Creation and analysis of a Polish speech database for use in unit selection synthesis

Dominika Oliver[*], **Krzysztof Szklanny**[†]

[*]Institute of Phonetics, Saarland University
FR 4.7 Phonetics, Building 17.2, Room 4.10, 66123 Saarbrücken, Germany
dominika@coli.uni-sb.de
[†] Multimedia Department, Polish-Japanese Institute of Information Technology
Koszykowa 86, 02-008 Warsaw, Poland
kszklanny@pjwstk.edu.pl

Abstract

The main aim of this study is to describe the process of creating a speech database to be used in corpus based text-to-speech synthesis. To help achieve natural sounding speech synthesis, the database construction was aimed at rich phonetic and prosodic coverage based on variable length units (phoneme, diphone, triphone) from different phonetic and prosodic contexts. Following previous work on determining the optimal coverage (Szklanny and Oliver, 2005), text selection was based on the existing text corpus containing parliamentary statements. Corpus balancing was followed by recording of the material. Automatic segmentation was performed, followed by both an automatic and manual check of the data to determine speaker specific phenomena and correct the labelling. Additionally, prosodic annotation involving assignment of the intonation contours was performed in order to assess the accent realisation and determine the prosodic coverage of the database. The prototype speech synthesiser was built to determine the validity of the above steps and test the resulting voice quality.

1. Introduction

The process of creating a speech database for unit selection concatenative speech synthesis is a time-consuming task. Manually designing the corpus is, in practice, only applicable in limited domain speech synthesis and recognition systems. The sentence selection tools used while designing the corpus are usually based on a greedy algorithm. The bigger the text set, the better the chance to fulfil given criteria.

Earlier experience with preparing diphone databases for speech synthesis and using it in concatenative synthesis (Santen and Buchsbaum, 1997) confirms that diphones help obtain natural sounding speech. As for triphones, they can also be easily concatenated, but obtaining a full coverage for triphones is impractical because of the huge number of triphones (Villasenor-Pineda et al., 2003).

It has to be noted that in creating a unit selection text-to-speech system there is a great need to produce a speech database which would adequately cover both the phonetic segments and prosodic events existing in a language in a variety of contexts. These preparatory tasks can be time consuming but the quality of the resulting speech synthesis system relies heavily on them. After the appropriate sentences have been chosen and their correct phonetic transcription prepared, they have to be recorded and subsequently segmented and annotated. Segmentation of the recorded speech is another task which has an influence on the quality of the generated speech (Adell and Bonafonte, 2004).

The main aim of this study is to design a speech corpus for Polish Unit Selection Speech Synthesis on the basis of unit frequency distribution and to determine its validity through a prototype TTS application.

The paper will describe how the database was created, including the discussion of the issues encountered while creating the database. It is organised according to the steps involved in creating a speech database for a synthetic voice:

1. Designing the speech database
   - Balancing the database
   - Grapheme to phoneme conversion
   - Text type selection
2. Recording the prompts for the speech database
   - Speaker description and recording procedure
   - Quality issues
3. Automatic segmentation of the prompts
4. Verification of the recorded and labelled material
   - Phonetic check of the recordings
   - Automatic and manual verification of auto-labelling
   - Prosodic annotation
5. Building a prototype synthesizer for testing and tuning

2. Designing the speech database

2.1. Balancing the database

The size of the initial corpus comprising parliamentary statements used for sentence selection was 300 MB, which corresponded to 5778460 sentences. The necessary preprocessing of these sentences included the removal of all the tags and other meta data. Abbreviations, acronyms and number forms were manually expanded. Next, all sentences in graphemic form had to be transformed into their phonetic transcription to enable a phonetic balancing process.

2.2. Grapheme to phoneme conversion

Phonetic transcription according to Polish Sampa was automatically generated for each of the sentences. Two different algorithms for grapheme to phoneme conversion (statistical and rule based) were used and their output compared for consistency. The rule based approach used expert written rewrite rules for the Festival system (Black and Taylor, 1998) and the automatic method used decision trees (C5.0).

2.3. Text type selection

Statistical analysis comparing the corpus containing parliamentary statements to sources from other domains (newspaper reviews) has shown that despite the difference in domain and size ratio (10:1), neither of the data sources differed significantly as far as the relative frequency of phonemes present was concerned.

The balancing *proper* of the phoneme, diphone and triphone form of the textual corpus was then carried out using a greedy algorithm.

An example input sentence in our initial corpus is in its orthographic and phonetic form represented by a) orthography b) phonemes c) diphones and d) triphones.

a jeśli chodzi o utrzymanie infrastruktury szacuje się potrzeby roczne

b j e s’ l i x o dz’ i o u t S I m a n’ e i n f r a s t r u k t u r I S a ts u j e s’ e p o t S e b I r o tS n e

c j je es’ s’l li ix xo odz’ dz’i io ou ut tS SI Im ma an’ n’e ei in nf fr ra as st tr ru uk kt tu ur rI IS Sa ats tsu uj je es’ s’e e p po ot tS Se eb bI Ir ro otS tSn ne e

d je jes’ es’l s’li lix ixo xodz’ odz’i dz’io iou out utS tSI SIm Ima man’ an’e n’ei ein inf nfr fra ras ast str tru ruk ukt ktu tur urI rIS ISa Sats atsu tsuj uje jes’ es’e s’e p e po pot otS tSe Seb ebI bIr Iro rotS otSn tSne ne

2.4. Greedy algorithm

The CorpusCrt program (Bailador, 1998) was used as a corpus balancing tool for sentence selection. The following criteria were used:

1. The minimum phonetic length of a sentence is 30 phonemes
2. The maximum phonetic length of a sentence is 80 phonemes
3. The output corpus should contain 2500 sentences
4. Each phoneme should occur at least 40 times in the corpus
5. Each diphone should occur at least 4 times in the corpus
6. Each triphone should occur at least 3 times

These requirements were inputted to the greedy algorithm program and twelve different versions of balanced corpora with 2500 sentences each were created.

After two-step balancing we obtained an increase in diphone number (148479 vs. 150814), a reduction in the number of diphones appearing less than four times from 175 to 68, and an increase in the number of different diphones from 1096 to 1196. The process also increased the triphone number from 145979 to 148314 and gave an increased number of different triphones from 11524 to 13832. The final text corpus of 2150 sentences was composed of statements, questions and exclamations and also enriched with rare words. Its phonetic distribution is shown in Figure 1.

Sentences selected with this method had to be manually verified in order to eliminate any markers, abbreviations and acronyms which were not expanded in initial preprocessing. The sentences selected by the greedy algorithm were also manually checked to ensure that they did not contain material which would be too hard to pronounce or contains obscene or otherwise loaded material which would introduce an emotional bias to the recordings.

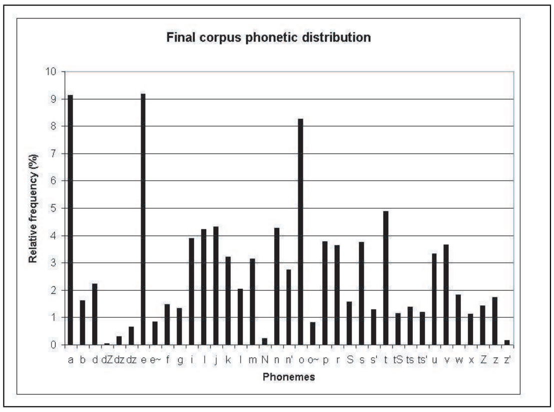


Figure 1: Phonetic distribution in the final corpus

3. Recordings

3.1. Speaker

The next step in the database creation was the recording of the prompts using a Polish voice talent. It was recorded by a semi- professional male speaker. The speaker had enough time to get familiarised with the text to be recorded. The speaker was also familiar with phonetic transcription of Polish language. During the recordings phonetic transcription was displayed to the speaker and the person supervising the recordings. This ensured the prompts could be corrected to reflect the desired pronunciation, especially in case of foreign language words.

3.2. Procedure

The recordings took place in an anechoic chamber in the Multimedia Department of the Polish-Japanese Institute of Information Technology, Warsaw using one table stand dynamic microphone (Rode NT1000). A 48 kHz sampling frequency and 16 bit resolution was used as a 48 kHz signal can easy be re-sampled to 16 kHz or 8 kHz, the most often used frequencies in TTS systems. Quality tests showed that the M-Audio Transit sound card 24 bit, 96kHz audio interface would be appropriate for the task. Closed headphones (Philips) were used to check a sample of the initial recordings to check sound level and quality, special attention was paid to background sounds, echoes and reverberation. Open headphones were also used during recording to monitor the input and to share and review the recordings with participants. The database was recorded using digital audio interface. Due to technical constraints, the process was split into sessions lasting between one to four hours, recorded over a period of one month. To achieve consistency a record was kept of the microphone used, location and its orientation. The final speech database consists of 2150 sentences, varying in length from 2.3 to 13.4 sec, with an average length of 6.3 sec.

3.3. Quality issues

For any system relying on speech, it is important to record the best possible sound quality. The quality of a recording is very important as every minor distortion can end up appearing very often in synthesised speech if it belongs to frequent speech units used in the language. Fewer reverberations and distortions in recorded speech also make the signal manipulation less demanding (van Santen, 2004).

During the recordings we encountered technical problems, namely, 25% of prompts had to be re-recorded due to DC distortions, DC component being added to the waveform, causing saturation and clipping, see Figure 2.

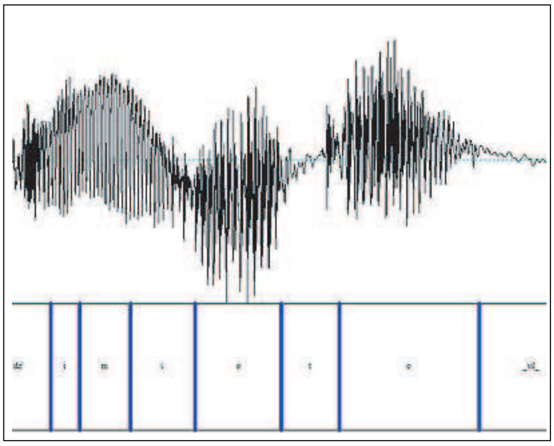


Figure 2: DC distortions

Recording environments are never silent, even when no one is present. There is always some environmental sound, for example wall creaks, air conditioner or heating unit. The recordings in this study were conducted during the academic year. The recording studio used is located in the close proximity of class-rooms and this affected the quality of the signal. During the sessions, additional distortion included chair noise, hand-dryer noise, and a harmonic frequency in signal about 400Hz - 500 Hz. Due to these distortions, some prompts had to re-recorded.

Finally 2150 sentences and 100 rare polish words were recorded. The speech database file was 1,2 GB in size. To obtain natural sounding speech synthesis, paralinguistic elements like murmuring, laughing, and coughing were also recorded.

A post-recording signal check including de-clicking and de-noising, channel balancing, normalising, frequency range equalisation and DC offset removal was also carried out. The shift itself means nothing, but any further file processing will be done incorrectly and result in a distortion if the DC offset is not removed.

4. Automatic Segmentation

The signal was automatically aligned with the transcription using an HMM based model trained using the HTK Toolkit (Young, 1994). The mixed Gaussian model was built on 500 words, 25 ms analysis window with 10 ms frame period . Evaluation of the performance of the model based on 120 phrases resulted in 98% recognition rate for phones and 90% in the case of phrases.

Automatic segmentation is a very helpful tool while creating TTS. Until now, the highest results have been achieved by manually processing the corpus. Some researchers claim that actual automatic methods for voice segmentation can already achieve accurate enough results for its use in concatenative speech synthesis. They support this claim on perceptual evaluation of the systems. However, the influence of phone segmentation in the naturalness and intelligibility of the speech depends on the philosophy of each system. It would affect it in different manner if we use different units to concatenate (Adell and Bonafonte, 2004). As the unit selection speech synthesis is based on concatenating different length units a manual correction is required.

The most studied method of segmentation is based on the speech recognition paradigm. A Hidden Markov Model can be used to perform a recognition task (Adell and Bonafonte, 2004). Automatic segmentation is also known as alignment. Aligner requires as input orthographic text and sound files. It tries to find the boundaries of phonemes, knowing exactly what was said by the speaker. The correct transcription is necessary, otherwise one gets errors in alignment. There are 37 phonemes in the Polish language. The aligner, created using HTK Toolkit (Young, 1994), contained 38 models, as a model of silence was added.

Each model consisted of three states which corresponded to: initial sound, mid sound and a final sound of the phoneme (Zhang et al., 2004).

Three structures of HMM were constructed:

1. 10 ms frame period and 25 ms analysis window
2. 5 ms frame period and 15 ms analysis window
3. 1 ms frame period and 5 ms analysis window

Each HMM consists of 39 coefficients, namely: 13 MFCC plus 13 delta coefficients plus 13 acceleration coefficients.

The initial training of speaker dependent HMMs gives incorrect estimation of phoneme boundary (Kim and Syrdal, 2004). HMMs were trained on 585 sound files. These files were phonetically balanced and consisted of words and phrases. The Gaussian Mixture was added to each state of HMM and then a re-estimation was prepared. The procedure was repeated three times.

Next, the overall estimation was conducted. HMMs were estimated on recordings of 40 speakers, where each speaker recorded 585 set composed of words and phrases.

To choose the best structure of HMM, a special test was prepared. The aim of the test was to obtain the highest score of recognition level. The test contained 125 phrases from computer domain.

Figure 3 illustrates the comparison between 10 ms frame period and 5 ms frame period model. The top tier shows 10 ms frame period model which is more accurate and the bottom tier the 5 ms model.

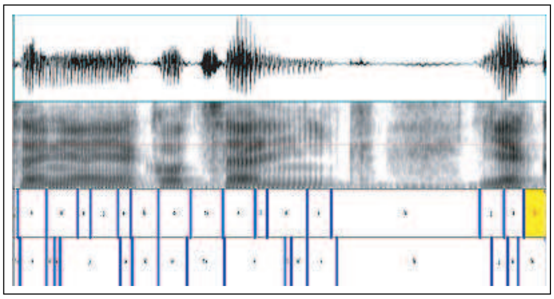


Figure 3: Comparison between 10 ms frame period and 5 ms frame period model

The best result, namely 93.27% of correctly recognized words, was achieved using a 10 ms frame period model and a 25 ms analysis window. The initial segmentation was made using a 10 ms frame period and a 25ms analysis window. Due to a large number of wrongly labelled phonemes, it was decided to run the estimation of HMMs on a segmented database. The prepared test was re-run and 92.95% of correctly recognized words was achieved.

Although the correctness of recognition has fallen by 0.32%, as compared to the other models used, we decided to use these models because the number of wrongly set boundaries in the database has fallen overall.

We also constructed the diphone models. There are 1443 diphones in the Polish language, out of which 1100 are most frequent. The training and estimation was made on the Speecon database (Marasek and Gubrynowicz, 2004), which contains recordings of 600 speakers, each recorded one hour of speech.

The accuracy of correctly recognized word was about 72%. Table 4. illustrates a comparison between different structures of HMMs.

5. Verification procedure

Unit selection systems are highly sensitive to the accuracy of phonetic labelling. Thus upon completing the recordings, a series of necessary checks including a manual correction of phonetic and graphemic transcription were performed. First, a phonetic check was carried out by a phonetic expert to verify the transcription against the speaker's realisation. The goal was to adapt the transcription to what the voice talent produced during the recording sessions.

Second, a manual check of phoneme boundaries was carried out. Wrong labelling by an automatic aligner can affect the quality of generated speech in a number of ways. If a boundary of a phoneme is inaccurately placed not only will a phonetically incorrect unit be chosen and as a result a wrong target word may be produced but also a particular word be said with an undesired accent (Kominek and Black, 2004).

The most frequent error of an aligner is a misalignment in the position of phoneme, which means that the boundary of one phoneme is placed well into the neighbouring segments. This is why a manual correction is always needed. To help reduce the extend of the manual check we also include automatic diagnostics described below to target specific problems like durational outliers,

5.1. Automatic labelling error detection

Based on the results of the speech alignment evaluation, an automatic procedure for detecting durational outliers was implemented. A tool was developed to target misaligned automatic labels based on durational outliers. We classified an alignment as a segmentation error if any phone duration was more than 2SD from the mean duration for that phone in the database.

As output we generated a list of potential misalignments, which would be taken into account during the manual check. The tool detected on average two phones per sentence to have an abnormal duration.

5.2. Manual check

The manual correction was made using Praat (Boersma, 2001) speech analysis program. Each sentence was listened to. The word or phrase which was incorrectly pronounced was either corrected or selected as the prompt to be recorded once more. The phonetic transcription of each prompt was corrected manually. The errors in phonetic transcription proved to greatly influence the automatic aligner. Figure 4 illustrates the effect of wrong phonetic transcription and its influence on alignment (top annotation tier). On the second tier we show manually corrected boundaries.

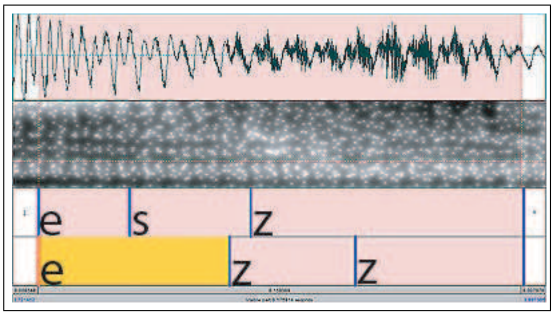


Figure 4: Effect of wrong alignment

The most frequent errors of the automatic aligner were as follows:

- Plosive phonemes especially voiceless were too short, they began to early, usually started in the second part of the previous phoneme;

- Affricate phonemes e.g. /ts'/ /tS/ were in most cases too long;

- Nasal geminates were modelled either too short or too long;

- Approximant /j/ in the neighbourhood of vowels was either too long or too short;

Model	Recognised words (%)	Recognised phrases (%)
1 ms frame period 5 ms analysis window	38,14	33,06
5 ms frame period 15 ms analysis window	71,47	55,65
10 ms frame period 25 ms analysis window	93,27	89,52
10 ms frame period 25 ms analysis window estimated on Unit Sel. Database	**92,95**	**89,52**
10 ms frame period 25 ms analysis window (diphone model)	71,79	53,23

Table 1: Comparison between different structures of HMMs.

- In case of vowel + fricative (/s/, /z/, /s'/, /S/,/Z/) part of the vowel was labelled as a fricative;
- The between words silence was not taken into account;

In Figure 5 the phoneme /f/ also includes the silence which should have been labelled as "_sil_" . In the bottom tier we mark the correctly set boundary.

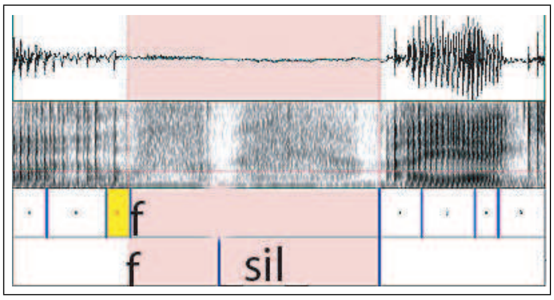


Figure 5: Missing silence

Figure 6 illustrates the incorrectly aligned boundary between phonemes /z'/ and /dz'/. In the bottom tier we mark the manually corrected boundaries.

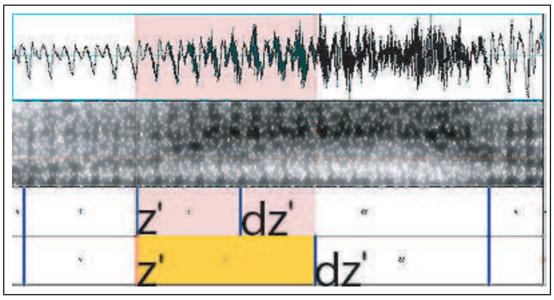


Figure 6: The incorrectly labelled phoneme z' and dz'.

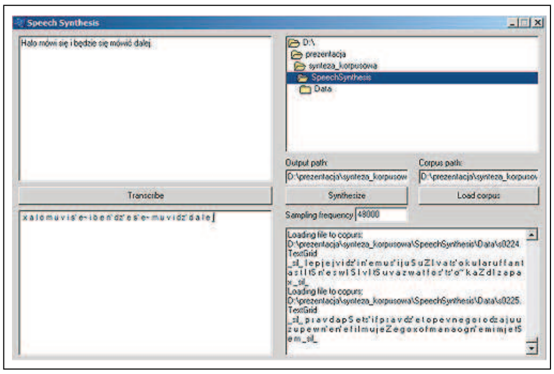
5.3. Prosodic Annotation

After performing the automatic and manual phonetic checks, the database signal was stylised and the Insint (Hirst, 1999) transcription system for annotation of the intonation patterns was applied to annotate it with prosodic labels. Additional prosodic annotation was made on separate files for phrase breaks.

The analysis shows that our speech base is prosodically rich and Insint annotation adequate even if in the future other kinds of annotations, such as ToBI (Silverman et al., 1992), should be derived.

6. Prototype

To verify the quality and segmentation of the database a Unit Selection a prototype synthesis engine was developed. Written in C++, it is a simple speech synthesizer based on concatenation of the longest possible phonetic units, see Figure 7.



Halo mówi się i będzie się mówić dalej

x a l o m u v i s' e - i b e n' dz' e s' e - m u v i dz' d a l e j

Output path: D:\prezentacja\synteza_korpusow

Corpus path: D:\prezentacja\synteza_korpusow

Sampling frequency 48000

```
Loading file to copurs:
D:\prezentacja\synteza_korpusowa\SpeechSynthesis\Data\s0224.
TextGrid
_sil_ l e p j e j v i dz' i n' e m u s' i j u S u Z I v a ts' o k u l a r u f f a n t
a s I I tS n' e s w I S I v I tS u v a z w a t f o s' ts' o~ k a Z d I z a p a
x _sil_
Loading file to copurs:
D:\prezentacja\synteza_korpusowa\SpeechSynthesis\Data\s0225.
TextGrid
_sil_ p r a v d a p S e ts' i f p r a v dz' e t o p e v n e g o r o dz a j u u
z u p e w n' e n' e f i l m u j e Z e g o x o f m a n a o g n' e m i m j e tS
e m _sil_
```

Figure 7: A prototype synthesiser.

The aim of this was to check the quality of voice and also the quality of the automatic segmentation.

The prototype engine includes a phonetic transcription module, which was based on the regular expressions. The whole speech database in the form of textgrids and sound files is loaded into the program.

First the synthesizer converts the orthographic text into a sequence of phonemes. An algorithm builds the list of all phoneme n-grams that match the input text. Next, all transition paths are constructed between the n-grams. Only the paths containing appropriate phonetic sequences are considered. The result is one path, the one with the lowest cost. By the lowest cost we mean the smallest number of acoustic units, lowest number of n-grams.

Because the algorithm analyses all the paths its speed is logarithmic. To make it linear, the following heuristics is used. A sequence of phonemes which has to be synthesized is divided into parts composed of at most twenty n-grams. Each of these sequences of n-grams is analysed individually and their paths are joined. Additionally, the algorithm joining the paths favours the biggest n-grams.

During a test, 100 random sentences from the list of corpora sentences not used as a recording material in a speech base was chosen. During tests using the prototype it can be observed how the quality of generated speech changes after the manual verification of the database. It also enables correction of errors not detected manually.

7. Conclusions

By addressing the above issues in the manner indicated and creating dedicated tools we hope to have produced a phonetically and prosodically annotated speech base adequate for the process of creating a unit selection voice. We aim to demonstrate the impact of proper diagnostics and tuning on the resulting quality of the synthesised voice in a subsequent study.

8. Acknowledgements

The authors would like to thank prof. Krzysztof Marasek for supplying the grapheme to phoneme converter, Danijel Korzinek and Łukasz Brocki for constructing the prototype synthesis program and Michał Wójtowski for help with correcting the automatic segmentation.

9. References

Jordi Adell and Antonio Bonafonte. 2004. Towards phone segmentation for concatenative speech synthesis. In *Fifth ISCA ITRW on Speech Synthesis (SSW5)*, Pittsburgh, PA, USA.

Alberto Sesma Bailador. 1998. CorpusCrt. Technical report, Polytechnic University of Catalonia (UPC).

A. Black and P. Taylor. 1998. Festival Speech Synthesis System: system documentation. Technical Report HCRC/TR–83, University of Edinburgh, Human Communication Research Centre.

Paul Boersma. 2001. Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345.

Daniel Hirst. 1999. The symbolic coding of segmental duration and tonal alignment: an extension to the INTSINT system. In *In Proceedings of EUROSPEECH'99*, pages 1639–1642.

Yeon-Jun Kim and Ann Syrdal. 2004. Improving tts by higher agreement between predicted versus observed pronunciations. In *Fifth ISCA ITRW on Speech Synthesis (SSW5)*, Pittsburgh, PA, USA.

John Kominek and Alan W. Black. 2004. Impact of durational outlier removal from unit selection catalogs. In *Fifth ISCA ITRW on Speech Synthesis (SSW5)*, Pittsburgh, PA, USA.

Krzysztof Marasek and Ryszard Gubrynowicz. 2004. Multi-level Annotation in SpeeCon Polish Speech Database. In *IMTCI*, pages 58–67.

Jan P. H. Van Santen and Adam L. Buchsbaum. 1997. Methods for Optimal Text Selection. In *Proc. 5th Euro. Conf. on Speech Communication and Technology (EUROSPEECH-97)*, pages 553–6, Rhodes, Greece.

K. Silverman, M. E. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. 1992. ToBI: a standard for labelling English prosody. In *In Proceedings of ICSLP 1992*, volume 2.2, pages 867–870.

Krzysztof Szklanny and Dominika Oliver. 2005. Corpus creation for polish unit selection speech synthesis. In *In Proceedings of Speech Analysis, Synthesis and Recognition: Applications of Phonetics, SASR 2005*, Cracow.

Esther Klabbers Jan P. H. van Santen. 2004. Clustering of foot-based pitch contours in expressive speech. In *Fifth ISCA ITRW on Speech Synthesis (SSW5)*, Pittsburgh, PA, USA.

L. Villasenor-Pineda, M. Montes y Gómez, M. A. Pérez-Coutino, and D. Vaufreydaz. 2003. A Corpus Balancing Method for Language Model Construction. In *Computational Linguistics and Intelligent Text Processing, 4th International Conference, CICLing*, pages 393–401, Mexico City, Mexico.

S. J. Young. 1994. The HTK Hidden Markov Model Toolkit: Design and Philosophy. Technical Report CUED/F-INFENG/TR.152, Cambridge University.

Jason Y. Zhang, Arthur R. Toth, Kevyn Collins-Thompson, and Alan W Black. 2004. Prominence prediction for super-sentential prosodic modeling based on a new database. In *Fifth ISCA ITRW on Speech Synthesis (SSW5)*, Pittsburgh, PA, USA.