Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics

Automatic speech generation in infocommunication systems

PhD thesis booklet
PhD School in Computer Science and Information Technology

Csaba Zainkó, MSc

Supervisors:
Géza Németh, PhD
Gábor Olaszy, DSc

Budapest, 2010

1. Introduction

Natural communication between human and machine has an increasing role in infocommunication systems. The users and the nature of applications have changed: today's systems are used not exclusively by computer experts but rather by a wide range of people, on a daily basis. A successful human-computer interface allows everyone, not just experts, both to understand the commands and outputs of the machine and to be able to give commands.

One of the research areas aiming at improving human-computer communication, and the one that I have been in the focus of my research, is automatic speech generation and the related field of automatic text processing. In these fields, most of the problems and solutions are language-dependent, thus one needs to consider even those issues that are solved for other languages (mainly English). Due to the peculiarities of the Hungarian language solutions developed for other kinds of languages do not work or are only partially successful, while procedures rejected for other languages may work in Hungarian. On the other hand, research results for Hungarian can be applied in an international setting as well, as they can easily be adapted to similar inflective languages, provided that the language-dependent parts are well delimited.

The first experiments aiming at speech synthesis were reported by Farkas Kempelen over 200 years ago, in 1791 (Kempelen 1969 (original edition: 1791)). However, the first electronic synthesizer (with manual control) was built only in the 20th century, at Bell Labs in 1939 (Homer et al. 1939). The foundations of automatic speech generation were laid by Fant (1960), while the first text-to-speech system based on formant synthesis was published by Rabiner (1968). Speech generation research for Hungarian started a decade later, and the first Hungarian-language computer-based speech synthesizer, called Hungarovox (Kiss–Olaszy 1984) was implemented in 1980 in the Phonetics Laboratory of the Linguistics Research Institute, Hungarian Academy of Sciences. The first synthesizers produced intelligible speech but their voice was very robotic. Researchers from the Budapest University of Technology (now Budapest University of Technology and Economics, BME), led by Géza Gordos, also joined the research and the development (Gordos–Takács 1983, Gordos–Sándor 1985). In those days, MITalk (Allen et al. 1987) and DecTalk (Hallahan 1995) were created for English. Another major result was the development of the PSOLA algorithm (Hamon et al. 1989) for automatic prosody modification, that has since been frequently employed in speech technology research.

In the 90s, the increase in computing power allowed purely software-based solutions, with a sound quality close to human speech. A review of these results can be found in Péter Olaszi's PhD dissertation (Olaszi 2002). The large amount of written material available on the Internet was a great opportunity for the advancement of text processing. For English, the British National Corpus was started to collect a text database in 1991 (Burnard 1995). As for Hungarian, the collection of the Hungarian National Corpus started in 1998 at the Department of Corpus

Linguistics of the Linguistics Research Institute, Hungarian Academy of Sciences (Váradi 1999).

I joined the Laboratory of Speech Technology at the Department of Telecommunications and Telematics of BME (now Department of Telecommunications and Media Informatics of BME) in 1997, just when the diphone-based waveform-concatenation speech synthesizer, Profivox, was under development (Olaszy et al. 2000).

For English, CHATR (Black–Taylor 1994) and the open-source Festival (Black et al. 2006) were major speech synthesis systems. After the millennium, the main goal was to achieve natural speech quality. One successful solution for this goal was Rhetorical's unit selection system, rVoice, for which there were speech databases available on multiple languages (Rutten et al. 2002, Rutten–Fackrell 2003). Another new approach appeared in the early 2000s, one that adapts the long standing technique of speech recognition applying Hidden Markov Models (HMM) to speech synthesis (Tokuda et al. 2000, Zen et al. 2007). Besides natural-sounding speech synthesis, the production of speech with emotions also started to get attention in research (Scherer 2003).

For Hungarian, smaller companies created their own, closed speech synthesis systems, for example Speech Technology Ltd. There were no papers published about these systems. Further, the freely available MBROLA synthesis system was adapted to Hungarian (MBROLA 2006). In 2010, larger companies, like SVOX and Nuance, launched their Hungarian speech synthesizers.

After 2000, it was only the Department of Telecommunications and Telematics of BME where Hungarian speech synthesis research was persistently going on. A hybrid system was developed for reading names and addresses [C4], and then a corpus-based synthesis system approaching the quality of natural speech [C12]. Following the international research trends, the Laboratory of Speech Technology at the department joined the research efforts on HMM-based speech synthesis and developed a solution for Hungarian (Tóth–Németh 2008). Besides synthesizing neutral speech, investigations on systems capable of expressing emotions were also started [C5,C9].

Speech synthesis

The first step in the process of speech synthesis is the preparation of the input text. Figure 1.1 shows a simplified block diagram of speech synthesis. During text preparation, the input string is transformed into a letter sequence that contains only alphabetical characters, spaces and punctuation. Normalization involves the resolution of acronyms, the transcription of foreign names and words into a form suitable for conversion to speech. Numbers are transcribed into their textual forms, considering whether the number in question is a phone number, an amount of money, a date, a cardinal or an ordinal number. This processing step also involves the reconstruction of diacritics in texts written without them.

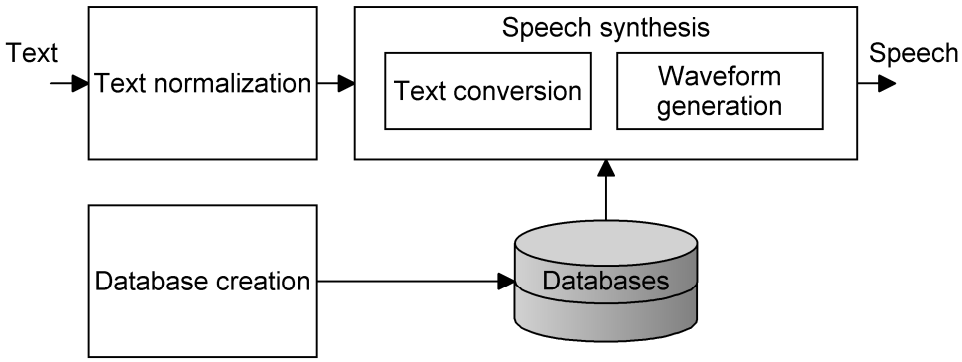

Figure 1.1: Simplified block diagram of the speech synthesis process

The normalized text serves as the input for the central block of the speech synthesizer. This central part can be divided into two main units. The first is the text conversion, while the second is the waveform synthesis. During text conversion, the input letter sequence is transformed into a speech sound sequence and related prosodic information. The waveform is generated based on these data, that can either be the final speech signal or a raw sound sequence, depending on the speech synthesis technology. A raw sound sequence is turned into the final speech output by further signal processing steps.

Speech synthesizers usually work with various types of databases. These databases contain waveforms or texts, or, in case of an HMM synthesizer, speech coding parameter values.

My results are presented in relation to the main components on Figure 1.1. My theses are organized into thesis groups along the following areas: Thesis group I presents procedures for diacritic reconstruction, related to text preparation. In Thesis group II, I summarize my results in speech database creation. In Thesis groups III and IV, my novel solutions are described in automatic speech generation.

2. Research objectives

My goal is to develop algorithms and methods to improve the quality of communication between human and computer in infocommunication systems.

Accordingly, one of my objectives is to statistically explore the Hungarian language, focusing on the aspects relevant for speech synthesis. This also allows the assessment of the adaptability of methods and procedures developed for other languages. In this field, it was earlier impossible to examine large-scale databases due to the limitations of computer technologies. I consider it important to verify data (using the novel processing capabilities) obtained with earlier, smaller databases and thus to show new relationships among them.

My other objective is to extend earlier speech synthesis methods, in order to extend the range of potential applications. My goal is to make synthetic speech sound more and more like human speech, while its robotic, machine-like nature

becomes less and less noticeable. I extend automatic speech synthesis with new attributes, such as the different ways to generate emotionally expressive speech.

In my work, I evaluated the procedures I developed for Hungarian but I always kept in mind their applicability to other languages as well. I paid special attention to either separate the language-dependent parts of the proposed solutions or to make them easily adaptable to other languages.

In reaching my objectives, I have always considered the capabilities and constraints of infocommunication systems. Infocommunication systems, such as the widely used Interactive Voice Response (IVR) systems, require real-time, scalable solutions. Resources are often limited: a single PC might need to serve 240 parallel channels at the same time.

3. Methodology

The research methods applied are widespread principally in speech technology and in related disciplines.

In the first phase of my research, I used text corpora that I collected from the Internet. In later phases, I also used the Hungarian National Corpus, that is reasonably representative for the typical utterances in contemporary Hungarian. For processing and analysis, I mainly employed general sorting and classification algorithms.

For the measurements on speech databases, there was no large-scale, Hungarian read-speech corpus available. For this reason, I performed the measurements on speech material that was recorded for various research and development projects at the Laboratory of Speech Technology. I used the functions of Praat (Boersma–Weenink 2010) for analyzing speech databases and waveforms.

I implemented the algorithms and procedures with the tools of MATLAB and RapidMiner, as well as in C language, when limited resources made this language a reasonable choice.

4. New results

4.1 Thesis group I: Procedures for the automatic reconstruction of texts written without diacritics

It is common in speech processing, especially in speech synthesis that input texts with missing or partially missing diacritics need to get processed. Diacritic-free texts may be created by a user not using letters with diacritics, either because it is inconvenient, or because the device or configuration does not allow an appropriate input method for these characters. Another reason for having such texts is the loss of diacritic information during storage or transmission, i.e. during conversion, some symbols with diacritics become diacritic-free.

An example in Hungarian:

```
Agyunk a beszedet nem onmagaban dolgozza fel,
hanem az osszes erzekszervunkbol kapott informaciot
kombinalja es ertelmezi.
```

Humans can read diacritic-free texts without any significant difficulties, our brain can almost completely recover the original contents of the text. In this case, our visual perception builds on our linguistic knowledge to infer the missing diacritics. However, if a text without diacritics is converted to speech by machine, our perceptual system is not capable of easily processing and understanding it by listening, because the lack of diacritics usually changes the phoneme. We do not understand the distorted sound body or misunderstand it. For example, an utterance of „*mogott*", instead of the word „*mögött*" ('behind' in Hungarian), is meaningless. Further, hearing „*agyat*" ('brain', accusative) in place of „*ágyát*" ('his bed', accusative) may cause confusion in understanding due to its different meaning. This is why it is necessary to check the presence or absence of diacritics before reading texts, and to perform diacritic recovery.

The problem of diacritic recovery is not limited to Hungarian but it is also necessary to handle this issue in other languages using diacritics (Mihalcea–Nastase 2002). Internationally, there are papers for other languages that use diacritics (Mihalcea–Nastase 2002, De Pauw et al. 2007, Ungurean et al. 2008). The problem was investigated mainly for Romanian. The published algorithms are usually based on machine learning, that give a general solution, but do not take into account the requirements of infocommunication systems and speech synthesis. For example, the error categories caused by the methods on the word level - e.g., creates an agrammatical word - were not examined, although they may influence the quality of synthesized speech.

Thesis I.1: I developed a dictionary-based procedure for recovering the original form (with diacritics) of computer-readable texts written without diacritics. [P1,B2,J1,J2,J3,C1]

The basis of the dictionary-based solution is the building of a dictionary (based on a large text database), that contains the diacritic-free word forms, as well as the corresponding, linguistically correct forms with diacritics. When building the dictionary, word frequencies are also taken into account. Whenever a diacritic-free word may have multiple possible correct forms, the algorithm chooses the statistically most likely one. The procedure was evaluated for Hungarian, where it reached an accuracy of 95% in texts of e-mails [C1], while 96% in a general domain [C10]. The procedure is protected by Hungarian patent [P1] number 226740 P 00 03443.

The advantage of this solution is that it always returns a grammatically correct word form for lexical units used during dictionary building. The algorithm uses a large text database in the preparatory stage, but the dictionary created for recon-

struction has a small size and allows fast lookup. Its disadvantage is the lack of generalization capability, as it cannot reconstruct words that are out of the dictionary. If it should reconstruct a word that was missing from the text database used for extracting statistical data, then, due to the lack of data, it would keep the diacritic-free version (that can even be meaningless).

Thesis I.2: I developed a procedure for reducing the errors of the dictionary-based algorithm [C10]

The procedure described in Thesis I.1 does not handle ambiguous cases (words), as it always decides for the more frequent case. Ambiguous cases are the word forms where there is more than one correct form for a single diacritic-free word, for example, „*veres - verés - véres*". I eliminated this shortcoming by disambiguating the ambiguous cases by a decision tree, making the decision among the potential alternatives based on the environment. My procedure is an extension of the method of (Mihalcea–Nastase 2002) that use machine learning on the character level. I adapted this method for word-based processing, as explained below:

The list of ambiguous cases, together with their frequencies are available as an output of the procedure described in Thesis I.1. I built a J48 tree on these words, that takes into account the environment of the word to disambiguate it, i.e. to choose from the multiple possible versions with diacritics. The J48 decision tree is the freely available Java-based implementation of C4.5 (Quinlan 1993).

The algorithm fits well with the procedure of Thesis I.1, because I separated the processing steps with high computational complexity from the fast reconstruction procedure. Building the dictionary is memory-intensive, and a large amount of text (several 10 millions of words) needs to get processed. During reconstruction, however, only a small decision tree needs to be parsed, that can be performed quickly. One limitation of the algorithm is that, for training, there needs to be a substantial amount of training material for each ambiguous case. The advantage of the algorithm is that, compared to the procedure in Thesis I.1, it has generalization capabilities: it can make a decision, even for an environment for which there are no training data.

The algorithm with this error-reduction procedure, trained with material from the Hungarian National Corpus, can handle 60% of the ambiguous cases and correctly reconstructed the diacritics in 93% of these cases [C10]. Although the improvement is small compared to the total number of errors, it handles cases that (in Hungarian) would otherwise make speech comprehension harder, for example: „*Megvetette az ágyát.*" - „*Megvetette az agyát.*" (He/She prepared his/her bed - He/She prepared his/her brain)

Further, the algorithm of Thesis I.1 does not handle the cases that are not in the dictionary of the algorithm. I applied a decision tree for these cases that works by considering the character context.

The J4.8 decision tree is built based on the 20-character environment of the letters to recover. In Hungarian, we apply diacritics to vowels only, for which the corresponding letters are: „*a, e, i, o, u*"). The resulting decision trees (5 for Hungarian) are applied for the letters of words that were left unresolved in the algorithms presented earlier.

The algorithm based on decision trees has a high level of generalization capability, it can be run on any words, even those that are not part of the training dataset. The procedures described in Thesis I.2 corrected the mistakes of the method of Thesis I.1 in 60% of the cases, in a general domain [C10]. Ambiguous cases, that were left unhandled before, were reconstructed with an accuracy of 83% [C10].

4.2 Thesis group II: Analysis and comparison procedures of texts written in Hungarian and in other languages, mainly for supporting automatic speech generation

The input of automatic text-to-speech conversion is usually written text. In order to synthesize high-quality speech, knowledge about the properties of the input text is necessary. Statistical data about written texts can be applied in the construction of speech databases (that form the basis of speech synthesizers) and in testing the synthesis algorithms as well. Further, data comparisons across languages can contribute to the adaptation of algorithms developed for other languages to Hungarian, as well as to the international comparison of the results.

Thesis II.1: I determined the basic applicability conditions for the comparison of English-, German- and Hungarian-language word-based automatic speech technology methods [B3,J4,J5,C2]

Due to the inflective nature of Hungarian, the number of grammatically correct and meaningful word forms is extremely large – according to some estimates, it is in the range of billions (one million lexemes (Kenesei et al. 1984) and, for example, a single verb can have 1000 inflected forms (Prószéky 1988)). The number of words that are actually used is smaller than that and there are also loanwords in the language. In order to establish the number and distribution of actually used word forms, I employed texts from the following sources: Hungarian-language works from the Hungarian Electronic Library, works from the Digital Literary Academy, articles from online periodicals and the Hungarian National Corpus. These sources altogether constitute a collection of 80 million word forms.

I determined the frequency ranking of Hungarian words and their coverage (ratio of the full corpus covered by their occurrences). The results of the word-frequency analyses are shown with thick solid lines on Figure 4.1.

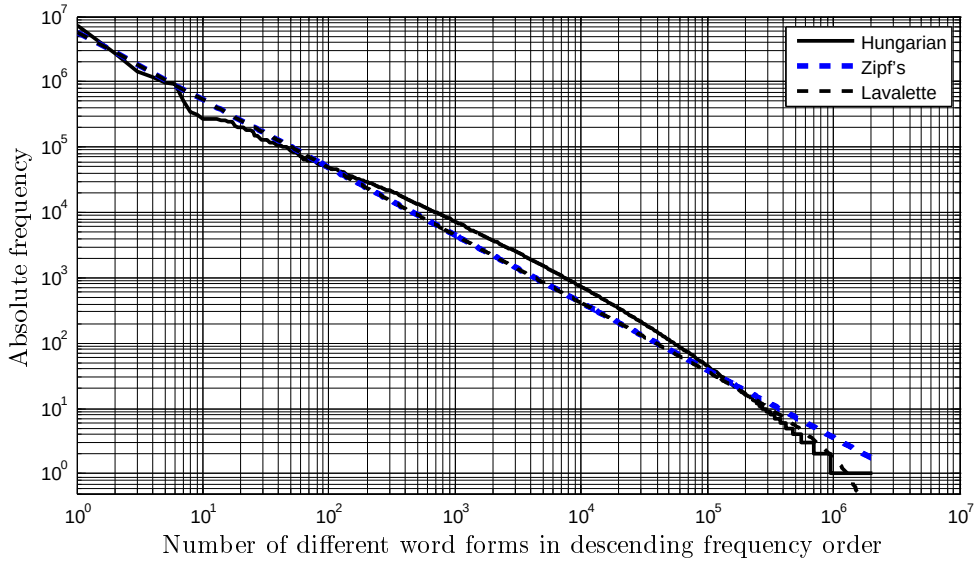

Figure 4.1: Relationship of word frequency and ranking

I compared the frequency data with Zipf's law – a widely used principle in linguistic research (Li 1992) –, that is shown on the figure with a thick dashed line. The frequency data followed Zipf's law ($f(r) = C \cdot r^{-b}$) where C is a normalization constant and b is usually around 1. Both axes of the figure are logarithmic, thus the exponential curve can be plotted as a straight line. The constants of the function fitted to the frequency data are the following: $C = 10^{6.7573}$, $b = 1.033$. The curve described by Zipf slightly deviates from the measured data at low frequency items. In order to overcome this problem, I used the Lavalette formula (Popescu 2003), that can be described by Equation 4.1. The constants are the same as in Zipf's formula.

$$f(r) = C \cdot \left(\frac{r_{max} r}{r_{max} - r + 1} \right)^{-b} \tag{4.1}$$

The Lavalette curve can also be seen on Figure 4.1 (thin dashed line).

I used various text corpora for comparing the three languages. For English, a version of the British National Corpus (BNC), containing 89 million word forms were used, as well as the English-language texts of the Hungarian Electronic Library. For German, the basis of the comparison was the material of the Gutenberg project. The Hungarian data was from examined 80 million words corpus. Further, the translation of the Bible was available for all three languages. The analysis of this latter text was performed separately, because it was uniform across languages both in its content and in its size.

I determined the ratio of the corpus covered by the words (analyzed in frequency rank order). Table 4.1 shows three examples of coverage ratios and the number of words needed to achieve these ratios in the three languages.

Another way to analyze the data is when we compare the coverage in different languages based on a given number of words. An example for this kind of analysis can be seen in Table 4.2.

Table 4.1: The number of most frequent words needed as a function of corpus coverage

Language	Coverage		
	75 %	90%	97,5%
English	1250	5800	20 100
German	2000	14 550	80 000
Hungarian	10 650	70 000	400 000

Table 4.2: Corpus coverage with a given number of words

Language	Number of most frequent words		
	1000	20 000	100 000
English	72,8%	97,5%	(100%)
German	69,1%	91,8%	98,1%
Hungarian	51,8%	80,7%	92,0%

The detailed figures and data can be found in [J5] and in the dissertation.

I also performed the comparative analyses on texts in the same domain, on three versions of the Bible. The coverage curves are shown on Figure 4.2. The absolute frequencies are different from the data obtained from the large corpora above, but the ratios across languages remained the same. The Hungarian data is denoted by a dotted line, the English is shown with thick lines, while German is plotted with thin lines. For example, when comparing the 90% coverage of large corpora in Table 4.1 with the 90% line in Table 4.2, one can see that the ratio of three between English and German is the same, as well as the one order of magnitude difference between English and Hungarian.

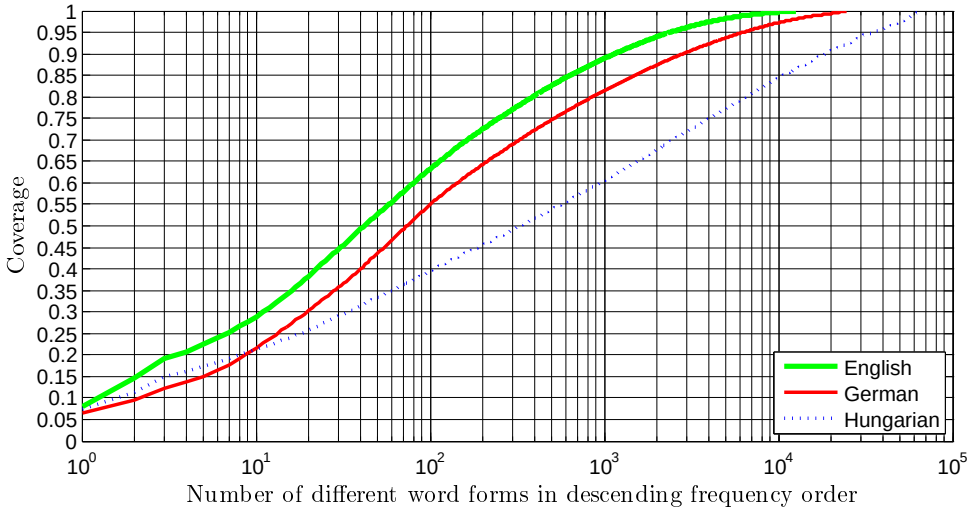

Figure 4.2: Coverage curves of the Bible in three different languages

My method and the published data can thus be applied to examine the adaptability of word-based algorithms from one language to another. My analysis method is not restricted to the three languages I examined, but it can be applied to show similarities and differences across other languages.

Thesis II.2: By extending the notion of the letter, I developed a novel method for the qualification of texts for the purpose of speech synthesis [C8]

The written and spoken forms of language are closely related. Language and speech processing brought the need for statistical analyses that consider the interaction of these two levels. This needs a novel approach, one that is based on word statistics and on the statistical processing of the letters constituting the words. The information about phones contained in the written form also needs to be considered. One can get a full overview of the statistical properties of a language only if the same large-scale linguistic material is measured both on the text level and on the acoustic level. During classification, I also considered the sound representation that can be attributed to the letter sequence. For the measurements, I have developed automatic sorting and collection algorithms, specifically for this purpose.

I extended and customized the notion of letter for this purpose. The longest letter sequence analyzed for the character- and letter statistics was word, the non-letter characters were ignored (numbers, relation symbols, etc). Extending the notion of letter means that phones were projected back to the writing level. For example, the characters *ch* in the word *pech* are considered a single letter consisting of two characters (similarly to *sz*), while the same two characters in the word *lánchíd ch* are treated as a separate *c* and an *h* letter. This novel letter-based classification preserves information that is lost in phonetic transcriptions. For example, letter statistics are available for *j* and *ly*, even though they are both pronounced as [j]. Similarly, *i* and *y* (that is common at the ends of historical names) are treated separately, even though they are both pronounced as [i].

It is important to note that these classifications are based on the requirements of speech technology, thus some of the decisions may seem incomplete from a linguistic point of view. For transcription and classification, we used an algorithm based on hyphenation rules (*Ri-chárd*, *Mün-chen*, *Ben- czúr*), as these letters are not hyphenable. Decisions were based on the vocabulary of the Hungarian hyphenation sample collection (Nagy 2008). Besides the two-character letters mentioned above, the algorithm takes into account the two-character letters in the Hungarian alphabet (*gy, ty, ny, sz, zs, cs*), as well as their long versions. The long versions were treated as two letters (*zzs= zs + zs*) during statistical analyses.

The text was converted into a phoneme sequence by using automatic speech technology methods. There were especially three tools that we used, a rule-based algorithm (the phonetic transcriber and rule collection of the Profivox text-to-speech system (Olaszy et al. 2000)), the Hungarian electronic pronunciation dic-

tionary (Abari–Olaszy 2006) and the proper name collection of the name announcer system [C4]. The rules developed represent standard Hungarian.

Using this method, I examined the material of the 2006 version of the Hungarian National Corpus. The detailed statistical analysis can be found in [C8] and in the dissertation.

4.3 Thesis group III: Procedures for improving the quality of synthetic speech

The quality of synthetic speech depends on the quality of each component, as well as the on the limitations concerning the domain of the input text. As a general principle for speech synthesis, the product of the size of the domain and of the quality is constant. Thus in a more restricted domain, one can synthesize higher quality speech than in a wider domain. This thesis group summarizes those of my speech generation-related results, that improved the quality of synthesized speech, in an environment with a limited domain but still having a large amount of variation.

Thesis III.1: I developed procedures and algorithms for the text-to-speech conversion of proper names, company names, and Hungarian addresses. [B1,C4,B5a]

Although reading names and addresses is a smaller domain than reading general texts, the domain is not bounded due to the nature of names. Continuously appearing new person names with a foreign origin and company names ignoring the rules of the language require the capability to read arbitrary letter sequences. However, one can benefit from knowledge about frequently occurring parts in input texts.

In the name and address reader, I combined triphone-based speech synthesis [J7], number reading (Olaszy–Németh 1999) and dictionary-based synthesis. The procedure is based on the fact that different types of elements can occur only in certain positions, so their prosody can be described and modeled in advance.

The generated sentence thus contains units from the triphone-based system, number units and separately read items. When compiling the data to be read, I identified and examined the categories introduced in Table 4.3.

I ran the analysis on a list of names and addresses, consisting of 3 million records, using the method described in Thesis II.1. I extracted the word ranking for each of the categories in the table. It is not possible to have all the words from these categories recorded because the voice talent is unable to read such a number of words with uniform style and voice. I reduced the number of words so that the reading list is not longer than what a trained announcer can read in a single session (4 hours of recording). The second column in Table 4.3 shows the number of

words found during the analysis, while the last column gives the number of words that was read. I determined the number of words to read based on the coverage curve, with the goal of achieving over 95% coverage, but using no more than 1000 items (due to limitations in the sound recording).

Table 4.3: Categories examined and their properties

Category	Number of different words	Items read
Family names	103850	0
Given names	1797	313
Business entity types	8	8
Town names	3523	1000
Types of thoroughfares	14	14
Spelling	36	36

In order to allow automatic text-to-speech conversion, I developed text processing rules that identify the various information items in common inputs and pass it to the waveform generator subsystem in proper order. The information items are the following:

Proper names: title, family name, given name
Business entities: business name, type of business entity
Addresses: zip code, town name, district name, name of thoroughfare, type of thoroughfare, number, staircase, floor, door

Identification of the information items is done based on the following three parameters: the item's position in the record, its presence or absence in the item dictionary, and position relative to items that have been identified already.

The information items were ordered the same way as above, and the waveforms belonging to each item were matched (synthesized from triphones, numbers generated by the number reader, and separately read items). The fundamental frequency component of prosody can be manipulated in case of the triphone-based waveform concatenation synthesizer. The number items can be produced with two different prosody contours, while the items read separately were recorded with the desired prosody. The implementation of prosody follows the intonation patterns of statements. As the intensity component of prosody, I used a simplified trajectory, that equalizes all the items, except the last, for which it prescribes an intensity decrease. Among the temporal components of prosody, pausing is the only one that can be manipulated (it was set with the goal of keeping the intelligibility high). Pauses were inserted at boundaries in the utterance, while their length was refined in perceptual experiments. I proved the effectiveness of the name- and address-reading procedure by an intelligibility test.

Thesis III.2: I developed a procedure, based on virtual word intensity, to intensity-normalize the speech produced by a corpus-based synthesizer [B5b, J6]

There is only limited measurement data about the intensity of Hungarian speech sounds (Olaszy 1989), according to the literature. Large amounts of data have not been processed yet. The speech sound intensities were examined only on sample sentences.

In my work, I used speech databases that I collected in sound recording studios [C12] and thus the signal-to-noise ratio of the recordings are in the range of 40-60 dB so that it does not influence the intensity measurements. When compiling the speech databases, recordings with a reading mistake, with extraneous sound material or with any other kind of error were removed. The boundaries of the speech sounds in the database were annotated semi-automatically [C6,C7] and major annotation errors were corrected by various algorithms [J10]. Sentences in the database consisted of 15 words on average, therefore it was possible to perform intensity normalization based on the root-mean-square level (RMS) of the sentence.

Measurements were performed on the speech material of 9 speakers (2 females and 7 males), constituting 57 hours of speech in total. I measured automatically the intensity relationships on the labeled database. The results are summarized on Figures 4.3 and 4.4.

I measured the intensity for each speech sound, between the sound boundaries, using RMS. On the figures, the data is normalized to the intensity of the sound *a*. For the convenience of the reader, sounds are denoted by the corresponding letters. Error bars represent standard deviation across speakers.

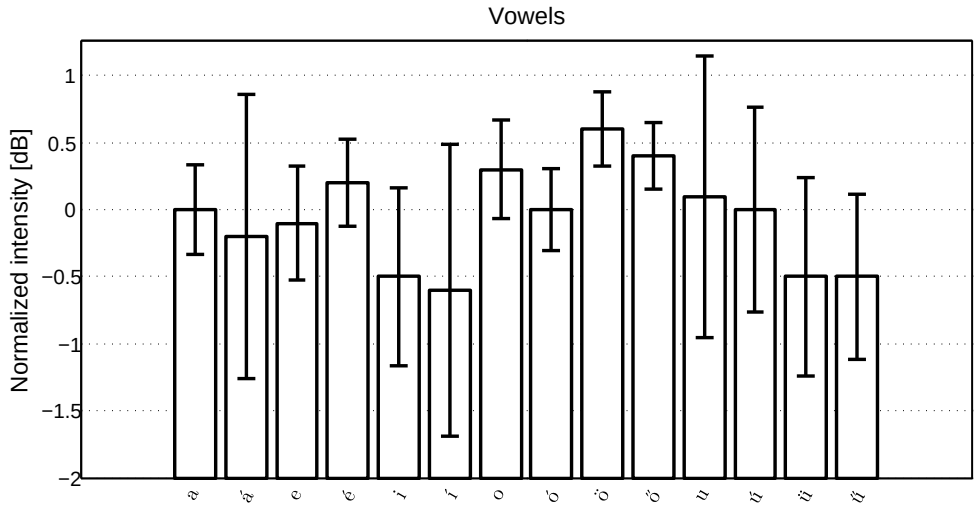

Figure 4.3: Intensity relationships of the vowels

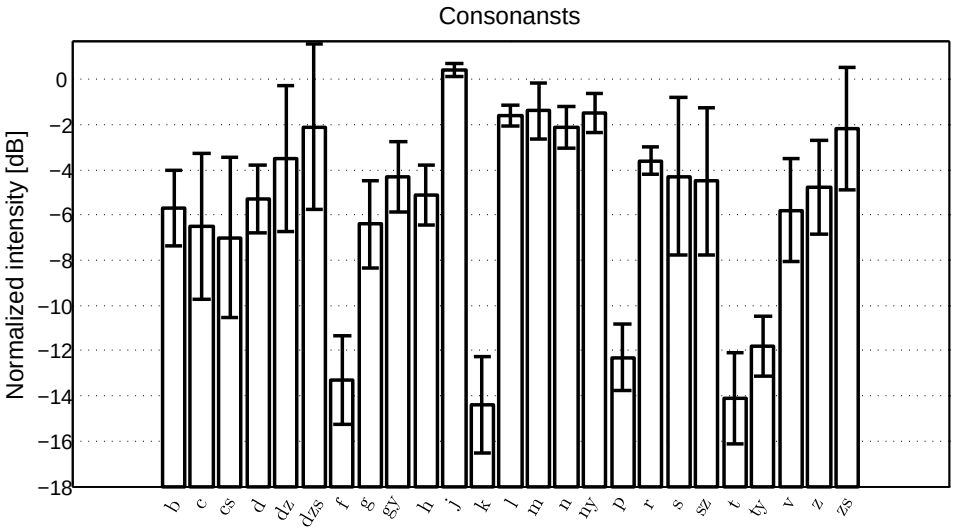

Figure 4.4: Intensity relationships of the consonants

These results are different from those in Olaszy (1989): the intensity values are more balanced. The intensity maps of the sounds are based on the average intensity levels, that makes it possible to study the intensity differences due to sentence prosody.

Creating the database of speech synthesizers is a complex process. The size of the speech databases used for corpus-based synthesis ranges from several hours to hundreds of hours. Such large databases can only be recorded in a number of sessions over several weeks or months. The intensity of the recordings needs to be uniform, but this can mostly be achieved by controlling the recording conditions, and by postprocessing. In case of an inhomogeneous speech database, containing short, individually uttered items, such as exclusively words or exclusively numbers, balancing the signal levels is a complex task. It cannot be achieved by general normalization procedures used in signal processing – such as the RMS normalization applied in introduced examination – because the normalized items would have different relative intensities in the concatenated signal. If a short item contains mainly sounds that are weaker in intensity than the average, then normalizing over the word leads to a wrong (overly intense) output. Inserting such normalized items leads to intensity jumps.

Corpus-based synthesizers can also build speech from units containing single sounds (Taylor 2009) whose intensity varies. In order to balance the intensity of such items, I introduced virtual word intensity that gives the intensity of the word carrying the sound in case all the sounds in the word would have average intensity (according to the values measured in introduced examination). An average, weighted by the sound durations, of the average sound intensities is calculated according to the formula below, where N is the number of sounds in the word, t_{ph} is the duration of the sound, $I_{ph}^{average}(ph)$ is the average intensity calculated in introduced examination and $ph(i)$ is the ith sound of the word:

$$I_{word}^{virtual} = \frac{\sum_{i=1}^{N} t_{ph}(i) I_{ph}^{average}(ph(i))}{\sum_{i=1}^{N} t_{ph}(i)} \tag{4.2}$$

Based on this value, one can calculate the amount of scaling needed:

$$gain_{ph} = \frac{gain_{prosody} \cdot gain_{base} \cdot I_{word}^{virtual}}{I_{word}^{real}} \tag{4.3}$$

In the formula above, $gain_{base}$ is the average intensity of the sentence, while $gain_{prosody}$ is the deviation from the average due to prosody. $I_{word}^{real}(ph)$ is the intensity of the word in which the sound to be normalized can be found. I evaluated this method of intensity normalization by a perceptual test.

When evaluating the quality of the test sentences, both normalization methods scored significantly ($p<0.05$) better than the versions without normalization. However, there was no significant difference between the two normalization procedures. Pairwise comparisons also showed that the sentences normalized with any of the methods were judged to be superior to the sentences without normalization. In case of two test sets, there was no significant difference between the two methods, but with the third test set, the method proposed in this thesis achieved significantly higher scores.

4.4 Thesis group IV: Procedures to modify the emotions conveyed by the voice of speech synthesizers

In the development of speech synthesizers, the fundamental goal was to produce good quality, intelligible speech. Synthesizers usually can produce only speech that is emotionally neutral, as the speech databases contain neutral read speech. One way to generate more expressive speech is to record databases with different emotions (Douglas-Cowie et al. 2003). This needs a substantial amount of time and resources thus, in many cases, such databases are not available. Another approach is to transform speech in the database so that the synthesizer produces speech similar to human speech expressing emotions. This latter way cannot produce the same quality as the other but it makes it possible to use existing speech databases.

In Thesis group IV, I present my results related to synthesizing speech with emotional content.

Thesis IV.1: I proposed a procedure to produce speech with emotional content that is based on direct transformation of the waveform. [B4,C5,C9]

I improved the procedure described by Přibilová–Přibil (2009) and developed the algorithm that can be applied to arbitrary speech. The procedure is based on a spectral transformation that builds on the observation that in speech with different emotions the ratio of low and high frequency components, as well as the distance of the first and second formant is different (Scherer 2003). Přibilová and Přibil developed the original method for an LPC-based synthesizer and applied it to Czech.

The improved procedure combines the PSOLA algorithm with Přibilová and Přibil's method. The time-domain signal is windowed pitch-synchronously by an asymmetric Hann window and then the windowed signal is transformed to the frequency domain by DFT. The spectral transformation – that is described in detail in the dissertation –, as well as the intensity adjustments are performed in the frequency domain. The transformation causes the signal to not converge to zero at the beginning and at the end of the window. I correct this distortion by windowing the signal again. The modified spectrum is converted back into a time-domain signal by inverse DFT, and then I overlap-and-add it similarly to the PSOLA algorithm. During overlap-and-add, all the necessary duration corrections are also performed.

In order to achieve a change in the conveyed emotion, fundamental frequency also needs to get modified, that can be implemented by shifting the overlaps – just like in PSOLA. There are two alternative solutions for unvoiced regions. One is to not perform the transformation for unvoiced regions, but then the transformation is incomplete. However, as unvoiced sounds have a low energy and the low frequency components in modified signals are typically weak, the lack of transformation is perceptually unnoticeable. Another possibility is to place uniformly spaced virtual pitchmarks on unvoiced regions and process these regions identically to voiced speech. During testing, I chose the first approach.

I designed a perceptual test to evaluate the procedure. Sentences generated by one human speaker and two corpus-based synthesizers were transformed. The three different utterances were manipulated to convey three emotions – sadness, joyful and angry. Participants were asked to judge the emotional content of the sentences that they listened to. The degree of manipulation applied was also a factor in the test. Results are shown in Table 4.4, while the details can be found in the dissertation.

The results suggest that, in case of female natural and synthetic sentences, sad and joyful emotions were most robustly recognized, thus these can be generated by the proposed methods.

Table 4.4: Confusion matrix of the emotions

		Judged													
		N	A	H	S		N	A	H	S		N	A	H	S	
Target	N	82%	0%	14%	6%	N	50%	23%	0%	27%	N	77%	0%	18%	5%	N=neutral
	A	27%	27%	41%	5%	A	36%	41%	0%	23%	A	45%	32%	14%	9%	A=angry
	H	27%	5%	68%	0%	H	40%	23%	14%	23%	H	9%	4%	82%	5%	H=joyful
	S	9%	5%	0%	86%	S	14%	5%	0%	81%	S	22%	5%	5%	68%	S=sad
		Corpus-based TTS-female				Corpus-based TTS-male					Natural female					

Thesis IV.2: I adapted the procedure of Thesis IV.1 to diphone- and triphone-based concatenative systems [B5b,C11]

The procedure proposed in Thesis IV.1 can be applied on the output of diphone- and triphone-based concatenative synthesizers (referred to as triphone synthesizer in the following), but the quality of the synthesized speech would be inferior due to the multiple stages of signal processing. As both the triphone synthesizer and the procedure in Thesis IV.1 performs a correction of the fundamental frequency, intensity and duration, the two processing steps can reasonably be merged for both quality and performance reasons.

I divided the procedure of Thesis IV.1 into two parts. The modification of the spectral components is performed on the waveform-database of the synthesizer. The speech database of the triphone synthesizer consists of diphones and triphones (two half-sound and one half, one full and another half sounds, respectively). The spectral modification needs to be done individually on all items. A separate database needs to be created for each emotion. The size of a database is in the range of 2-100 Mbyte, depending on the sampling rate and item count – thus the increased storage space needed is not significant considering the technologies available today. The other part of the emotion modification procedure consists of the rules that modify prosody. These rules were implemented by parameterizing the prosody prediction module.

I evaluated the procedure with a perceptual test similar to that in Thesis IV.1. I created 3 modified databases for the triphone synthesizer and then I synthesized sentences from a general domain. The results are summarized in Table 4.5. The details of the test are described in the dissertation.

Table 4.5: Confusion matrix of the emotions in case of the triphone synthesizer

		Judged				
		N	A	H	S	
Target	N	25%	0%	0%	75%	N=neutral
	A	29%	17%	29%	25%	A=angry
	H	25%	0%	67%	8%	H=joyful
	S	0%	0%	0%	100%	S=sad
		Triphone TTS-female				

For the triphone synthesizer, sad and joyful emotions can be expressed by the proposed method, however angry cannot. With the default, neutral settings of the synthesizer, participants judged the synthesized sentences to convey sadness.

Thesis IV.3: I adapted the procedure described in Thesis IV.1 to an HMM-based speech synthesis system [B5b,C11]

The principle of HMM speech synthesis is based on machine learning, thus there is no way to directly manipulate the parameters. It is feasible to apply the procedure of Thesis IV.1 to modify the output of the HMM synthesizer, however this solution is not satisfactory due to the multiple signal processing steps and the increased processing time (similarly to the triphone synthesizer in Thesis IV.2).

The sound databases of the HMM synthesizer are of the size of tens of hours, and the training phase is time-consuming (lasting for several weeks). Instead of the resource-intensive full training, I created emotional speech synthesis by adaptation of the training (Tóth–Németh 2009). I modified a small amount of prosodically rich acoustic material (about 10 minutes long) using the procedure of Thesis IV.1. I performed both the prosodic and spectral part of the modification. Besides neutral, I created sentences with sad, joyful and angry emotions. The transformed recordings were used to create speech databases [J10] that were adapted to models that had already been trained.

I evaluated the procedure with a perceptual test similar to that in Thesis IV.1. For the HMM synthesizer, I transformed a male voice to convey the three emotions and then I synthesized sentences with neutral topics. The results are summarized in Table 4.6.

Table 4.6: Confusion matrix of the emotions in case of the HMM synthesizer

		Judged				
		N	A	H	S	
Target	N	71%	4%	8%	17%	N=neutral
	A	42%	33%	21%	4%	A=angry
	H	21%	0%	75%	4%	H=joyful
	S	17%	0%	0%	83%	S=sad

HMM TTS-male

In case of the HMM synthesizer, sad and joyful emotions can be expressed by the proposed procedure, but angry cannot.

5. Applicability of the results

The majority of the results can be applied directly to automatically generate speech that is of higher quality and more diversified. The statistical results can be applied in a wider area, e.g. in speech recognition as well. Although I illustrated

the applicability of the results by examples in the descriptions of the theses, I give further potential applications for each thesis group in the following paragraphs:

The diacritic reconstruction procedure of Thesis I.1 has already been applied to aid the reading of electronic mails and text messages. It can also be applied in widely available cell phones, where users frequently write diacritic-free texts once again due to the limited input capabilities of cell phones.

The results of Thesis group II can be widely applied: the extracted coverage data can be useful when localizing language technologies. The modified letter statistics may be useful in further research on speech synthesis and can help in developing test procedures.

Thesis group III is already in a practical application, but it can also be employed for high-quality information reading with similar complexity. The intensity normalization procedure for corpus-based synthesizers has also been applied, but the results can form the starting point for further studies on prosody.

The results presented in Thesis group IV can be applied when building human-machine interfaces, where it is necessary to convey emotions beyond the linguistic content. The results of the emotional manipulation can be applied with other synthesis techniques as well.

6. References

Abari K. – Olaszy G. (2006): Internetes beszédadatbázis a magyar mássalhangzó kapcsolódások akusztikai szerkezetének bemutatására. In Alexin Z. – Csendes D. (szerk.) *IV. Magyar Számítógépes Nyelvészeti Konferencia*. Szeged, 213–222.

Allen, J. – Hunnicutt, M. – Klatt, D. – Armstrong, R. – Pisoni, D. (1987): *From text to speech: The MITalk system*. London, Cambridge University Press.

Black, A. – Taylor, P. (1994): CHATR: a generic speech synthesis system. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 983–986.

Black, A. – Taylor, P. – Caley, R. (2006): The Festival Speech Synthesis System, 1994–2006, Manual and source code avaliable at `http://www.cstr.ed.ac.uk/projects/festival`.

Boersma P. – Weenink D. (2010): Praat: doing phonetics by computer [Computer program]. Version 5.1.43, retrieved 4 August 2010 from http://www.praat.org/.

Burnard, L. (1995): The BNC Users Reference Guide. *British National Corpus Consortium, Oxford, May*.

De Pauw, G. – Wagacha, P. – de Schryver, G.-M. (2007): Automatic Diacritic Restoration for Resource-Scarce Languages. In *Text, Speech and Dialogue*. Lecture Notes in Computer Science sorozat, vol. 4629. Springer Berlin / Heidelberg, 170–179.

Douglas-Cowie, E. – Campbell, N. – Cowie, R. – Roach, P. (2003): Emotional Speech: Towards a New Generation of Databases. 40. évf. *Speech Communication*, 33–60.

Fant, G. (1960): *Acoustic theory of speech production*. Mouton De Gruyter.

Gordos G. – Sándor L. T. (1985): A limited vocabulary speech synthesiser terminal. In *Proc. of the Finnish-Hungarian symposium on information technology*. Helsinki, 3–10.

Gordos G. – Takács Gy. (1983): *Digitális beszédfeldolgozás*. Budapest, Műszaki Könyvkiadó.

Hallahan, W. (1995): DECtalk software: Text-to-speech technology and implementation. 7. évf. 4. sz., *Digital Technical Journal*, 5–19.

Hamon, C. – Mouline, E. – Charpentier, F. (1989): A diphone synthesis system based on time-domain prosodic modifications of speech. In *ICASSP89*. 238–241.

Homer, D. – Ries, R. – Vatkins, S. (1939): A synthetic speaker. 227. évf. *J. Franklin Institute*, 739–764.

Kempelen F. (1969): *Az emberi beszéd mechanizmusa (Mechanismus der Menschlichen Sprache)*. Budapest (Wien), 1969 (original edition: 1791), Szépirodalmi Kiadó.

Kenesei I. – Kelemen J. – Pap M. – Pléh C. – Radics K. – Réger Z. – Rohonci K. – Szabolcsi A. (1984): *A nyelv és a nyelvek*. Akadémiai Kiadó.

Kiss G. – Olaszy G. (1984): A Hungarovox magyar nyelvű, szótár nélküli, valós idejű párbeszédes beszédszintetizáló rendszer. 2. sz., *Információ Elektronika*, 98–111.

Li, W. (1992): Random texts exhibit Zipf's-law-like word frequency distribution. 38. évf. 6. sz., *IEEE Transactions on Information Theory*, 1842–1845.

MBROLA, s. (2006): The Festival Speech Synthesis System, 1996–2006, Manual and source code avaliable at `http://tcts.fpms.ac.be/synthesis`.

Mihalcea R. – Nastase V. (2002): Letter Level Learning for Language Independent Diacrtitics Restoration. In *Proc. Computational Linguistics*. 1–7.

Nagy B. (2008): Huhyphn: magyar elválasztásiminta-gyűjtemény, `http://www.tipogral.hu/`.

Olaszi P. (2002): *Magyar nyelvű szöveg-beszéd átalakítás: nyelvi modellek, algoritmusok és megvalósításuk*. Phd disszertáció (Budapesti Műszaki és Gazdaságtudományi Egyetem).

Olaszy G. – Németh G. (1999): IVR for banking and residential telephone subscribers using stored messages combined with a new number-to-speech synthesis method. In Gardner-Bonneau D. (szerk.) *Human Factors and Voice Interactive System*. Kluwer, 237–256.

Olaszy G. (1989): *Elektronikus beszédelőállítás, A magyar beszéd akusztikája és formánsszintézise*. Műszaki Kiadó.

Olaszy G. – Kiss G. – Németh G. – Olaszi P. (2000): Profivox: a legkorszerűbb hazai beszédszintetizátor. In *Beszédkutatás'2000*. Budapest, MTA Nyelvtudományi Intézet, 167–179.

Popescu, I. (2003): On a Zipf's law extension to impact factors. 6. évf. *Glottometrics*, 83–93.

Prószéky G. (1988): Hungarian-a Special Challenge to Machine Translation? In *New directions in machine translation: conference proceedings, Budapest, 18-19 August, 1988*. 219.

Přibilová, A. – Přibil, J. (2009): Spectrum modification for emotional speech synthesis. *Multimodal Signals: Cognitive and Algorithmic Issues*, 232–241.

Quinlan, J. (1993): *C4. 5: programs for machine learning*. Morgan Kaufmann.

Rabiner, L. (1968): Digital-Formant Synthesizer for Speech-Synthesis Studies. 43. évf. *J. Acoust. Soc. Am.*, 822–828.

Rutten, P. – Aylett, M. – Fackrell, J. – Taylor, P. (2002): A statistically motivated database pruning technique for unit selection synthesis. In *Seventh International Conference on Spoken Language Processing*. ISCA, 125–128.

Rutten, P. – Fackrell, J. (2003): The application of interactive speech unit selection in TTS systems. In *Eighth European Conference on Speech Communication and Technology*. 285–288.

Scherer, K. (2003): Vocal communication of emotion: A review of research paradigms. 40. évf. 1-2. sz., *Speech communication*, 227–256.

Taylor P. (2009): *Text-to-Speech Synthesis*. Cambridge University Press.

Tokuda, K. – Yoshimura, T. – Masuko, T. – Kobayashi, T. – Kitamura, T. (2000): Speech parameter generation algorithms for HMM-based speech synthesis. In *Acoustics, Speech, and Signal Processing, ICASSP'00. Proceedings. 2000 IEEE International Conference on* , vol. 3. IEEE, 1315–1318. ISBN 0780362934.

Tóth, B. – Németh, G. (2008): Rejtett Markov-Modell Alapú Mesterséges Beszédkeltés Magyar Nyelven. LXIII. köt. 2–6.

Tóth B. – Németh G. (2009): Rejtett Markov-modell alapú szövegfelolvasó adaptációja félig spontán magyar beszéddel. In Alexin Z. – Csendes D. (szerk.) *VI. Magyar Számítógépes Nyelvészeti Konferencia*. Szeged, 213–222.

Ungurean, C. – Burileanu, D. – Popescu, V. – Negrescu, C. – Dervis, A. (2008): Automatic diacritic restoration for a TTS-based e-mail reader application. 70. évf. 4. sz., *University" Politehnica" of Bucharest Scientific Bulletin, Series C: Electrical Engineering and Computer Science*, 3–12. ISSN 1454-234X.

Váradi, T. (1999): On developing the Hungarian national corpus. In *Proceedings of the Workshop Language Technologies-Multilingual Aspects, 32nd Annual Meeting of the Societas Linguistica Europea, Ljubjana, Slovenia*. 57–63.

Zen, H. – Nose, T. – Yamagishi, J. – Sako, S. – Masuko, T. – Black, A. – Tokuda, K. (2007): The HMM-based speech synthesis system (HTS) version 2.0. In *Proc. of Sixth ISCA Workshop on Speech Synthesis*. Citeseer, 294–299.

Publications

Connected to the thesis

Patent

[P1] Zainkó Cs, Németh G, Olaszy G, Gordos G.: Eljárás magyar nyelven ékezetes betűk használata nélkül készített szövegek ékezetes betűinek visszaállítására. Lajstromszám: P 0003443
Közzététel éve: 2000 Benyújtás helye: Magyarország

Chapters in edited books

[B1] Németh G, Zainkó Cs, Kiss G, Olaszy G, Fekete L, Tóth D: Replacing a Human Agent by an Automatic Reverse Directory Service.In: Magyar G, Knapp G, Wojtkowski W, W Wojtkowski G, Zupančič J (szerk.)
Advances in Information Systems Development.Springer, 2007. pp. 321–328.

[B2] Németh Géza, Zainkó Csaba, Bogár Balázs, Szendrényi Zsolt, Olaszi Péter, Ferenczi Tibor: Elektronikus-levél felolvasó.In: Gósy M (szerk.)
Beszédkutatás 98: Beszéd, spontán beszéd, beszédkommunikáció.Budapest: MTA Nyelvtudományi Intézet, 1998. pp. 189–203.

[B3] Németh G, Zainkó Cs: Statisztikai szövegelemzés automatikus felolvasáshoz.In: Gósy M (szerk.) Beszédkutatás 2000: Beszéd és társadalom.Budapest: MTA Kiadó, 2000. pp. 156–166.
Number of independent citation: 1

[B4] Zainkó Cs, Fék M: Beszédadatbázis prozódiájának szerepe a gépi beszéd hangzásában és érzelmi tartalmak kifejezésében. Vidám avagy szomorú a beszédszintetizátor? In: Gósy M (szerk.)
Beszédkutatás 2006.Budapest: MTA Kiadó, 2006. pp. 208–217.

[B5a] Németh G, Zainkó Cs: Automatikus szám szerinti tudakozó; In: Németh G, Olaszy G (szerk.) A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Budapest: Akadémiai Kiadó, 2010. pp. 561–562.

[B5b] Zainkó Cs: Magyar hang-, betű- és szóstatisztika; Érzelmi töltetű beszéd modellezése; Elemkiválasztás-alapú szövegfelolvasó; Érzelmes szövegfelolvasás In: Németh G, Olaszy G (szerk.) A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Budapest:

Akadémiai Kiadó, 2010. pp. 86–92., 466–467., 505–512., 518–520

Journal articles

[J1] Zainkó Cs, Németh G, Bogár B, Szendrényi Zs: E-levél felolvasó. HÍRADÁSTECHNIKA 49:(11-12) pp. 61–76. (1998)

[J2] Németh G, Zainkó Cs, Fekete L, Olaszy G, Endrédi G, Olaszi P, Kiss G, Kiss P: The design, implementation and operation of a Hungarian e-mail reader. INTERNATIONAL JOURNAL OF SPEECH TECHNOLOGY 3-4: pp. 216–228. (2000)
Number of independent citation: 1

[J3] Németh G, Zainkó Cs, Fekete L: Statistical analysis used for e-mail reader development and enhancement. HÍRADÁSTECHNIKA LVI:(4) pp. 29–36. (2001)

[J4] Németh G, Zainkó Cs, Fekete L: Statisztikai elemzések felhasználása e-levél felolvasó kialakításában és továbbfejlesztésében. HÍRADÁSTECHNIKA LVI:(1) pp. 23–30. (Pollák–Virág díjas) (2001)

[J5] Németh G, Zainkó Cs: Multilingual Statistical Text Analysis, Zipf's Law and Hungarian Speech Generation. ACTA LINGUISTICA HUNGARICA 49:(3-4) pp. 385–405. (2002)
Number of independent citations: 2

[J6] Zainkó Csaba: Magyar nyelvű, kötött témájú korpusz-alapú beszédszintézis.: és a kötetlenség felé vezető út vizsgálata. HÍRADÁSTECHNIKA LXIII:(5) pp. 12–17. (2008)

Papers in conference proceedings

[C1] Németh G, Zainkó Cs, Olaszy G, Prószéky G: Problems of creating a flexible e-mail reader for Hungarian. In: European Conference on Speech Communication and Technology (Eurospeech 1999). Budapest, Magyarország, 1999.09.05-1999.09.09.(2) Budapest: pp. 939–942.
Number of independent citations: 4

[C2] Németh G, Zainkó Cs: Word Unit Based Multilingual Comparative Analysis of Text Corpora. In: Paul Dalsgaard, Borge Lindberg, Henrik Benner, Zheng-hua Tan (szerk.)

European Conference on Speech Communication and Technology (Eurospeech 2001). Aalborg, Dánia, 2001.09.03-2001.09.07.Aalborg: pp. 2035–2038.
Number of independent citations: 3

[C3] Zainkó Cs, Németh G: Statistical Text Processing for Automatic Synthesis of Speech. In: EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services (ECMCS2001). Budapest, Magyarország, 2001.09.11-2001.09.13.Budapest: pp. 301–304.

[C4] Németh G, Zainkó Cs, Kiss G, Fék M, Olaszy G, Gordos G: Language Processing for Name and Address Reading in Hungarian. In: IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE 2003). Beijing, Kína, 2003.10.26-2003.10.29.Beijing: pp. 238–243.(ISBN: 0-7803-7902-0)

[C5] Fék Márk, Zainkó Csaba, Németh Géza: Érzelmes beszéd gépi előállítása érzelem specifikus beszédadatbázisok felhasználásával. In: Alexin Zoltán, Csendes Dóra (szerk.)
Magyar Számítógépes Nyelvészeti Konferencia. Szeged, Magyarország, 2007.12.06-2007.12.07.Szeged: Szegedi Tudományegyetem Informatikai Tanszékcsoport, pp. 34–43.

[C6] Németh G, Zainkó Cs, Fék M, Olaszy G, Bartalis M: Promptgenerátor - Ügyfélszolgálati hangos üzenetek automatikus gépi előállítása egy adott bemondó hangjára. In: Alexin Zoltán, Csendes Dóra (szerk.)
Magyar Számítógépes Nyelvészeti Konferencia. Szeged, Magyarország, 2007.12.06-2007.12.07.Szeged: Szegedi Tudományegyetem Informatikai Tanszékcsoport, pp. 3–11.

[C7] Géza Németh, Csaba Zainkó, Mátyás Bartalis, Gábor Olaszy, Géza Kiss: Human Voice or Prompt Generation? Can They Co-Exist in an Application? In: Interspeech 2009: Speech and Intelligence. Brighton, Nagy-Britannia, 2009.09.06-2009.09.10.ISCA, pp. 620–623.

[C8] Zainkó Csaba: A magyar nyelv betűstatisztikája beszédfeldolgozási szempontok figyelembevételével. In: VI. MAGYAR SZÁMÍTÓGÉPES NYELVÉSZETI KONFERENCIA:. Szeged, Magyarország, 2009.12.03-2009.12.04.Szeged: pp. 238–245.

[C9] Csaba Zainkó , Márk Fék, Géza Németh: Expressive Speech Synthesis Using Emotion-Specific Speech Inventories. LECTURE NOTES IN COMPUTER SCIENCE 5042, Proc. of COST 2102: pp. 225–234. Paper 17. (2008)
Number of independent citation: 1

[C10] Csaba Zainkó , Tamás Gábor Csapó, Géza Németh: Special Speech Synthesis for Social Network Websites. LECTURE NOTES IN COMPUTER SCIENCE 6231, Proc. of TSD 2010: pp. 455–463. (2010)

Conference presentation

[C11] Csaba Zainkó , Géza Németh: Emotional modification for verbal communication. In: The PINK COST 2102 International Conference on Analysis of Verbal and Nonverbal Communication and Enactment: The Processing Issues Budapest, Sep. 7–10, (2010)

Other publications

Chapters in edited books

[B5c] Németh G, Zainkó Cs: Telefonról elérhető e-levél felolvasó; In: Németh G, Olaszy G (szerk.) A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Budapest: Akadémiai Kiadó, 2010. pp. 555–557.

[B5d] Zainkó Cs, Bartalis M, Németh G: Automatikus áru- és árlista-felolvasó In: Németh G, Olaszy G (szerk.) A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Budapest: Akadémiai Kiadó, 2010. pp. 569–573.

[B5e] Zainkó Cs, Németh G: Ékezetek gépi helyreállítása; SMS-felolvasó vezetékes telefonra; Időjárás-előrejelzés írott szöveges és hangos modalitással; Vasútállomási utastájékoztató In: Németh G, Olaszy G (szerk.) A MAGYAR BESZÉD; Beszédkutatás, beszédtechnológia, beszédinformációs rendszerek. Budapest: Akadémiai Kiadó, 2010. pp. 485–488., 557–560., 575–576., 579

[B6] Németh G, Olaszy G, Bartalis M, Kiss G, Zainkó Cs, Mihajlik P, Haraszti Cs: Beszédkommunikáció az ember és a gép között. In: Talyigás Judit (szerk.) Mozaikok a hazai telematika eredményeiből.Budapest: Hírközlési és Informatikai Tudományos Egyesület, 2007. pp. 37–52.

[B7] Németh G, Olaszy G, Bartalis M, Kiss G, Zainkó Cs, Mihajlik P, Haraszti Cs: Automated Drug Information System for Aged and Visually Impaired Persons. In: Miesenberger K, Klaus J, Zagler W, Karshmer A (szerk.) Computers Helping People with Special Needs.Springer-Verlag, 2008. pp. 238–241.

[B8] Zainkó Cs, Németh G: Az automatikus SMS-felolvasás problémai. In: Gósy Mária (szerk.)
Beszédkutatás 2002: Kísérleti beszédkutatás.Budapest: MTA Nyelvtudományi Intézet, 2002. pp. 197–211.

[B9] Németh G, Kiss G, Zainko Cs, Olaszy G, Tóth B: Speech Generation in Mobile Phones. In: Gardner-Bonneau D, Blanchard H. (szerk.)
Human Factors and Interactive Voice Response Systems: Speech Generation in Mobile Phones. Springer, 2008. pp. 163–191.
Number of independent citation: 1

Journal articles

[J7] Olaszy G, Németh G, Olaszi P, Kiss G, Zainkó Cs, Gordos G: Profivox - a Hungarian TTS System for Telecommunications Applications. INTERNATIONAL JOURNAL OF SPEECH TECHNOLOGY 3–4: pp. 201–215. (2000)
Number of independent citations: 7

[J8] Fék M, Pesti P, Németh G, Zainkó Cs: Generációváltás a beszédszintézisben. HÍRADÁSTECHNIKA LXI:(3) pp. 21–30. (2006)

[J9] Olaszy G, Németh G, Bartalis M, Kiss G, Zainkó Cs, Fegyó T, Árvay G, Szepezdi Zs, Terplánné Balogh M: Kísérleti gyógyszerinformációs rendszer beszédmodulokkal. HÍRADÁSTECHNIKA LXI:(3) pp. 8–13. (2006)

[J10] Németh Géza, Olaszy Gábor, Bartalis Mátyás, Zainkó Csaba, Fék Márk, Mihajlik Péter: Beszédadatbázisok előkészítése kutatási és fejlesztési célok hatékonyabb támogatására. HÍRADÁSTECHNIKA LXIII:(5) pp. 18–24. (2008)

[J11] Tamás Gábor Csapó, Csaba Zainkó, Géza Németh: A Study of Prosodic Variability Methods in a Corpus-Based Unit Selection Text-To-Speech System. INFOCOMMUNICATIONS JOURNAL LXV:(1) pp. 32–37. (2010)

Papers in conference proceedings

[C12] Fék M, Pesti P, Németh G, Zainkó Cs, Olaszy G: Corpus-Based Unit Selection TTS for Hungarian. LECTURE NOTES IN COMPUTER SCIENCE 4188, Proc. of TSD 2006: pp. 367–373. (2006)
Number of independent citations: 2

[C13] Abari K, Olaszy G, Kiss G, Zainkó Cs: Magyar kiejtési szótár az Interneten. In: Alexin Zoltán, Csendes Dóra (szerk.)
Magyar Számítógépes Nyelvészeti Konferencia. Szeged, Magyarország, 2006.12.07-2006.12.08.Szegedi Tudományegyetem Informatikai Tanszékcsoport, pp. 223–230.

[C14] G Németh, G Olaszy, M Bartalis, G Kiss, Cs Zainkó, P Mihajlik: Speech based Drug Information System for Aged and Visually Impaired Persons. In: Interspeech 2007 - Eurospeech: 9th European Conference on Speech Communication and Technology. Antwerpen, Belgium, 2007.08.27-2007.08.31.ISCA, pp. 2533–2536.