

Design of Optimal Slovenian Speech Corpus for Use in the Concatenative Speech Synthesis System

Matej Rojc, Zdravko Kačič

Faculty of Electrical Engineering and Computer Science, University of Maribor
Smetanova 17, 2000 Maribor
matej.rojc@uni-mb.si, kacic@uni-mb.si

Abstract:

In the paper the development of Slovenian speech corpus for use in concatenative speech synthesis system being developed at University of Maribor, Slovenia, will be presented. The emphasis in the paper is the issue of maximising the usefulness of the defined speech corpus for concatenation purposes. Usefulness of the speech corpus very much depends on the corresponding text and can be increased if the appropriate text is chosen. In the approach we used, detailed statistics of the text corpora has been done, to be able to define the sentences, rich with non-uniform units like monophones, diphones and triphones.

1. Introduction

The modern approaches in speech synthesis and prosodic analysis require longer units of speech, to enable analysis of features such as turn-taking, variation in speaking style, and paragraph-level prosodic characteristics. Obviously realistic-sounding machine generated speech can be created if raw waveforms segments are directly concatenated without any signal processing modifying their prosody. But the cost of such approach is that the source corpus must be very large to consist of all the possible basic speech units in all contexts.

Basically the corpus should have all possible units inside. This is possible in case of monophones and diphones, but the number of all possible triphones (e.g. 45^3 in case of 45 phonemes) is already too big if we are dealing with limited speech source corpus.

In our approach we decided to define the text in such a way that it will have as many different monophones, diphones and triphones without duplications as possible. To achieve that, some statistic on the available text corpora has to be performed.

First, how the needed resources for statistical analysis and final text corpus were obtained, will be given. Then data preparation step will be described. In the following section the architecture of the system for performing statistical analysis will be described in more details. In the end the statistical results will be presented and conclusion drawn.

2. Text corpora

Nowadays there are a lot of text corpora resources available in electronic form (e.g. CD-ROMs, Internet). In our work most of the corpora to be analysed for build-up of appropriate text corpus, rich with different

monophones, diphones and triphones, were available on CD-ROMs with newspaper articles or were downloaded from the internet in various formats t (also texts available from the literature). The whole text corpora consisted of about 31 million words. After conversion into text format, the automatic segmentation of input text into sentences was performed. The goal of the text selection was a set of 1200 sentences where each sentence has more than 15 words and less than 25 words. According to the predefined specifications for final text corpus, all to short or too long sentences were eliminated. Before any statistic analysis was performed, four different text corpora were generated each consisting of about 5000 sentences.

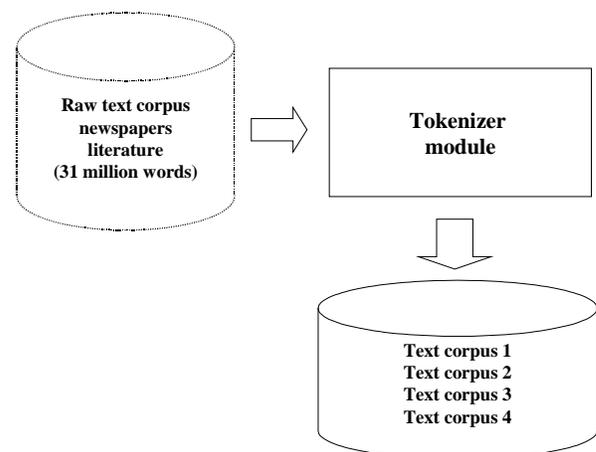


Figure 1. Data preparation step

2.1. The tokenization of text using tokenization module

As we can see on Figure 1, the input text corpus (raw ASCII text) was fed to a tokenizer engine (finite-state machine) (Rojc,1999), which emits hypotheses about tokens and segments the input text into words.

The tokenizer engine is multilevel organised. At the lowest level tokens are determined by lexical scanner. Then next text processing level follows, able to insert possible end of sentence token. The tokens separated by the lexical scanner may not be available in the dictionary form, appropriate for grapheme-to-phoneme conversion. In this case normalisation text processing level breaks such tokens into corresponding words. All tokens like date, hour, time, cardinal and ordinal numbers are expanded into corresponding word forms during tokenization process at the 'expand text processing level'. The text corpus obtained through the tokenization process was then partitioned in four parts.

3. Architecture of the system for performing statistical analysis

The system architecture consists of two basic levels: the analysis level, where the detailed statistic of input text is performed, and the generation level, where the final text is chosen, based on the results of statistical analysis (Figure 2).

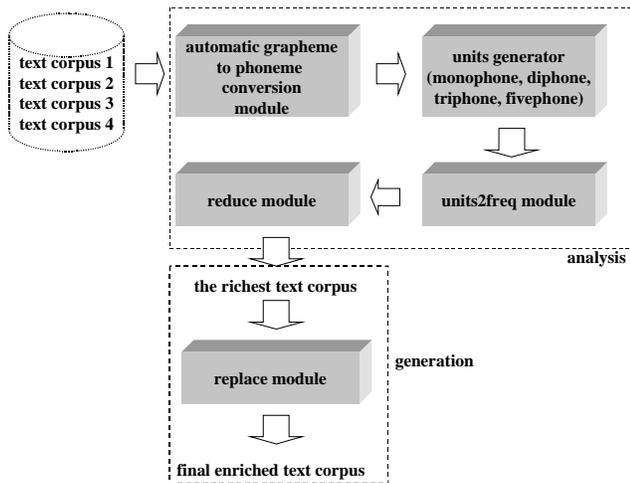


Figure 2. Architecture of the system for performing statistical analysis

The analysis level contains four modules: automatic grapheme-to-phoneme conversion module, unit's generator and module for assigning frequency to each word in the corpora and reduce module.

The generation level generates final text corpus using the statistical results obtained during analysis level. At this stage the corpus should contain the maximum possible number of different units (monophones, diphones, triphones and fivephones). The level consists only of 'replace module'.

In the following all modules will be described in more details.

3.1. Automatic grapheme-to-phoneme conversion module

The automatic grapheme-to-phoneme conversion module consists of two parts. The first part represents rule-based approach and the second part represents the data-driven approach, using neural network.

The first part is intended for the use in case, we don't have available phonetic lexicon for learning neural network. First rule based stress assignment is done, followed by grapheme-to-phoneme conversion procedure. The step of stress marking before grapheme-to-phoneme conversion is very important for Slovenian language, since the latter very much depends on the type and place of the stress.

In case we have available phonetic lexicon, data driven approach, representing the second part in the module, using neural network can be used. Here, the phonetic lexicon is used as a data source for training the neural networks. The neural network which was taken for the basis of this part is based on a method used and described in the Stuttgarter Neronaler Netz Simulator SNNS (Zell,1994). The data preparation, the generation of the training patterns and the training of neural networks are done completely automatically. The transcription is performed in two steps. The first one converts the graphemes into phonemes and inserts the syllable breaks in the phoneme string, and the second one inserts stress marks. The problem how to perform mapping between graphemes and phonemes by generation of training patterns for neural network, was solved automatically as proposed in (Hain,1994).

For both neural networks a multilayer perceptron (MLP) feedforward network with one hidden layer was used. As learning algorithm the backpropagation algorithm was chosen.

The pronunciation is derived from the IPA-Alphabet. In order to represent the IPA symbols in ASCII characters the SAMPA format is widely used. In our grapheme-to-phoneme conversion module the SAMPA phonetic transcription symbols for Slovenian language are used (SAMPA,1998).

For performing statistical analysis of the input text corpora, data-driven approach using neural network was used, since we had available enough material for learning neural network (from Slovenian phonetic lexicon and

Onomastica phonetic lexicon of names for Slovenian language). The generated phonetic transcriptions are sent to the unit generator.

3.2. Non-uniform units generator

This module gets at the input phonetic transcriptions of all sentences from the grapheme-to-phoneme conversion module. All non-uniform type of units for each sentence are generated (monophones, diphones, triphones, fivephones). For each text corpus, generated monophones, diphones, triphones and fivephones are sent to the module, for statistic analysis of the generated non-uniform units.

3.3. Module for statistic evaluation of the text corpora

This module performs statistic analysis of corresponding units in each corpus. The statistics on non-uniform units, generated in the previous module, is performed on the sentence level and the whole text corpus level.

The module sorts all units and assigns to them corresponding unit frequencies. The obtained statistic results define which sentences are richer in various units and also define the non-uniform unit structure of corresponding text corpora.

The sentences, assigned with statistic data and generated all non-uniform units, are sent to the 'reduce module'.

3.4. Reduce module

The maximal length of speech corpus was decided to be 1200 sentences what corresponds approximately to three hours of speech. To achieve that goal, we had to decrease the number of sentences in the chosen part of the input text corpus. Since we did not want to lose any unit from the corpus but only wanted to remove duplications of units, a careful elimination of sentences considering the unit context was performed. At the end, all sentences with duplicated units were removed. The processing stopped when the number of 1200 sentences in the input text corpus was achieved.

At the end, four corpora with 1200 sentences were generated. They all had similar unit statistic, although the units across the corpora were not the same.

3.5. Replace module

At the output of the analysis level, the text corpus with maximum number of different non-uniform units (monophones, diphones, triphones, fivephones) is selected as target corpus and is used as a starting-point

for generation of the final corpus out of four, generated so far. The replace module performs comparison between the target text corpus that has maximum number of different units and between other three corpora. It performs analysis on the sentence level. Basically the richest sentences with different units in the three text corpora are searched and added to the target corpus if these units are not already in it. If this is the case, the module replaces the poorest sentences in the target corpus, regarding the non-uniform units, with these sentences.

The user of the system can define the number of different units in the sentence that can be considered as a non-uniform unit rich sentence.

4. Statistical results of text corpora regarding the non-uniform units

In the following the analysis results, obtained with the reduce module for each input text corpus, that was taken into the account in overall statistical analysis, will be given.

In the table 1 and 2 the statistics of non-uniform units in text corpora 1-4 is presented. The text corpus 1 consisted of 4657, the text corpus 2 of 4546, the text corpus 3 of 6991 and the text corpus 4 of 4842 sentences. The right column in the table represents the total number of different units in the corresponding text corpus.

Text corpus 1	
Monophones	38
Diphones	1030
Triphones	11398
Fivephones	64811
Text corpus 2	
Monophones	38
Diphones	1001
Triphones	10233
Fivephones	55218
Text corpus 3	
Monophones	38
Diphones	1028
Triphones	11283
Fivephones	69668
Text corpus 4	
Monophones	38
Diphones	1028
Triphones	9126
Fivephones	52039

Table 1: Statistics of non-uniform units on all text corpora.

The reduction of sentences was performed analysing different triphones. The reduce module checked for every

sentence, if elimination of the corresponding sentence would not cause, that the frequency of any triphone would fall below predefined threshold (min. number of triphones in the corpus). If this was the case, the reduce module did not reject the sentence, but moved to the next one. The results of reduction on each input text corpus are presented in tables 2-5. In the first column there are threshold numbers for reduce module. Basically the module can reduce the corpus until the number of 1200 sentences in the input text corpus is achieved, but the process stops if the number of corresponding triphones falls below the predefined threshold number. In the third column the number of input rejected sentences for given threshold is presented and in the fourth the number of accepted sentences is given.

Text corpus 1			
min. num. of triphones	num. of sentences	rejected	accepted
10	4657	1187	3470
9	4657	1395	3262
8	4657	1489	3168
7	4657	1757	2900
6	4657	1894	2763
5	4657	2289	2368
4	4657	2498	2159
3	4657	3130	1527
2	4657	3402	1255
1	4657	3457	1200

Table 2: Results of reduction process

Text corpus 2			
min. num. of triphones	num. of sentences	Rejected	accepted
10	4546	1415	3131
9	4546	1627	2919
8	4546	1750	2796
7	4546	2004	2542
6	4546	2165	2381
5	4546	2538	2008
4	4546	2744	1802
3	4546	3272	1274
2	4546	3346	1200
1	4546	3346	1200

Table 3: Results of reduction process

Text corpus 3			
min. num. of triphones	num. of sentences	rejected	Accepted
10	6991	2910	4081
9	6991	3203	3788
8	6991	3419	3572
7	6991	3807	3184

6	6991	4067	2924
5	6991	4544	2447
4	6991	4846	2145
3	6991	5520	1471
2	6991	5791	1200
1	6991	5791	1200

Table 4: Results of reduction process

Text corpus 4			
min. num. of triphones	num. of sentences	rejected	Accepted
10	4842	1503	3339
9	4842	1736	3106
8	4842	1888	2954
7	4842	2212	2630
6	4842	2398	2444
5	4842	2800	2042
4	4842	3036	1806
3	4842	3609	1233
2	4842	3642	1200
1	4842	3642	1200

Table 5: Results of reduction process

According to this statistical results it will be the best to select the text corpus 1 as the richest corpus with different non-uniform units. But text corpora 1-3 contain mainly newspaper sentences, which contain many foreign words (especially names). These names were detected using tokenizer module. After replacing these names with optional Slovenian ones, the number of different units rapidly decreased in all text corpora from 1-3 (approximately for 3000, in case of triphones). The text corpus 4 contains mostly sentences from literature and the exchange of foreign names with Slovenian ones did not much affect the previous obtained statistical results.

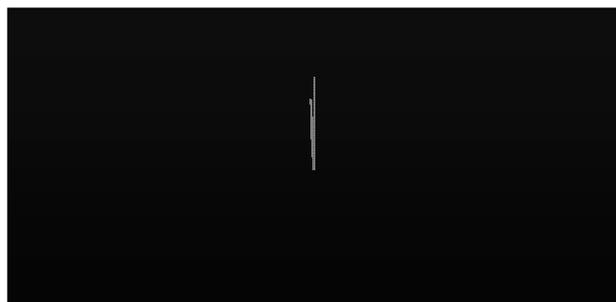


Figure 3. Triphone representation in finally chosen text corpus

Therefore the text corpus 4 was chosen as a target corpus and was sent to the replace module. The replace module compared the units in the sentences of the target

corpus with units in sentences of all other corpora. The module replaced the poorest sentences in the corpus with sentences that consist of units (triphones) that did not occur in the target corpus at all. Finally, the statistic on this newly obtained text corpus was done.

The finally obtained number of triphones was **9391**. In figure 3 we can see the representation of each triphone in the database. As it can be seen from the figure the frequency of triphones is well balanced.

5. Conclusion

Using the proposed architecture for defining optimal text corpus, possibly without degradation of richness with non-uniform units, the defined text corpus corresponds to the predefined specification. At the same time, the richness with different non-uniform units remained at the same level as in the initial corpus with almost four times more sentences.

As we can see, the statistic did not show very big differences in number of non-uniform units between all processed parts of input text corpus. After the optimisation process, the finally generated target text corpus with maximal number of different non-uniform units has been defined and used for recording of the Slovenian Speech Corpus. The target corpus consists of 1200 sentences, what corresponds to about 3 hours of speech.

6. References

- Matej Rojc, Janez Stergar, Ralph Wilhelm, Horst-Udo Hain, Martin Holzapfel, Bogomir Horvar,(1999) A Multilingual text processing Engine for PAPAGENO Text-To-Speech Synthesis system, Proceedings EUROSPEECH 1999, Budapest, pp. 2107-2110
- Zell, A.(1994). Simulation Neuronaler Netze. Bonn, Paris; Reading, Mass., Addison-Wesley
- Horst-Udo Hain, (1999) Automation of the training procedure for neural networks performing multilingual grapheme to phoneme conversion, Proceedings EUROSPEECH 1999, Budapest, pp. 2087-2090
- SAMPA for Slovenian, (1998)
<http://www.phon.ucl.ac.uk/home/sampa/sloven-uni.html>

