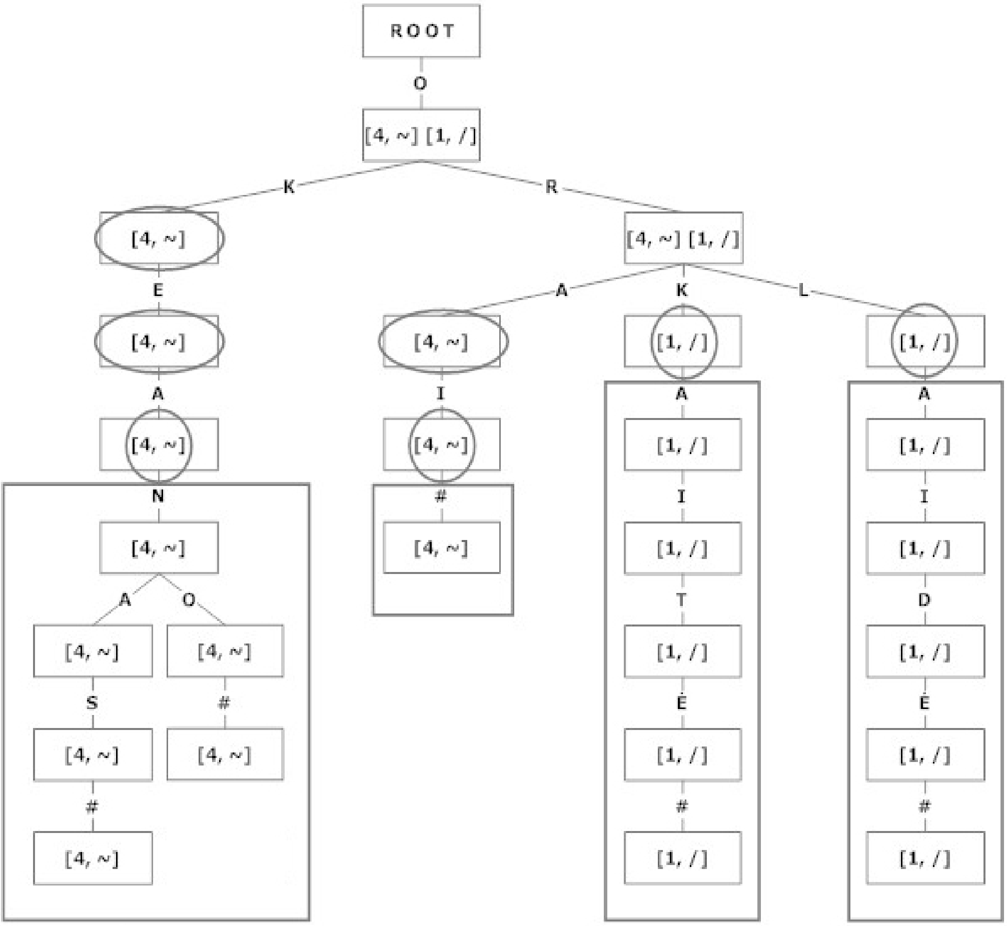


Fig. 3. Decision tree of word beginnings. Circles indicate the decision nodes; rectangles show children of the decision nodes, which are not checked; ovals denote those nodes, which cannot be decision nodes because the stressed letter has not been reached

5.4 Algorithm of the Word Middle Rules

Now we shall consider letter sequences, which can be anywhere in a word: at the beginning, in the middle and at the end. For the sake of simplicity, let us call them the word middle rules. The algorithm is similar to that of the word beginning; only each word is added to the tree several times cutting one letter from the word beginning. To stress the word, it is necessary to search for any part of the word that matches the rule. This slows down the search; therefore, the issue of decreasing the number of rules becomes important.

The list of the word middle rules contains the rules that coincide with the ending of another rule, for example: ORAĨ, **RAĨ**. The longer rules are discarded (**first reduction**).

The main idea of another rule rejection algorithm (**second reduction**) is as follows: all words from which the rules were created are taken, and these rules are applied as long as all words become stressed. After the rules have been found for all words, the remaining rules can be deleted from the list. The rules are applied starting with those that suit the maximum number of words.

5.5 Experimental Results

The available stressed texts (see Chapter 3) were divided into five roughly equal parts containing 200000 words each. Data sets containing 200000, 400000, 600000 and 800000 words were used for training (creation of rules). Testing was conducted with all the words that were not used for training. The average error and the average number of rules were calculated for each training data quantity.

Experiments, using seven methods for creating stressing rules, were performed: 1. stressed words (**wrd**); 2. word beginning rules, then word ending rules (**bgn-end**); 3. word ending rules, then word beginning rules (**end-bgn**); 4. word beginning rules (**bgn**); 5. word ending rules (**end**); 6. word middle rules (**mid**) or **mid** after the first reduction (**mid1**); 7. **mid** after the second reduction (**mid2**). Averages of the text stressing accuracy are presented in Table 5. Here 800000 words for training and 200000 for testing were used.

Table 5. Averages of the text stressing accuracy for different methods. Columns: A – clitics stressed (erroneous); B – clitics unstressed (correct); C – words unstressed (erroneous); D – unknown words unstressed (correct); E – unknown words stressed (erroneous); F – a wrong stress mark or stress place (erroneous); G – correct stressing; H – total errors (A+C+E+F); I – total correct (B+D+G)

Method	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
1 wrd	0.19	15.82	8.81	0.69	0.10	1.10	73.28	10.21	89.79
2 bgn-end	0.19	15.82	1.54	0.46	0.33	2.53	79.13	4.59	95.41
3 end-bgn	0.19	15.82	1.54	0.46	0.33	2.41	79.25	**4.47**	**95.53**
4 bgn	0.19	15.82	3.51	0.57	0.22	2.15	77.53	6.08	93.92
5 end	0.19	15.82	3.64	0.55	0.24	2.00	77.56	6.07	93.93
6 mid,mid1	0.19	15.82	1.04	0.35	0.44	2.99	79.17	4.66	95.34
7 mid2	0.19	15.82	1.93	0.44	0.35	2.29	78.98	4.76	95.24

Method 3 (**end-bgn**) gives the best result – 4.47% of error. The word middle method after the second reduction (**mid2**) requires the minimum number of rules. The method that is best in respect of the error (**end-bgn**) requires about three times more rules than **mid2**.

5.6 Comparison of Results with the Morphology-based Method

In Table 6 the best method (**end-bgn**) is compared with the method proposed in [Kasparaitis, 2000], [Kasparaitis, 2001] that is based on morphological rules. Both methods were tested with only one set containing 800000 words, so the average was not calculated. Though the results of the method proposed are slightly worse than those of the morphological approach (about 0.8%), the proposed method is much simpler with respect to both the creation and application of the rules.

Table 6. The best method (**end-bgn**) as compared with a morphological approach (**morpholog**) testing with only one set containing 200000 words. Columns: see Table 5

Method	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
3 end-bgn	0.17	15.80	1.32	0.51	0.36	2.37	79.47	**4.22**	**95.78**
morpholog	0.07	13.49	1.54	3.07	0.11	1.67	80.05	**3.40**	**96.61**

On the basis of the error values obtained using 200000, 400000, 600000 and 800000 words for training the most accurate method (**end-bgn**), the attempt was made to forecast (extrapolate) an error for a greater number of training words. The method of the least squares was used for extrapolation. Results are presented in Fig. 4. The number of errors, similar to that achieved by means of morphological method (3.40%) would be achieved when training the method (**end-bgn**) with 1500000 words, whereas having trained this method with 2000000 words the forecasted error accounts for 2.96%.

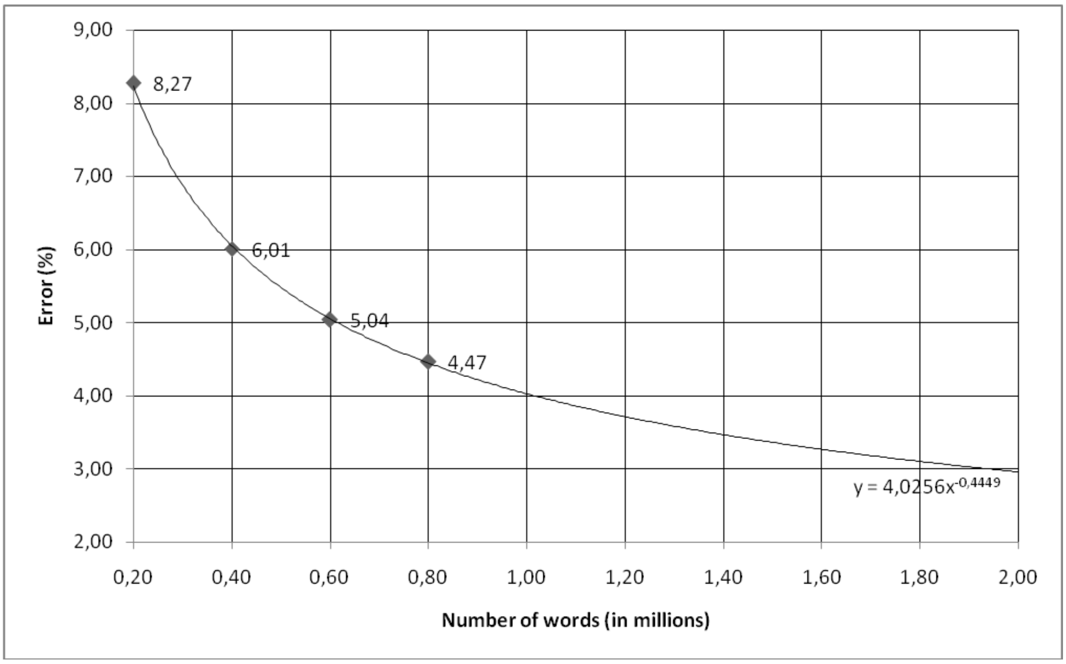


Fig. 4. Forecast of an influence of the training data size on the stressing accuracy

6 Algorithms for Detecting Clitics in the Lithuanian Text

6.1 Concept of the Clitic

The rhythm (alternation of stressed and unstressed syllables) is inherent feature of the spoken language. Seeking to preserve the rhythm some words remain unstressed. The unstressed words join the stressed ones as if becoming syllables of the latter. Unstressed words in a sentence are called **clitics**.

So far algorithms for stressing Lithuanian words stressed all words, however about 20% of words in text should be unstressed. In linguistic papers [Mauricaitė, 1985], [Stundžia, 1991] only common tendencies of clisis can be found. Algorithms of a search for clitics in a text of the Lithuanian language are a theme that has not been investigated yet. Moreover, it is quite a complicated problem because the same word can be stressed in one context, whereas in another context it can be a clitic.

Whether the word is a clitic depends on various factors: the number of syllables in a given word, its functional weight in the text, a distance between the stressed syllables, the number of unstressed syllables at the beginning and at the end of the phrase, the length of the phrase, and a logical stress.

6.2 Clitics in Combinational Forms

Certain parts of speech (pronouns, adverbs, conjunctions, particles) that are made up of several words and are called combinational forms, e. g. *bet kas* (anybody), *kaip nors* (somehow). In the majority of combinational forms, one word is not stressed but it joins another word in the combinational form [Mauricaitė, 1987].

The list of combinational forms was made on the basis of dictionary of Lithuanian. Its 95 combinational forms belong to the non-inflectional parts of speech (adverbs, conjunctions, and particles) and 35 belong to the inflective parts of speech (pronouns). The total list of combinational forms consisted of 699 word groups.

The majority of words that form combinational forms can be used separately; therefore, it is not always easy to detect combinational forms. It was established based on statistical data received from the Lithuanian language corpus and the Internet that a group of words capable of making up a combinational form usually does make it up, with a few exceptions.

Let us assume that we have a word group consisting of three following words: *bet kas nors*. A combinational form can be made up of both the words *bet kas* and the words *kas nors*. In order to make clear how to solve such disintegration problems, the list of such word groups was drawn up, an analogous statistical analysis was performed and the rules were made.

6.3 Search for Clitics on the Basis of Statistical Stressed/Unstressed Frequency of Word

In further experiments texts with the stress marked were used, clitics were left unstressed in these texts. In the following experiments the part of texts made of 8397 words was used, with 1842 (21.9%) of them being unstressed.

An attempt was made to find out how successfully unstressed words could be detected on the basis of the texts to statistically determine that the given word is more often unstressed than stressed. A list of such words was made. The texts were divided into two approximately equal parts that were alternately used for making a list and for testing. Two types of mistakes can be made in the experiments: 1) the word that should be stressed is left unstressed; 2) the word that should be left unstressed is stressed. Attempts will be made in all following experiments to minimize the sum of mistakes of both types. An accuracy of 12.7% is obtained when the same data for making the list and for testing are used, whereas when using different data the accuracy accounts for 19.2%.

6.4 Finding Clitics on the Basis of Grammar

An attempt was made to formulate the rules on the basis of the Lithuanian language grammar so that the rules could be applied to the entire group of words and the results obtained would be compared with the results we obtained by applying a statistical method to each word separately.

Whether the word is stressed depends largely on what part of speech it is. The statistical method can be successfully applied to some non-inflectional words (prepositions, conjunctions, and particles). Those non-inflectional words were picked out from the list of unstressed words described in Section 6.3, which were more than twice unstressed than stressed. The list of 41 words was received. In the texts given, the words from the above-mentioned list were encountered as many as 1354 times, and 16 times these words were stressed in the texts. In this way, the above presented list covers 72.6% of unstressed words and the mistakes account for as little as 1.2%.

However, there are words, which being one part of speech tend to be unstressed, whereas being another part of speech they tend to be stressed. Seeking to recognize such parts of speech words were grouped into sets and the special rules were created based on punctuation marks, grammatical form of neighbouring words, stressing of neighbouring words and so on. E.g., the following set and the rule was build:

The set of demonstrative pronouns DEM_PRON = {*tas*, *to*, *tam*, *tą*, *tuo*, *tame*, *tie*, *tų*, *tiems*, *tuos*, *tais*, *tuose*, *ta*, *tos*, *toje*, *toms*, *tomis*, *tose*, *šis*, *šio*, *šiam*, *šį*, *šiuo*, *šiame*, *šie*, *šių*, *šiems*, *šiuos*, *šiais*, *šiuose*, *ši*, *šios*, *šiai*, *šią*, *šia*, *šioje*, *šioms*, *šias*, *šiomis*, *šiose*}.

Rule 3. Do not stress the word of the set DEM_PRON, if it is followed by the word in the same case, otherwise stress it.

Stressing of some words depends solely on whether the adjacent words are stressed or not. The following experiments were carried out: a certain set of words was chosen for which one rule was going to be drawn up. Statistical data about how many times the words of this group were stressed/unstressed in a certain context are collected. Statistics was recorded in a Table. On the basis of Table the rule was drawn up, according to which the words belonging to the set for each context pair “following” and “preceding” had or did not have to be stressed.

Six groups of words or separate words were analyzed. E. g., statistics for the set BUTI = {*buvo*, *bus*, *buvau*, *buvai*, *yra*, *esu*, *esi*, *nesu*, *nesi*, *nėra*} see in Table 7. Here you can also see the contexts analyzed.

Table 7. Dependency of stressing the forms of the verb *būti* on the context stress

preceding following	the punctuation mark	the unstressed word	the stressed syllable	one unstressed syllable	2 unstressed syllable	3 or more unstressed syllables
the punctuation mark	*1/0*	*1/0*	*3/0*	*3/1*	–	–
the unstressed word	*5/0*	*5/0*	*3/1*	*5/0*	*1/0*	*2/0*
the stressed syllable	*6/0*	*3/1*	**3/5**	**2/3**	**2/3**	–
one unstressed syllable	*4/0*	*1/0*	**3/8**	**5/6**	**1/2**	*1/0*
2 or more unstressed syllables	*1/0*	–	*2/0*	*1/0*	–	–

Rule 5. Do not stress the words of the set BUTI following the stressed syllable or following one unstressed syllable in the stressed word preceding the stressed syllable, 1 or 2 unstressed syllables, otherwise stress them.

Nine rules were built in total. The results for all rules are presented in Table 8.

Table 8. Comparison of the rules formulated and the statistical method

Rule No.	A set of words (*examples*)	Stressed words + unstressed words = total, (**ratio of words to all words)**	Mistakes of type 1+ type 2 = total *(mistakes and words ratio)*	
			Rule applied to a group of words	Statistics applied to separate words
1	O&NE (*o, ne, nebe*)	2+86=88 **(11.3%)**	0+0=0 *(0.0%)*	2+0=2 *(2.3%)*
2	PREP_G (*link*), PREP_A (*prieš*)	2+6=8 **(1.0%)**	1+0=1 *(12.5%)*	0+2=2 *(25.0%)*
3	DEM_PRON (*tas, šis*)	35+56=91 **(11.7%)**	9+5=14 *(15.4%)*	19+3=22 *(24.2%)*
4	PERS_PRON (*aš, tu, jis*)	84+152=236 **(30.3%)**	23+23=46 *(19.5%)*	68+1=69 *(29.2%)*
5	BUTI (*yra, esu*)	64+30=94 **(12.1%)**	16+3=19 *(20.2%)*	0+30=30 *(31.9%)*
6	INT_PRON (*koks, kuris*)	19+20=39 **(5.0%)**	2+7=9 *(23.1%)*	3+3=6 *(15.4%)*
7	*čia*	10+23=33 **(4.2%)**	2+2=4 *(12.1%)*	10+0=10 *(30.3%)*
8	*vis*	6+13=19 **(2.4%)**	3+0=3 *(15.8%)*	6+0=6 *(31.6%)*
9	IP&A (*kas, kaip*)	65+105=170 **(21.9%)**	13+13=26 *(15.3%)*	16+26=42 *(24.7%)*
Total		287+491=778 **(100%)**	69+53=122 *(15.7%)*	124+65=189 *(24.3%)*

Hence, when applying special rules to word groups 15.7% of the mistakes was made, which is by 8.6% less than in the cases where the stressed/unstressed frequency calculated for each word separately was taken as the basis.

In formulating the rules described above, we assumed the context stressing of the word to be already known. Actually several words can go together, stressed according to the stress rules described herein. Thus, the interaction of the rules was analyzed.

6.5 Common Algorithm, Testing Results, Improvement

The common algorithm for detecting clitics can be described in the following steps:

1. Stress the words on the basis of the list of combinational forms.
2. Find all the unstressed words on the basis of the list of the unstressed words. Stress the words according to Rules 1 – 3.
3. Mark the words, to which Rules 4 – 9 will be applied, stress all the remaining words.
4. Apply Rules 4 – 9.

All the above-specified rules and their interaction were implemented in the form of a computer program. Testing was carried out with the texts used earlier for drawing up the rules (more than 8000 words), as well as the texts (almost 1000 words) that have not been used thus far. The results of testing using both texts are shown in Table 9.

Table 9. Testing results

Method of detecting clitics	The same texts for drawing up and testing the rules		Different texts for drawing up and testing the rules	
	Stressed words + unstressed words = total, (**ratio of words to all words)**	Mistakes of type 1 + type 2 = total *(mistakes and words ratio)*	Stressed words + unstressed words = total, (**ratio of words to all words)**	Mistakes of type 1 + type 2 = total *(mistakes and words ratio)*
A total of words used in rules	287+491=778 **(9.3%)**	72+56=128 *(16.5%)*	34+64=98 **(10.2%)**	15+11=26 *(26.5%)*
Combinational forms	49+49=98 **(1.2%)**	1+1=2 *(2.0%)*	9+9=18 **(1.9%)**	0+0=0 *(0.0%)*
List of unstressed words	14+1252=1266 **(15.1%)**	14+0=14 *(1.1%)*	3+124=127 **(13.3%)**	3+0=3 *(2.4%)*
Other words	6205+50=6255 **(74.5%)**	0+50=50 *(0.8%)*	702+12=714 **(74.6%)**	0+12=12 *(1.7%)*
Total	6555+1842=8397 **(100%)**	87+107=194 *(2.3%)*	745+207=957 **(100%)**	18+23=41 *(4.3%)*

In carrying out the testing with the data on the basis of which the rules were formulated, 2.3% of mistakes were made, and the ratio of the mistakes to the unstressed words was 10.5%. Having carried out the testing with the data that have not been used before, 4.3% of mistakes were made, and the ratio of the mistakes to the unstressed words was 19.8%. The unstressed words most often are detected on the basis of the list of the unstressed words.

The algorithm created was improved by supplementing the sets of words with non-inflectional form-words that were not found in texts. The number of mistakes for the testing data decreased to 4.1% and the ratio of the mistakes to the unstressed words now is 18.8%.

Results and Conclusions

The present work is concerned with automatic stressing of a text of the Lithuanian language and other two tasks related to that – disambiguation of homographs and a search for clitics.

1) The work defines the role of automatic stressing, disambiguation of homographs and searches for clitics within a general speech synthesis scheme, their interaction with other modules, the data obtained and transferred. Methods applied to solving these tasks in other languages, their selection depending on the degree of inflection of the language and the stressing paradigm are considered.
2) Having applied the methods based on morphological rules to stress a text of the Lithuanian language, some of the words (homographs) can be stressed in several ways. To increase the number of stressed words it is necessary to have the disambiguation algorithm. The homograph disambiguation algorithm proposed by the present author is based on frequencies of lexemes and morphological features. Existence of some lexemes in the vocabulary sooner interferes with stressing than helps to stress words. Having rejected them, the share of correct hypotheses among all the hypotheses (for the training data) increases by as much as 6.1%. A set of rules based on frequencies of morphological features (1215 rules), which enables to select the correct variant of stressing within the accuracy of 84.3% for training data, has been made up. The methods proposed allow disambiguating the homographs with the accuracy of 85.01% for testing data. Though the algorithm proposed does not use any information about the context, the results obtained are close to algorithm ID3 that uses contextual information.

3) Methods for stressing the Lithuanian text that are based on morphological rules are noted for their complexity; therefore, it is problematic to port them into other systems or programming languages. The methods proposed by the author in the present work are based on sequences of letters only and do not require any knowledge of the language: morphemes, parts of speech, word inflections, syllabification, etc., and stressing rules are especially simple and are created simply from a list of stressed words. Such methods are usually applied to non-inflectional languages. The decision tree algorithm is used to create the rules. Several ways of drawing up rules have been investigated. It was shown that the method of endings and beginnings gives the greatest accuracy (95.53%), whereas the smallest number of rules is received when applying the method of the mid-word rule, and the set is decreased by using one of the algorithms proposed by the author. In its accuracy the proposed method is inferior to the method based on morphological rules only by as little as 0.8%, however, it was shown that with increasing the number of texts used for creating the rules, this accuracy can be expected to be achieved and exceeded.

4) A rhythmical pattern, that is, alternation of stressed and unstressed syllables, is inherent feature of the spoken language. Seeking to avoid the adjacent stressed syllables some of the words remain unstressed (become clitics). The present author proposes methods of four types to search for the unstressed words in a Lithuanian text: methods based on recognising the combinational forms, based on statistical stressed/unstressed frequency of a word, grammar rules and stressing of the adjacent words. Word classes are defined for each method to which the method suites best and it is explained how to unite all the methods into a single algorithm. When minimising the sum of errors of the first and second kind, 4.1% of errors was obtained for the testing data among all the words, and the ratio of errors and unstressed words accounts for 18.8%.

References

[Kasparaitis, 2000] Kasparaitis, P. (2000). Automatic Stressing of the Lithuanian Text on the Basis of a Dictionary. *Informatica*, **11**(1), 19-40.

[Kasparaitis, 2001] Kasparaitis, P. (2001). Automatic Stressing of the Lithuanian Nouns and Adjectives on the Basis of Rules. *Informatica*, **12**(2), 315-336.

[Kazlauskienė et al., 2004] Kazlauskienė, A., G. Norkevičius, G. Raškinis (2004). Automatizuotas lietuvių kalbos veiksmažodžių kirčiavimas: problemos ir jų sprendimas. *Baltų ir kitų kalbų fonetikos ir akcentologijos problemos*, 166-173.

[Mauricaitė, 1985] Mauricaitė, V. (1985). Kai kurių frazės faktorių įtaka žodžių šlijimui. *Kalbotyra*, **36**(1), 38-43.

[Mauricaitė, 1987] Mauricaitė, V. (1987). Samplaikinių formų dėmenų akcentinis šlijimas. *Mūsų kalba*, **2**, 3-6.

[Norkevičius et al., 2004] Norkevičius, G., A. Kazlauskienė, G. Raškinis (2004). Bendrinės lietuvių kalbos daiktavardžių ir būdvardžių kirčiavimo struktūrinis modelis, algoritmas ir realizacija. *Kalbų studijos*, **6**, 72-76.

[Rimkutė, Grybinaitė, 2004] Rimkutė, E., A. Grybinaitė (2004). Dažniausios lietuvių kalbos morfologinio daugiareikšmiškumo rūšys ir jų automatinis vienareikšminimas. *Kalbų studijos*, **5**, 74-78.

[Rimkutė, Grigonytė, 2006] Rimkutė, E., G. Grigonytė (2006). Automatizuotas lietuvių kalbos morfologinio daugiareikšmiškumo ribojimas. *Kalbų studijos*, **9**, 30-37.

[Stundžia, 1991] Stundžia, B. (1991). Kirtis tekste. *Žodžiai ir prasmės*, **1**, Mokslas, Vilnius, pp. 86-92.

Publications on the Thesis Topic

1) Anbinderis, T., P. Kasparaitis (2007). Klitikų paieškos lietuviškame tekste algoritmai. *Kalbų studijos*, **10**, 30-37.
2) Anbinderis, T., P. Kasparaitis (2009). Lietuvių kalbos homografų vienareikšminimas remiantis leksemų ir morfologinių pažymų vartosenos dažniais. *Kalbų studijos*, **14**, 25-31.
3) Anbinderis, T., (2010). Automatic Stressing of Lithuanian Text Using Decision Trees. *Information Technology And Control*, **39**(1), 61-67.

About the Author

Tomas Anbinderis was born in 1981. In 2005 he graduated from Vilnius University (Faculty of Mathematics and Informatics) and he has been admitted as a PhD student in Vilnius University. Current research interests include text-to-speech synthesis and digital image processing. From 2006 to 2010: lecturer at Vilnius University, Faculty of Mathematics and Informatics.

Kai kurių lietuvių kalbos teksto kirčiavimo aspektų matematinis modeliavimas

Temos aktualumas. Kalbos sintezės sistemos – tai sistemos, kurios automatiškai generuoja žmogaus kalbą iš bet kokios tekstinės įvesties. Tam, kad sintezuota kalba skambėtų suprantamai ir natūraliai (o tiksliau – teisingam teksto transkribavimui bei intonacijos ir garsų trukmių modeliavimui), reikia nustatyti teksto žodžių kirčiavimą. Papildomų problemų atsiranda kirčiuojant žodžius, kurie yra vienodai rašomi, bet skirtingai tariami (homografai). Be to, šnekamajai kalbai būdingas ritmas, t. y. kirčiuotų ir nekirčiuotų skiemenų kaitaliojimasis. Siekiant išlaikyti kalbos ritmą, kai kurie žodžiai lieka nekirčiuoti (tampa klitikais). Lietuvių kalba yra laisvo kirčio stipriai kaitoma kalba, todėl automatinio kirčiavimo uždavinys yra netrivialus.

Darbo tikslai ir uždaviniai – sukurti lietuvių kalbos tekstų automatinio kirčiavimo algoritmus ir realizuoti juos kompiuterinėse programose. Algoritmai turėtų nenusilesti tikslumu kitiems jau egzistuojantiems algoritmams (jei tokie yra). Siekiant šio tikslo buvo sprendžiami šie uždaviniai:

1) Nusakyti automatinio kirčiavimo, homografų vienareikšminimo ir klitikų paieškos vietą bendroje balso sintezės schemoje, jų sąveiką su kitais moduliais, gaunamus ir perduodamus duomenis. Išnagrinėti kitose kalbose šių uždavinių sprendimui taikytus metodus.
2) Paruošti didelį (maždaug milijono žodžių) įvairių žanrų kirčiuotą tekstyną. Tam tikslui sukurti programinę įrangą, reikalingą tekstyno paruošimui. Šis tekstynas bus naudojamas eksperimentams ir algoritmų tikslumui įvertinti.
3) Pasiūlyti naują lietuvių kalbos homografų vienareikšminimo algoritmą.
4) Pasiūlyti naują lietuvių kalbos žodžių kirčiavimo algoritmą.
5) Pasiūlyti kalbos lietuvių kalbos ritmo nustatymo (klitikų paieškos) algoritmą.
6) Realizuoti pasiūlytus algoritmus ir eksperimentiškai įvertinti jų tikslumą.

Tyrimų metodika apima kalbotyros, kompiuterinės lingvistikos, atpažinimo teorijos, programavimo žinias. Eksperimentiniai tyrimai ir šiame darbe pasiūlyti algoritmai atlikti naudojantis specialiai šiuo tikslu sukurta programine įranga, parašyta *C++* programavimo kalba, naudojant *Microsoft Visual Studio 6.0* programavimo aplinką.

Mokslinis naujumas. Lietuvių kalbos automatiniam kirčiavimui iki šiol buvo išimtinai taikyti tik morfologine analize grįsti metodai. Tokie algoritmai yra sudėtingi, todėl sunkiai perkeliami į kitas programavimo kalbas ar operacines sistemas, juos sunku modifikuoti ar optimizuoti. Šiame darbe pasiūlytas metodas remiasi tik raidžių sekomis, nereikalauja jokių žinių apie kalbą, todėl yra itin paprastas, greitas, lengvai pritaikomas kitoms kalboms. Kirčiavimo taisyklės sudaromos automatiškai iš didelio kiekio kirčiuotų tekstų. Tokie metodai paprastai taikomi nekaitomoms arba silpnai kaitomoms (nefleksinėms) kalboms.

Lietuvių kalbos homografų vienareikšminimui iki šiol buvo naudoti HMM, ID3 ir sintaksine analize grįsti metodai. Visi jie remiasi žodžio kontekstu. Šiame darbe pasiūlytas metodas remiasi leksemų ir morfologinių pažymų dažniais, o kontekstinės informacijos visai nenaudoja.

Taip pat pasiūlytas lietuvių kalbos klitikų paieškos algoritmas. Kalbotyros darbuose galima rasti aprašytas tik bendras žodžių virsmo klitikais tendencijas, o klitikų automatinio radimo algoritmai dar visai nebuvo nagrinėti.

Ginamieji teiginiai

1) Lietuvių kalbos homografų vienareikšminimo algoritmas, pagrįstas leksemų ir morfologinių pažymų vartosenos dažniais.
2) Lietuviško teksto kirčiavimo algoritmas, pagrįstas raidžių sekomis žodžiuose. Kirčiavimo taisyklių automatinio sudarymo iš didelio kiekio kirčiuotų tekstų algoritmas. Kirčiavimo taisyklių skaičiaus sumažinimo algoritmas.
3) Lietuvių kalbos klitikų paieškos tekste algoritmas, pagrįstas samplaikinių formų atpažinimu, žodžio kirčiavimo/nekirčiavimo statistiniu dažniu, gramatikos taisyklėmis bei gretimų žodžių kirčiavimu.

Praktinis taikymas. Disertacijoje pasiūlyti homografų vienareikšminimo, klitikų paieškos bei žodžių kirčiavimo algoritmai yra naudojami 1) internetiniame lietuvių kalbos sintezatoriuje [http://www.studijos.lt/sintezatorius, žiūrėta 2010.04.13], 2) *UAB „Etalinkas“* lietuvių kalbos sintezatoriuje (prieiga per internetą: [http://www.etalink.lt/lietuviu-kalbos-sintezatorius, žiūrėta 2010.01.05]), 3) kirčiavimo programoje AccentTools (žr. šio darbo 3 skyrių).

Darbo apimtis. Disertacija susideda iš įvado, šešių skyrių, išvadų, literatūros sąrašo, dviejų priedų, bei sąvokų ir santrumpų sąrašo. Pagrindinė dalį sudaro 139 puslapiai įskaitant 22 paveikslėlius ir 25 lenteles. Literatūros sąraše 160 nuorodų. Disertacija parašyta lietuvių kalba.

Bendrosios išvados

Šiame darbe nagrinėjamas lietuvių kalbos teksto automatinis kirčiavimas bei su tuo susiję kiti du uždaviniai – homografų vienareikšminimas ir klitikų paieška.

1) Darbe nusakyta automatinio kirčiavimo, homografų vienareikšminimo ir klitikų paieškos vieta bendroje balso sintezės schemoje, jų sąveika su kitais moduliais, gaunami ir perduodami duomenys. Išnagrinėti metodai, taikyti šiems uždaviniams spręsti kitose kalbose, jų pasirinkimas atsižvelgiant į kalbos kaitymo laipsnį ir kirčiavimo paradigmą.

2) Lietuvių kalbos tekstui kirčiuoti pritaikius morfologinėmis taisyklėmis grįstus metodus, kai kuriuos žodžius (homografus) galima sukirčiuoti keliais būdais. Norint padidinti kirčiuotų žodžių skaičių, reikalingas vienareikšminimo algoritmas. Autoriaus pasiūlytas homografų vienareikšminimo algoritmas, pagrįstas leksemų ir morfologinių pažymų dažniais. Kai kurių leksemų buvimas žodyne labiau kliudo kirčiuoti, nei padeda. Jas atmetus, teisingų kirčiavimo hipotezių dalis tarp visų hipotezių (mokymo duomenims) padidėja 6,1%. Sudarytas morfologinių pažymų dažniais grįstų taisyklių rinkinys (1215 taisyklių), kuris mokymo duomenims leidžia teisingą kirčiavimo variantą parinkti 84,3% tikslumu. Pritaikius pasiūlytus algoritmus tekstui kirčioti, homografus pavyko vienareikšminti 85,01% tikslumu. Nors pasiūlytas algoritmas nenaudoja jokios informacija apie kontekstą, tačiau gauti rezultatai artimi kontekstinę informaciją naudojančiam ID3 algoritmui.

3) Morfologinėmis taisyklėmis grįsti lietuvių kalbos teksto kirčiavimo metodai pasižymi sudėtingumu, todėl juos problematiška perkelti į kitas sistemas ar programavimo kalbas. Šiame darbe autoriaus pasiūlyti metodai, kurie remiasi tik raidžių sekomis ir nereikalauja jokių žinių apie kalbą: morfemas, kalbos dalis, žodžių kaitymą, skiemenavimą ir pan., o kirčiavimo taisyklės yra itin paprastos ir sudaromos tiesiog iš kirčiuotų žodžių sąrašo. Tokie metodai paprastai taikomi nefleksinėms kalboms. Taisyklėms sudaryti naudotas sprendimo medžių

algoritmas. Nagrinėti keli taisyklių sudarymo būdai. Parodyta, kad didžiausią tikslumą (95,53%) duoda pabaigos-pradžios taisyklių metodas, o mažiausiai taisyklių gaunama taikant žodžio vidurio taisyklių metodą ir kai taisyklių aibė sumažinama naudojant vieną iš autoriaus pasiūlytų algoritmų. Savo tikslumu pasiūlytas metodas tik 0,8% nusileidžia morfologinėmis taisyklėmis grįstam metodui, tačiau parodyta, kad, didinant taisyklėms sudaryti naudojamų tekstų kiekį, galima tikėtis šį tikslumą pasiekti ir aplenkti.

4) Šnekamajai kalbai būdinga ritmika, t. y. kirčiuotų ir nekirčiuotų skiemenų kaitaliojimas. Siekiant išvengti greta esančių kirčiuotų skiemenų, kai kurie žodžiai lieka nekirčiuoti (tampa klitikais). Nekirčiuojamų žodžių paieškai lietuvių kalbos tekste autoriaus pasiūlyti keturių tipų metodai: pagrįsti samplaikinių formų atpažinimu, žodžio kirčiavimo/nekirčiavimo statistiniu dažniu, gramatikos taisyklėmis bei gretimų žodžių kirčiavimu. Kiekvienam metodui apibrėžtos žodžių klasės, kurioms jis geriausiai tinka, bei paaiškinta, kaip visus metodus sujungti į vieną algoritmą. Minimizuojant pirmosios ir antrosios rūšies klaidų sumą, testavimo duomenims gauta 4,1% klaidų tarp visų žodžių, o klaidų ir nekirčiuotų žodžių santykis yra 18,8%.

Trumpos žinios apie autorių

Tomas Anbinderis gimė 1981 metais. 2005 metais įgijo informatikos magistro laipsnį Vilniaus universiteto Matematikos ir informatikos fakultete. Nuo 2006 iki 2010 metų dirbo Vilniaus universiteto Matematikos ir informatikos fakultete asistentu.