INFORMATICA, 2008, Vol. 19, No. 4, 505–516

Lithuanian Speech Recognition Using the English Recognizer

Pijus KASPARAITIS

Department of Computer Science II, Faculty of Mathematics and Informatics, Vilnius University
Naugarduko 24, 03225 Vilnius, Lithuania
e-mail: pkasparaitis@yahoo.com

Received: November 2007; accepted: February 2008

Abstract. The present work is concerned with speech recognition using a small or medium size vocabulary. The possibility to use the English speech recognizer for the recognition of Lithuanian was investigated. Two methods were used to deal with such problems: the expert-driven (knowledge-based) method and the data-driven one. Phonological systems of English and Lithuanian were compared on the basis of the knowledge of phonology, and relations between certain Lithuanian and English phonemes were established. Situations in which correspondences between the phonemes were to be established experimentally (i.e., using the data-driven method) and the English phonemes that best matched the Lithuanian sounds or their combinations (e.g., diphthongs) in such situations were identified. The results obtained were used for creating transcriptions of the Lithuanian names and surnames that were used in recognition experiments. The experiments without transcriptions, with a single transcription and with many transcriptions were carried on. The method that allowed finding a small number of best transcriptions was proposed. The recognition rate achieved was as follows: 84.2% with the vocabulary containing 500 word pairs.

Keywords: speech recognition, small and medium size vocabulary, expert-driven approach, data-driven approach.

Introduction

The idea of controlling a computer by means of voice emerged a long time ago but a low recognition rate has been the main obstacle to doing that for a considerable time. During the past years the speech recognition rate has already achieved the level necessary to control a computer by voice. One of complete implementations of this idea is the Microsoft's new Windows Vista operating system. However, large companies (like Microsoft) are interested mostly in the most popular languages, so the best results are achieved for these languages. E.g., in Vista, Speech Recognition is available in English (U.S.), English (U.K.), German (Germany), French (France), Spanish (Spain), Japanese, Chinese (Traditional), and Chinese (Simplified) (`http://www.microsoft.com/enable/products/windowsvista/speech.aspx` downloaded 2007 10 20). What could be the simplest solution for the users of smaller national languages (e.g., Lithuanian)?

Possible solutions are as follows: to create one's own engines or to adapt the ones created for other languages. Many authors have already tried to create the Lithuanian speech recognizers (e.g., Filipovič and Lipeika, 2004; Laurinčiukaitė, 2003; Lipeika *et al.*, 2002; Raškinis and Raškinienė, 2003; Rudžionis and Rudžionis, 1996). Having chosen the former method, which seems more complicated and time-consuming, one can be faced with the problems relating to the compatibility of the new engine and other software. Consequently, the second option will be discussed in more detail in this paper. Systems created for a certain language can be successfully adapted to other languages, e.g., a list of seven successful projects is presented in (Schultz and Waibel, 2001). Two following methods – expert-driven and data-driven ones – are used when porting recognition engines from one language into another (Villaseñor-Pineda *et al.*, 2005). In the event of the first method an expert makes a decision about the similarity of certain sounds on the basis of the similarity of their phonological features. The acoustic data are used in the second case; the similarity between the sounds is determined by analyzing confusion matrixes or using distance metrics. In some works better results were obtained when employing the first method, e.g., (Žgank *et al.*, 2004), whereas in other works, e.g., (Byrne *et al.*, 2000) the second method was more efficient. The expert driven method can be applied first where correspondences between the sounds are obvious, and the second method is used where there are uncertainties (Villaseñor-Pineda *et al.*, 2005). After the most similar sounds have been found, their models can still be adjusted.

The possibilities to apply the speech recognition engine Microsoft English Recognizer 5.1 from the package Microsoft Speech SDK 5.1 (`http://www.microsoft.com/downloads/` downloaded 2007 05 02) for the recognition of Lithuanian will be investigated in this paper. Microsoft English Recognizer 5.1 can recognize both a continuous speech and separate commands. In the latter case the words or phrases from which the recognition engine must choose the answer are written in a special file having the xml format and called Command and Control Grammar (Microsoft Speech SDK 5.1 Help). Certain Lithuanian and English words sound very similar, e.g., the Lithuanian word *taip* (*yes*) sounds like the English word *type*. So, seeking to recognize the Lithuanian word *taip* the following xml tag should be written in the Command and Control Grammar: <PHRASE DISP="taip">type</PHRASE>, where the attribute DISP indicates the text that will be returned by the recognition engine. Most Lithuanian words have no such equivalents in English but we can create artificial words that sound like the Lithuanian words when pronounced according to the English rules, e.g., the surname of the author of this paper can be written as follows: <PHRASE DISP="Kasparaitis">Kuspurightis</PHRASE>. However, this method is quite complicated, and the rules for creating such artificial words are not clear. It is difficult to find an exact equivalent even for a highly useful word *ne* (*no*). Fortunately, this is unnecessary. The pronunciation using the English phonemes can be given to the above-mentioned recognition engine, e.g., the word *ne* can be written as follows: <PHRASE PRON="n eh 1">ne</PHRASE>. The transcription presented in the attribute PRON (rather than the textual form) will be used for recognition therefore the textual form can be written directly in the xml tag.

The problem of transcribing the Lithuanian words using the English phonemes will be considered in this paper. First the expert-driven method will be used, and then, in the

remaining unclear cases, the data-driven method will be applied. No training in the recognition engine will be provided. The use of the English speech recognition engines for Lithuanian on the basis of the data-driven method only was also investigated in (Rudžionis *et al.*, 2007).

1. Expert-Driven Approach

The IPA phoneme systems of both languages are usually used in the expert-driven methods. Microsoft English Recognizer 5.1 uses another system of phonemes. The list of phonemes can be found in Microsoft Speech SDK 5.1 help. A list of 49 signs is presented here, the first 9 signs are intended for marking the boundaries of sentences, the stress and so on, the remaining 40 signs are names of phonemes. The names are built of small letters or pairs of small letters, i.e., the system is similar to the ARPAbet (Jurafsky and Martin, 2000).

There are 58 phonemes in Lithuanian (Girdenis, 1995). We shall use the notation system proposed in this work, i.e., different notation systems will be used for the Lithuanian and English phonemes.

Since a text is the result of recognition, the problem under investigation can be treated as follows: transcribing a Lithuanian text using the Lithuanian phonemes and finding the relation between the Lithuanian and English phonemes. The problem can be simplified by removing the intermediate step, i.e., it can be treated as transcribing a Lithuanian text using the English phonemes. Rules (Kasparaitis, 1999) or a dictionary (Skripkauskas and Telksnys, 2006) can be used for transcribing, or this can be done manually, because the number of transcriptions is defined by the size of the vocabulary of recognition.

The following notation will be used in this work: the Lithuanian phonemes will be written between slashes //, the Lithuanian letters will be written between double quotes and the English phonemes – between brackets [].

It should be noted, that soft and hard consonants are different phonemes in Lithuanian, e.g., *vagių* (*thief* Gen. case, plural) and *vagų* (*furrow* Gen. case, plural), i.e., the same letter (in this example "g") is used for two phonemes (soft and hard) except "j" that means only a soft consonant. Soft and hard consonants are not distinguished in English, so two Lithuanian phonemes correspond to a single English phoneme. Now, on the basis of (Piesarskas and Svecevičius, 1991), relations between most Lithuanian and English phonemes can be found, see Table 1, though some of the Lithuanian and English phonemes are pronounced quite in a different way (comments are given in Table 1). The list of Lithuanian phonemes was taken from (Girdenis, 1995).

Table 1 allows us to have a general impression of how well the Lithuanian phonological system is covered by the English one. However, further in this paper we are going to discus mainly the problem of transcribing the Lithuanian letters (or their combinations) using the English phonemes.

The following English phonemes were left unused: the vowels [ax] and [er], the diphthongs [aw], [ay], [ey], [ow], [oy], the consonants [dh], [ng], [th], [w]. The consonant

Table 1
Relation between Lithuanian and English phonemes

	Lithuanian phonemes	Lithuanian letters	English phoneme	Example		Lithuanian phonemes	Lithuanian letters	English phoneme	Example
1	/a/	“a”	[ah]	cut	19	/ts/, /ts’/	“c”	–	–
2	/e/	“e”	[eh]	pet	20	/dz/, /dz’/	“dz”	–	–
3	/i/	“i”	[ih]	fill	21	/t$\int$/, /t$\int$’/	“č”	[ch][3]	chin
4	/o/, /o:/	“o”	[ao][1]	dog	22	/dʒ/, /dʒ’/	“dž”	[jh][4]	joy
5	/u/	“u”	[uh]	book	23	/s/, /s’/	“s”	[s]	sit
6	/a:/	“a”, “ą”	[aa]	father	24	/z/, /z’/	“z”	[z]	zap
7	/e:/	“e”, “ę”	[ae]	cat	25	/$\int$/, /$\int$’/	“š”	[sh]	she
8	/i:/	“į”, “y”	[iy]	feel	26	/ʒ/, /ʒ’/	“ž”	[zh]	pleasure
9	/u:/	“ų”, “ū”	[uw]	too	27	/x/, /x’/	“ch”	–	–
10	/ė:/	“ė”	–	–	28	/h/, /h’/	“h”	[h]	help
11	/ie/	“ie”	–	–	29	/f/, /f’/	“f”	[f]	fork
12	/uo/	“uo”	–	–	30	/j’/	“j”	[y]	yard
13	/p/, /p’/	“p”	[p][2]	put	31	/v/, /v’/	“v”	[v]	vat
14	/b/, /b’/	“b”	[b]	big	32	/l/, /l’/	“l”	[l]	lid
15	/t/, /t’/	“t”	[t][2]	talk	33	/m/, /m’/	“m”	[m]	mat
16	/d/, /d’/	“d”	[d]	dig	34	/n/, /n’/	“n”	[n]	no
17	/k/, /k’/	“k”	[k][2]	cut	35	/r/, /r’/	“r”	[r][5]	red
18	/g/, /g’/	“g”	[g]	gut					

[1] The same phoneme is used for short and long vowel.

[2] It is pronounced with aspiration in some cases.

[3] Two consonants [t][sh] can be used instead of this one.

[4] Two consonants [d][zh] can be used instead of this one.

[5] It sounds very differently from the Lithuanian counterpart.

[ng] can be successfully used as an allophone of the consonant /n/ before /g/ and /k/. The Lithuanian consonants “c” and “dz” can be transcribed into [t][s] and [d][z] respectively. The English diphthongs can be used when transcribing relevant Lithuanian diphthongs: “ai” – [ay], “ei” – [ey], “au” – [aw], “oi” – [oy], but the English diphthongs are always stressed with the falling accent and the Lithuanian diphthongs can be stressed with the rising accent too. Hence, we are free to use different variants for the Lithuanian diphthongs.

The following questions of aligning the phoneme systems remain unanswered:

1) how to model soft consonants before the vowels “o”, “u”, “ų”, “ū”;
2) should the letters “ia” (including the diphthongs “iai”, “iau”) be transcribed like “e” and should “ja” (“jau”, “jai”) be transcribed like “je”;
3) what left and right component should be used to build the diphthongs “ai”, “ei”, “oi”, “ui”, “au”, “eu”, “ou”, “ie”, “uo”;

4) should the consonants “č” and “dž” be built of two phonemes (like “c” => [t][s] and “dz” => [d][z]), or a single phoneme should be used;
5) what English phonemes suit best to transcribe Lithuanian letters “ė” and “ch”.

2. Data-Driven Approach

The data-driven approach can be used where the alignment of phonological systems of two languages leaves open questions. In this case the list of alternatives (or combinations of alternatives) should be drawn first, e.g., transcriptions [ah ih], [ah iy], [ah y], [aa ih], [aa iy], [aa y] , [ax ih], [ax iy], [ax y] and [ay] if we wish to investigate the Lithuanian diphthong “ai” (see Table 2). Then we need a word containing the phoneme or diphthong of interest, e.g., “taip”. It is advisable to find a word where transcription of other letters is obvious. Now we need to transcribe the word in all possible ways and to put the transcriptions into the Command and Control Grammar, e.g.,

<PHRASE PRON=“t ah ih p”>t ah ih p</PHRASE>,
<PHRASE PRON=“t ah iy p”>t ah iy p</PHRASE>,
...
<PHRASE PRON=“t ay p”>t ay p</PHRASE>.

Now we can simply say the word into a microphone and the recognition engine chooses the best transcription for us. Repeating this procedure many times with different speakers and different words we can calculate the percentage each transcription variant was recognized.

We usually want to have a single Lithuanian phoneme that corresponds to a single English phoneme. This requirement is unnecessary to fulfil if a limited vocabulary is used. In this case it is only important to have the vocabulary entries that have at least one different phoneme. E.g., if we want to recognize one of the two words *re* and *fa* they can be transcribed as follows: [r ax] and [f ax], where the same English phoneme [ax] corresponds to two different Lithuanian phonemes /a/ and /e/. In most experiments we shall try to meet the above-mentioned requirement.

Experiments were carried out seeking to verify if stressing had an impact on recognition. Two Lithuanian words *likime* and *kilime* were used for this purpose. Any syllable can be stressed in these words, so 3 stressing alternatives of each word were used, e.g., [l ih 1 k ih m eh], [l ih k ih 1 m eh] and [l ih k ih m eh 1], where the figure of one marks the stressed syllable. Experiments showed that the first alternative was recognized in all experiments and that the results did not depend on stressing. This means that stressing does not have a significant influence on recognition. Analogous experiments showed that putting the stress mark on different phonemes seeking to model different accents has no influence on recognition either.

Most experiments were carried out with diphthongs. The same number of samples with the unstressed, stressed with falling and stressed with rising accent was used. Taking into account the fact that stressing does not have a considerable influence on recognition, the results were averaged rather than calculated separately.

Table 2
The frequency of transcription variants of diphthongs

Diphthong	Left side of diphthong	Speaker I	Speaker II	Speaker III	Average	Right side of diphthong	Speaker I	Speaker II	Speaker III	Average
ai	ah	6%	6%	6%	6%	ih	3%	5%	19%	9%
	aa	28%	65%	29%	**41%**	iy	79%	92%	47%	**73%**
	ax	61%	27%	33%	40%	y	13%	1%	2%	5%
	ay						6%	2%	33%	14%
ei	eh	81%	45%	26%	**51%**	ih	4%	0%	17%	7%
	ae	2%	14%	6%	7%	iy	77%	59%	15%	**50%**
						y	2%	0%	0%	1%
	ey						17%	41%	67%	42%
oi	ao	48%	17%	24%	30%	ih	0%	2%	0%	1%
						iy	24%	15%	24%	21%
						y	24%	0%	0%	8%
	oy						52%	83%	76%	**70%**
ui	uh	3%	21%	46%	23%	ih	2%	0%	4%	2%
	uw	97%	79%	54%	**77%**	iy	80%	94%	96%	**90%**
						y	18%	6%	0%	8%
au	ah	9%	55%	20%	32%	uh	3%	2%	2%	2%
	aa	19%	22%	19%	21%	uw	28%	20%	16%	21%
	ax	61%	14%	48%	**38%**	w	58%	67%	70%	**65%**
	aw						11%	11%	13%	11%
eu	eh	63%	70%	33%	**55%**	uh	5%	0%	0%	2%
	ae	5%	0%	8%	4%	uw	45%	0%	0%	15%
						w	18%	70%	40%	**43%**
	ow						33%	30%	60%	41%
ou	ao	0%	10%	17%	9%	uh	0%	0%	0%	0%
	ah	7%	18%	3%	9%	uw	33%	1%	13%	16%
	aa	23%	16%	0%	13%	w	17%	64%	23%	35%
	ax	20%	22%	17%	20%					
	ow						50%	34%	63%	**49%**
uo	uh	5%	8%	12%	8%	ao	2%	1%	13%	5%
	uw	95%	92%	88%	**92%**	ah	2%	44%	28%	25%
						ax	96%	48%	58%	**67%**
						aa	1%	6%	0%	2%
ie	ih	9%	1%	9%	6%	ax	71%	14%	60%	**52%**
	iy	91%	99%	91%	**94%**	eh	19%	32%	28%	26%
						ah	9%	48%	8%	22%
						ae	1%	0%	4%	2%

Table 3

The frequency of transcription variants of diphthongs after “i” and “j”

Diphthong after “i”, “j”	Left side of diphthong	Speaker			Average	Right side of diphthong	Speaker			Average
		I	II	III			I	II	III	
iai	ih-(ah,aa,ax,ay)	25%	5%	13%	14%	ih	2%	0%	0%	1%
	y-(ah,aa,ax,ay)	36%	1%	13%	17%	iy	64%	24%	25%	38%
	(eh, ae)	5%	17%	0%	7%	y	0%	0%	0%	0%
	ey						34%	76%	75%	**62%**
iau	ih-(ah,aa,ax,aw)	27%	30%	47%	35%	uh	0%	0%	0%	0%
	y-(ah,aa,ax,aw)	59%	23%	27%	**36%**	uw	31%	38%	13%	27%
	(eh, ae)	9%	21%	13%	14%	w	56%	52%	32%	**47%**
						aw	8%	10%	41%	20%
	ow						6%	0%	13%	6%
jai	y-(ah,aa,ax,ay)	10%	0%	10%	7%	ih	0%	0%	0%	0%
	y-(eh, ae)	90%	80%	80%	**83%**	iy	100%	80%	90%	**90%**
	ey						0%	20%	10%	10%
jau	y-(ah,aa,ax,aw)	67%	33%	3%	34%	uw	7%	10%	0%	6%
	y-(eh, ae)	27%	20%	3%	17%	w	87%	43%	10%	47%
						aw	0%	0%	3%	1%
	ow						6%	47%	87%	47%

Four sample words for each stressing variant were used in the experiments (a total of 12). Each word was pronounced 10 times. Three male speakers (aged form 22 to 40) took part in the experiment. Microsoft English Recognizer 5.1 was used for speech recognition. The Recognizer was not trained for a particular speaker; besides, automatic adaptation was switched off during the experiments. The results are presented in Tables 2, 3 and 4. To obtain more generalized results the number of occurrences of each phoneme on the left and right side rather than that of the whole diphthong was calculated. The notation ih-(ah,aa,ax,ay) in Table 3 and 4 means all combinations: [ih ah], [ih aa], [ih ax] and [ih ay].

3. Results of the Alignment of Phonemes

The following conclusions could be drawn on the basis of the data presented in Tables 2–4:

1) The diphthongs “oi” and “ou” should be transcribed into the English phonemes [oy] (70%) and [ow] (49%). Most words containing these diphthongs are from English. The remaining diphthongs should be made of two components.
2) The phoneme [iy] (much rarer the phoneme [y]) suits best as a second component of the diphthongs “ai”, “ei”, “ui” (73%, 50% and 90% respectively, see Table 2),

Table 4
The frequency of transcription variants of letters

Letter or combination	Variant of transcription	Speaker			Average
		I	II	III	
ia, ią	ih-(ah, aa)	42%	18%	24%	28%
	y-(ah, aa)	3%	0%	14%	6%
	(eh, ae)	55%	82%	62%	**66%**
ja, ją	y-(ah, aa)	0%	5%	0%	2%
	y-(eh, ae)	100%	95%	100%	**98%**
io, iu, iū, ių	(ao, uh, uw)	52%	40%	44%	**45%**
	ih-(ao, uh, uw)	30%	8%	20%	19%
	y-(ao, uh, uw)	18%	52%	36%	35%
č	ch	87%	56%	80%	**74%**
	t-sh	13%	44%	20%	26%
dž	jh	97%	80%	50%	**76%**
	d-zh	3%	20%	50%	24%
ė	eh	0%	1%	40%	14%
	ae	0%	0%	12%	4%
	ax	25%	0%	8%	11%
	er	0%	0%	0%	0%
	ey	75%	99%	40%	**71%**
ch	s	0%	35%	0%	12%
	th	8%	15%	3%	9%
	dh	3%	10%	35%	16%
	f	3%	25%	15%	14%
	sh	3%	0%	5%	3%
	ch	13%	5%	0%	6%
	h	73%	10%	43%	**42%**

though according to Lithuanian orthography it should be the phoneme [ih], and according to phonology – [y].

3) The phoneme [w] (somewhat rarer the phoneme [uw]) suits best as a second component of the diphthongs “au”, “eu” (and “ou” if we build it from two components). According to Lithuanian orthography it should be the phoneme [uh], according to phonology – [v].
4) The tense phonemes [iy], [uw] rather than the lax phonemes [uh], [ih] as could be expected are used on the left side of the diphthongs “ie”, “ui” and “uo”. The phoneme [ax] (somewhat rarer the phoneme [ah]) suits best to be used on the right side of the diphthongs “ie” and “uo”, rather than the phonemes [eh] and [ao], respectively, as could be expected according to orthography.
5) The phoneme [eh] suits best as the left component of the diphthongs “ei” and “eu”. There are no strong regularities for the diphthongs starting with “a” (i.e.,

"ai" and "au"). The phonemes [aa] and [ax] suited somewhat better the diphthong "ai" whereas [ax] and [ah] – the diphthong "au".

6) The vowels "ia" ("ią") should be changed into [eh] ([ae]), much rarer it should be changed into two phonemes [ih][ah] ([ih][aa]).
7) The vowels "o", "u", "ū" and "ų" following a soft consonant were recognized if they followed a hard one. This would pose a problem when recognizing the words, which differ in softness of the consonants only. If we still want to model softness, it is preferable to add the phoneme [y] after the consonant rather than [ih].
8) The affricates "č" and "dž" should be treated as a single phoneme rather than a combination of phonemes.
9) The vowel "ė" is most similar to the phoneme [ey], which was really difficult to expect.
10) It is difficult to find an equivalent for the consonant "ch", often it is similar to the voiced consonant [h].

4. Recognition Experiments

A set of experiments was carried out seeking to evaluate the improvement in the recognition rate achieved when using transcription. Pairs of words (a surname and a name) were used in the experiments. For the sake of simplicity we shall refer this pair as a surname. The recognition rate depends largely on the number of alternatives, thus the size of the vocabulary was 10, 50, 100 and 500 surnames. One test was carried out with 500 alternatives, five tests were conducted with 100 alternatives, 5 tests – with 50 alternatives and 10 tests – with 10 alternatives. No transcription was used in the first experiment. The Lithuanian letters in the surnames were replaced with the Latin ones in the following way: ą => a, ę,ė => e, į => i, ų,ū => u, č => c, š => s, ž => z. One transcription variant chosen on the basis of the data (maximum value) presented in Tables 2–4 was used in the second experiment, e.g., "ei" was transcribed into [eh iy] (these phonemes have the frequency 51% and 50% respectively, see Table 2). Two male speakers took part in the experiments (aged 22 and 40). They read all 500 surnames once. A computer recorded the voice of each speaker, so the same record was used in all experiments. The results are presented in Table 5, rows 1 and 2, respectively. An obvious improvement can be seen when transcriptions were used: from 34.7% to 68.4% using 500 surnames and from 77.5% to 95.0% using 10 surnames. Somewhat better results of the first speaker can be accounted for by the fact that only the first speaker participated in the experiments described in Section 2.

Seeking to improve the results even further many transcription variants of each surname were generated and used in the recognition. Only those transcription variants of diphthongs and other combinations of letters presented in Tables 2–4 were used, which had the frequency of at least 10% in the above-presented tables, e.g., when transcribing the diphthong "ou" (Table 2) the phonemes [aa] (13%) and [ax] (20%) were used as the left side of the diphthong, the phonemes [uw] (16%) and [w] (35%) were used as the right side of the diphthong and the phoneme [ow] (49%) was used as the whole diphthong.

Table 5
Results of the recognition of surnames with/without transcription

No	Experiment	Speaker	Size of the vocabulary			
			10	50	100	500
1	Transcription not used	I	76.0%	60.4%	50.8%	32.0%
		II	79.0%	60.8%	51.0%	37.4%
		Average	**77.5%**	**60.6%**	**50.9%**	**34.7%**
2	Single transcription variant	I	99.0%	90.8%	85.0%	72.6%
		II	91.0%	88.0%	78.6%	64.2%
		Average	**95.0%**	**89.4%**	**81.8%**	**68.4%**
3	Many transcription variants	I	82.0%	–	–	–
		II	79.0%	–	–	–
		Average	**80.5%**	–	–	–
4	Two best transcription variants	I	100.0%	95.2%	91.0%	85.2%
		II	98.0%	94.8%	92.0%	83.2%
		Average	**99.0%**	**95.0%**	**91.5%**	**84.2%**

A list of transcriptions of certain surnames was drawn up by producing all possible combinations of transcriptions of the letters. On average 200 transcriptions were produced for each surname. The test with 10 alternatives was carried out only. Since the results were rather disappointing (Table 5, row 3) tests with 50, 100 and 500 alternatives were not carried out. Obviously too many variants, which were similar to other surnames, were generated, which accounts for such a significant slump in the results.

Seeking to reduce the number of transcription variants it was decided to choose one or more best transcriptions for each surname and to use only them in recognition. Hence, a huge set of possible transcriptions was generated for each surname. In addition to the variants mentioned in the previous paragraph, the letters “a”, “ą”, “e” and “ę” were always transcribed in two ways (using a short and long phoneme), the letters “i”, “į”, “y”, “u”, “ų” and “ū” were also always transcribed in two ways (using a tense and lax phoneme). Now we received approximately 1200 variants per surname. Transcriptions of a single surname were put into the recognition grammar and the records of both speakers were used. In this way we found the best transcription of a certain surname for each speaker. After this procedure has been repeated with all the surnames, we had 1000 best transcriptions (500 for each speaker). These transcriptions were put into the recognition grammar and experiments with a different number of alternatives were carried out. The results are represented in Table 5, row 4. The improvement is obvious.

If there are many speakers the same method can be used to find the best transcription of a certain vocabulary entry. With the number of speakers increasing variants of transcriptions are expected to repeat themselves, therefore the number of different transcriptions of the same surname should not be too large. In the case of two speakers 10.8% transcriptions were the same. If the number of transcriptions is too large, rarer occurring

ones (or preferably those that are misrecognized) can be removed. The improvements referred to in this paragraph were not investigated in this work.

Conclusions

Seeking to find correspondence between the Lithuanian and English phonemes both expert-driven and data-driven methods were used. The expert-driven method was used to establish obvious relations between the phonemes whereas the data-driven method was employed where such relations were not so evident. Some experiments showed relations between components of the Lithuanian diphthongs and the English phonemes, which was difficult to expect. The results were used for creating transcriptions of the Lithuanian surnames that were used in the experiments of recognition from a small and medium size vocabulary (10, 50, 100 and 500 alternatives). The recognition rate increased by as much as 17.5–33.4% when transcriptions were used as compared with the experiment carried out without transcriptions (the Lithuanian letters were just replaced with the Latin ones). The recognition rate decreased by as much as 14.5% (10 alternatives) when many generated transcriptions of each vocabulary entry were used. The method for finding several best transcriptions of each vocabulary entry was proposed. In this case the recognition rate increased by 5–15.8%. Generally speaking the recognition rate achieved (99% with 10 alternatives and 84.2% with 500 alternatives) shows that the English speech recognition engine can be used for the recognition of the Lithuanian words provided that the vocabulary is small. The results obtained are comparable or only slightly worse than those obtained by other authors using engines designed specially for Lithuanian, e.g., 80% with 750 alternatives in (Raškinis and Raškinienė, 2003) and 86.7% with 750 alternatives in (Filipovič and Lipeika, 2004).

Acknowledgements

This research was supported by the Lithuanian State Science and Studies Foundation.

References

Byrne, W., P. Beyerlein, J. M. Huerta, S. Khudapur, B. Marthi, J. Morgan, N. Peterek, J. Picone, D. Vegyri and W. Wang (2000). Towards language independent acoustic modeling. In *In Proc. ICASSP*, vol. 2. pp. 1029–1032.

Filipovič, M., and A. Lipeika (2004). Development of HMM/neural network-based medium-vocabulary isolated-word lithuanian speech recognition system. *Informatica*, **15**(4), 465–474.

Girdenis, A. (1995). *Teoriniai fonologijos pagrindai*. Vilniaus universitetas, Vilnius (in Lithuanian).

Jurafsky, D., and J.H. Martin (2000). *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Upper Saddle River, New Jersey 07458.

Kasparaitis, P. (1999). Transcribing of the Lithuanian text using formal rules. *Informatica*, **10**(4), 367–376.

Laurinčiukaitė, S. (2003). Isolated Lithuanian word recognition based on hidden Markov models. In *Proc. of Information Technologies 2003*, vol. IX. KTU, Kaunas. pp. 21–24 (in Lithuanian).

Lipeika, A., J. Lipeikienė and L. Telksnys (2002). Development of isolated word speech recognition system. *Informatica*, **13**(1), 37–46.

Piesarskas, B., and B. Svecevičius (1991). *Lithuanian–English Dictionary*. 2nd edition (revised). Mokslas, Vilnius.

Raškinis, G., and D. Raškinienė (2003). Building medium-vocabulary isolated-word Lithuanian HMM speech recognition system. *Informatica*, **14**(1), 75–84.

Rudžionis, A., K. Ratkevičius and R. Maskeliūnas (2007). Adaptation of English speech recognition engines for Lithuanian speech recognition. In *Proc. of 3rd Baltic Conference on Human Language Technologies*. Kaunas (to be published).

Rudžionis, A., and V. Rudžionis (1996). Izoliuotų žodžių atpažinimas vidurkinant fonetiškai segmentuotus kalbinių signalų parametrus. In *Informacinės technologijos-96*. Technologija, Kaunas. pp. 168–174 (in Lithuanian).

Schultz, T., and A. Waibel (2001). Language independent and language adaptive acoustic modeling for speech recognition. *Speech Communication*, **35**(1–2), 31–51.

Skripkauskas, M., and L. Telksnys (2006). Automatic transcription of Lithuanian text using dictionary. *Informatica*, **17**(4), 587–600.

Villaseñor-Pineda, L., V.B. Le, M. Montes-y-Gómez and M. Pérez-Coutiño (2005). Toward acoustic models for languages with limited linguistic resources. *Lecture Notes in Computer Science*, **3406**, 433–436.

Žgank, A., Z. Kačič, F. Diehl, K. Vicsi, G. Szaszak, J. Juhar and S. Lihan (2004). The COST 278 MASPER initiative – crosslingual speech recognition with large telephone databases. In *Proc. of 4th International Conference on Language Resources and Evaluation* (*LREC'04*), vol. VI. Lisbon (Portugal). pp. 2107–2110.

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. In 2001 he defended a thesis for a doctoral degree. Current research interests include text-to-speech synthesis and other areas of computer linguistic.

Lietuvių kalbos atpažinimas naudojant anglų kalbos atpažinimo variklį

Pijus KASPARAITIS

Šiame darbe nagrinėtas kalbos atpažinimas esant mažam arba vidutinio dydžio žodynui. Tirta galimybė anglų kalbos atpažinimo variklį panaudoti lietuvių kalbai. Tokio tipo problemoms paprastai naudojamas vienas iš metodų: paremtas žiniomis ir paremtas duomenimis. Remiantis fonologijos žiniomis palygintos anglų ir lietuvių kalbų fonologinės sistemos ir nustatytos atitinkamybės tarp kai kurių lietuvių ir anglų kalbų fonemų. Surasti tie atvejai, kuomet atitinkamybes tarp fonemų reikia rasti eksperimentiškai ir kokios anglų kalbos fonemos kokius lietuvių kalbos garsus ar jų junginius (pvz., dvibalsius) geriausiai atitinka šiais atvejais. Rezultatai panaudoti sudarant lietuviškų vardų ir pavardžių transkripcijas, kurios buvo naudojamos fiksuoto žodyno atpažinimo eksperimentuose. Atlikti eksperimentai siekiant palyginti atpažinimo tikslumą nenaudojant transkripcijų, kiekvienai pavardei naudojant po vieną transkripciją ir po daug transkripcijų. Pasiūlytas metodas, kaip išrinkti kelias geriausias transkripcijas. Pasiektas atpažinimo tikslumas 84,2%, kai naudojamas 500 žodžių porų žodynas.