Automatic Construction of a Prosodically Rich Text Corpus for Speech Synthesis Systems

T. Lambert

tlambert@freeola.net, UK

Abstract

This paper presents a method for an automatic compilation of a phonologically rich text database, which is used in a concatenative text-to-speech (TTS) synthesis system. In this method, linguistic features are predicted from text using Festival's linguistic engine. A set of phonological units for a specific text is compiled from attribute value lists (AVLs). Phrases/sentences that contain the phonological units that are not included in the database are added to the database. This is an efficient way for generating database prompts with a specific prosodic content; the prompts can then be recorded and converted into voice. The method described here can be used for languages other than English.

1. Introduction

In concatenative TTS systems sequences of recorded speech are excised from a single-speaker speech database at run-time and then joined together to produce a new utterance. Past research has shown that natural-sounding synthetic speech can be produced by selecting non-uniform units (i.e. units of variable length) from large speech databases [1, 2, 6, 29, 17, 20]. Studies [2, 6, 29, 25] indicate that the naturalness of synthetic speech can be improved by excising longer sequences of recorded speech from the database, this reduces the number of concatenation points in a synthesized utterance. It was argued by [26] that longer chunks of recorded speech preserve natural rhythm and prosody better than shorter sequences. It was argued by [18] that units for synthesis have to be phonologically compatible regardless of their length if the prosody of synthetic speech is to sound natural.

1.1. Prosody In Natural Speech

Natural speech is made up of intonational phrases; these are usually marked by perceptual characteristics (pauses, amplitude changes, fundamental frequency (F_0) and changes in energy). In connected speech, prosody depends on many linguistic, paralinguistic and extra-linguistic factors. When reading a sample of text a speaker's prosody is affected by punctuation marks, a text layout, speaker's emotional and physical state, intended audience, speaking style and the speaker's understanding of the text being read. In addition to speaker-dependent prosodic variations, there are also prosodic variations in speech which are associated with syllable, word-level and phrase-level stress. Stress patterns affect the intonation of speech as well as phonetic realizations (e.g. the alternation [t] $\rightarrow$ [t$^{\text{h}}$] in *con'test v.* after the primary stress). On a phonetic level each sound has its own intrinsic 'microprosody', which is affected by neighbouring phonetic and phonological contexts.

1.2. Prosody Modelling in Speech Synthesis

Prosodic modelling in a TTS system has a task of effectively dealing with prosodic phrasing, intonation and segmental duration. Prosodic phrasing generally considers punctuation markers, syntax and semantics. Intonation modelling is associated with fundamental frequency (F_0) parameters, and can be constructed from acoustic [14], perceptual [22] and linguistic [23] models. Linguistic rules [23] and machine-labelled or hand-labelled speech corpora are important for the generation of prosody in a corpus-driven TTS system. Durational characteristics of normal speech can be rule-driven or can be derived from speech data in corpus-driven TTS systems. The patterns of intonation and duration contribute to the rhythm of natural speech. In [11] it was argued that in order to improve natural-sounding prosody it was important to improve the rhythm of synthetic speech; the authors described the system using the rhythmic analysis based on metrical phonology [16].

1.3. Source of Prosody In Concatenative TTS

In corpus-driven TTS systems prosodic characteristics are derived from a speech database. It has been suggested that the size of a speech database determines the quality of synthetic speech [1, 10]. Ideally, a speech database should have every speech segment in every prosodic context. However, due to the enormous amount of prosodic detail that a phoneme may have in continuous text and speech, such databases would be astronomical in size. A speech database may not even contain every speech segment in every lexical stress environment. In such cases, lexical stress variations are usually modelled by changing speech signal parameters (e.g. F_0, amplitude, duration) [24]. The past research revealed that more natural-sounding speech is obtained if prosodic information is included in unit selection [7]. However, signal processing techniques employed to do prosodic modifications were reported to reduce the quality of synthesized speech [1, 8, 29].

Bearing in mind that in concatenative TTS systems synthetic speech derives its characteristics from a speech database, it is reasonable to argue that in such systems the linguistic content of text recorded as speech is of paramount importance. This paper discusses an automatic construction of a text database with a specific linguistic and prosodic content. The paper is structured as follows: prosodic information in text is discussed in section 2; section 3 discusses the retrieval of prosodic content from text; an algorithm for automatic construction of prosodically rich text corpora for TTS is discussed in section 4; the conclusions and future considerations are presented in sections 5 and 6 respectively.

2. Prosodic Information in Text

The linguistic information in text and speech exists in hierarchical relationship structures. For instance, a word consists of syllable(s), a syllable structure relates to an onset and a rhyme, the rhyme structure is related to a peak and a coda; the onset, peak and coda are a sequence of vowels and consonants (*CVC*). Each segment in a sequence of vowels and consonants can be represented through relationships with suprasegmental linguistic structures by way of attribute-value lists (AVL). In the Festival Speech Synthesis System [12] feature-values are derived from heterogeneous relation graphs [27]. The linguistic engine of the Festival speech synthesis system exports a variety of features [4] (e.g. lexical stress, identity and lexical stress of syllables to the left and right of the syllable under investigation, the word in which the syllable resides, the word's category, position of the syllable in word, etc.).

It is stated in [18] that a speech database should include at least diphone units in all lexical stress environments; the author argues that *lexical units* (units inclusive of their lexical stress) capture the minimum amount of prosody necessary for synthesis. In corpus-driven TTS systems lexical units are a good compromise solution between the phonemic unit types, which are prosodically poor but not so abundant in language and speech, and the word-level and phrase-level prosodic unit types, which are attribute-rich but over-abundant. In [18] the stress of a phoneme unit is defined by the stress of its syllable. For example, in the word *reviews* (phonetically transcribed as /rɪˈvjuːz/) the phonemes inherit their stress from syllables. The lexical transcription for this word is [r0ɪ0ˈv1j1uː1z1] where the unstressed syllable is indicated by 0 and the stressed syllable by 1. In this manner, an insufficiently descriptive phoneme on a phonemic level is shifted to a more prosodically rich phonological level. Whilst a speech database with every phoneme in every prosodic context would be astronomical in size, a speech database with phonemes in every lexical stress environment, on the other hand, is feasible and can be automatically constructed from the linguistic information in text.

3. Retrieval of Information from Text

In the method described here linguistic features in text are predicted using the Festival Speech Synthesis System [12]. The attribute value lists (AVLs) are generated by the engine; each segment in a sequence of *CVC* is associated with its suprasegmental prosodic attributes (the attributes are user-defined and different levels of prosodic information can be specified). For example, an entry for a phoneme in the AVL may be given as [s 2 0 single] – this corresponds to [phoneme name, position in syllable, syllable stress, syllable position in word].

The algorithm for feature extraction is as follows:

1. For a sample of text *utterance structures* are exported for each sentence using Festival.
2. Festival's script *dumpfeats* [3] is run to generate an AVL, where each segment in a *CVC* sequence in the given text sample is associated with a set of features of interest (e.g. phoneme name, syllable stress, position of syllable in word, position of word in phrase, syllable break, word break).
3. Units with different levels of prosodic content can be compiled from AVLs, i.e. phonemic diphones, lexical diphones (diphone + lexical stress), syllable-level diphones (diphones that include the prosodic information relating to syllables), word-level and phrase-level diphones, sequences consisting of consonants and vowel (C_2V), vowel and consonants (VC_2) and *CVC* sequences preceded and followed by a silence.
4. Each word in the given text sample is *looked up* in the system's dictionary or letter-to-sound (LTS) rules in order to determine the information regarding syllable boundaries and syllable stress. Using this information a list of syllables (inclusive of the lexical stress) is generated from the given text sample.

Figure 1 shows the results of an experiment in which different unit types were generated from Festival's AVLs for a number of different text sources when using a British English accent. The figure 1 shows the frequency distribution of distinct unit types. Phonemic diphones (e.g. /pæ/ in *Patrick*) are considered here the poorest in their prosodic content. Lexical diphones include syllable stress (as /p1æ1/ in *Patrick*). Syllables are also distinguished by their stress information. S-level diphones represent diphones with the syllable-level information, i.e. syllable stress, position of a segment in syllable and the break level after the syllable. P-level diphones include the phrase-level and word-level information, i.e. syllable stress, a distinction of whether the word is a content or a function word, position of a syllable in the word, position of the word in a phrase and the ToBI phrase-level break. In this experiment all text sources except the text E contain approximately ten thousand words. It is clear in the figure 1 that the prosodically rich units (i.e. s-level and p-level diphones) are more diverse in English texts.

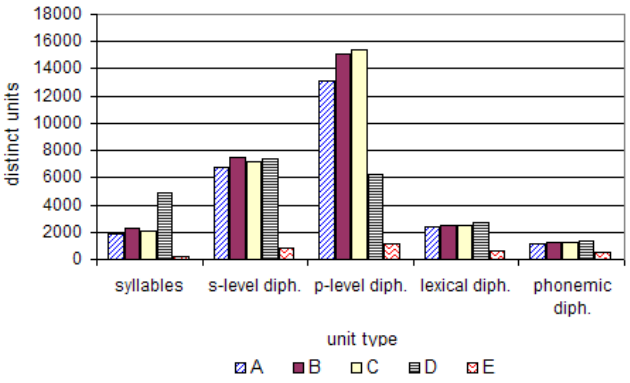

Figure 1: Frequency distribution of units with different prosodic information in various texts. Text sources are: A: My Man Jeeves by P. G. Woodhouse [15]; B: Selected Prose of Oscar Wilde [15]; C: an extract from a science journal [21]; D: a list of names and surnames [9]; E: Fern Hill by D. Thomas.

4. Automatic Construction of a Prosodically Rich Text Database

An automatic construction of a prosodically rich text database is a process in which new phrases and sentences that contain prosodic units that are not included in the database are added to the database. At the beginning of this process the database itself can be as small as one word. The process is shown in figure 2. Prosodically rich phonological units (e.g. lexical, s-level or p-level diphones, syllables) are compiled from the AVLs for a specific text sample, as discussed in section 3 of this paper. The prosodically rich units are then used with the correspond-

ing prosodically rich transcription of the text sample to generate a subset of that text sample. The prosodically rich transcription of text is also generated from the AVLs.

The algorithm for database construction is as follows:

1. export an AVL of features from a sample of text that is considered for addition to the database;
2. export an AVL from the database (the database can initially be as small as one word);
3. compile a list of phonological units from the text sample;
4. generate a prosodically rich transcription of text from the AVL;
5. compile a list of phonological units from the database;
6. find the units in the text sample that do not exist in the database;
7. find a subset of the text sample for units generated in step 6;
8. add the subset of the text sample to the database.

To clarify further: by referring to figure 2, database *D* contains text which is intended to be recorded as speech; the size of *D* can initially be as small as one word. Suppose a sample of text *T* (e.g. a newspaper article) is being considered for its inclusion into the database *D*. In order to ascertain whether *T* (the newspaper article) is worthy of the database assimilation, it is necessary to examine the linguistic content of both the text *T* and the database *D*. The AVLs for both *T* and *D* are generated by passing, in two separate runs, the textual content of *T* and *D* through the system's engine (this is given as the first step in the algorithm in section 3). Phonological units (*P*) are compiled from the AVL for *T* and these units may be diphones, triphones, syllables, C_2V and VC_2 sequences or other user-defined prosodically rich units. A list of the phonological units found in *D* is also compiled. It is now necessary to find the difference in unit coverage between *T* and *D*. Phonological units that appear in *T* but not *D* ($T_P \neg D_P$) are considered to be of interest. The units ($T_P \neg D_P$) and the corresponding transcription of the text sample *T* are subjected to a set cover algorithm (SCA). A subset of the text sample *T* that contains all units in $T_P \neg D_P$ is generated by the set cover algorithm and added to the database *D*. The process is repeated with a new text sample until all the phonological units in the pre-defined linguistic feature set are covered.

The step 7 in the above algorithm is concerned with finding a subset of *T* that contains all the units in ($T_P \neg D_P$). This is essentially a set covering problem and it can be resolved by a set cover algorithm (some forms of greedy set cover algorithms were used in the past [28, 5, 13, 19, 18]). For this purpose, the set cover algorithm in step 7 is run on the units (generated in step 6) and the text transcription (generated in step 4).

It is important to emphasize here that the prosodic richness of an automatically constructed text database and, ultimately, its linguistic quality are primarily determined by the prosodic criteria specified in the algorithm's design. The final text database (which is eventually recorded as speech) is only as good as the linguistic content it is designed to capture.

5. Conclusions

This paper discussed an efficient method for an automatic construction of a text database using linguistic features generated by the Festival Speech Synthesis System. Two algorithms were presented: the algorithm in section 3 dealt with an automatic compilation of prosodically rich units from text. The algorithm in section 4 dealt with an automatic compilation of a subset of phrases/sentences from text which contained prosodically rich units that were missing from the database. The text database created by the method presented in this paper can be recorded and converted into voice as per instructions for building voices in Festival [3]. This method is applicable to other TTS systems and languages other than English.

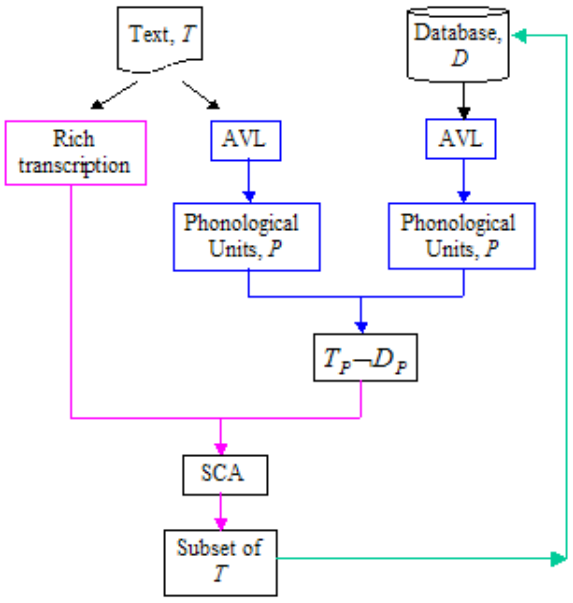

Figure 2: Method for an automatic compilation of a prosodically rich text database

The method discussed here generates a text database that contains a specific set of prosodic features (e.g. syllable stress, word-level and phrase-level prosodic information, intonation events, etc.). The TTS system's behaviour is considered when constructing the linguistic content of the database – in short, the database is created for the system by the system itself.

6. Future Work

The method presented here constructs a speech database from linguistic features present in text. Synthetic speech is greatly affected by the way a text database is recorded and by a speaker's accent features. It should be possible to model the speaker's accent features so that they can be included in the database construction. This should enable an investigation into which aspects of synthetic speech are attributable to the speaker and which to the system's algorithms.

7. References

[1] A. Black and N. Campbell. Optimising Selection of Units From Speech Databases for Concatenative Synthesis. In *Proceedings of Eurospeech*, pages 581–584, Madrid, Spain, 1995.

[2] A. Black and P. Taylor. Automatically Clustering Similar Units for Unit Selection in Speech Synthesis. In *Proceedings of Eurospeech*, volume 2, pages 601–604, Rhodes, Greece, 1997.

[3] A. W. Black and K. A. Lenzo. *Building Synthetic Voices.*

Centre for Speech Technology Research, University of Edinburgh, Edinburgh, 1999. For Festvox 2.0 Edition.

[4] A. W. Black, P. Taylor, and R. Caley. *The Festival Speech Synthesis System: System Documentation*. Centre for Speech Technology Research, University of Edinburgh, Edinburgh, 2002. Edition 1.4, for Festival Version 1.4.3.

[5] O. Boëffard and F. Emerard. Application-dependent Prosodic Models for Text-to-speech Synthesis and Automatic Design of Learning Database Corpus Using Genetic Algorithm. In *Proceedings of Eurospeech*, pages 2507–2510, Rhodes, Greece, 1997.

[6] A. P. Breen and P. Jackson. Non-uniform Unit Selection and the Similarity Metric Within BT's Laureate TTS System. In *Proceedings of the Third ESCA/COCOSDA Workshop on Speech Synthesis*, pages 201–206, Jenolan Caves, Blue Mountains, Australia, 1998.

[7] I. Bulyko and M. Ostendorf. Joint Prosody Prediction and Unit Selection for Concatenative Speech Synthesis. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 781–784, Salt Lake City, Utah, 2001.

[8] N. Campbell and A. Black. Prosody And the Selection of Source Units for Concatenative Synthesis. In P. H. van Santen, R. W. Sproat, J. P. Olive, and J. Hirschberg, editors, *Progress in Speech Synthesis*, pages 279–282. Springer-Verlag, New York, 1997.

[9] Census. American census. IRSB Internet Staff (Population Division) http://www.census.gov, 1990. U.S. Census Bureau, Population Division, Population Analysis and Evaluation Staff.

[10] A. Conkie. Robust Unit Selection System for Speech Synthesis. In *Joint Meeting of ASA/EAA/DAGA*, pages 978–982, Berlin, Germany, 1999.

[11] M. Edgington, A. Lowry, P. Jackson, A. P. Breen, and S. Minnis. Overview of Current Text-to-Speech Techniques: Part II - Prosody and Speech Generation. *BT Technology Journal*, 14 (1):84–99, 1996.

[12] Festival. The Festival Speech Synthesis System. http://www.cstr.ed.ac.uk/projects/festival.html, 1996. Centre for Speech Technology Research, University of Edinburgh, Edinburgh.

[13] H. François and O. Boëffard. Design of an Optimal Continuous Speech Database for Text-to-Speech Synthesis Considered as a Set Covering Problem. In *Proceedings of Eurospeech*, pages 829–832, Aalborg, Denmark, 2001.

[14] H. Fujisaki. The Role of Quantitative Modeling in the Study of Intonation. In *Proceedings of the International Symposium on Japanese Prosody*, pages 163–174, Nara, Japan, 1992.

[15] M. Hart. Project Gutenberg. http://www.gutenberg.org, 2003.

[16] R. Hogg and C. B. McCully. *Metrical Phonology: a Coursebook*. Cambridge University Press, Cambridge, 1987.

[17] E. Klabbers, K. Stöber, R. Veldhuis, P. Wagner, and S. Breuer. Speech Synthesis Development Made Easy: The Bonn Open Synthesis System. In *Proceedings of Eurospeech*, volume 1, pages 521–524, Aalborg, Denmark, 2001.

[18] T. Lambert. *Databases for Concatenative Text-to-Speech Synthesis Systems - Unit Selection and Knowledge-Based Approach*. PhD thesis, School of Computing Sciences, The University of East Anglia, Norwich, 2005.

[19] T. Lambert and A. Breen. A Database Design for a TTS Synthesis System Using Lexical Diphones. In *8th International Conference on Spoken Language Processing (ICSLP)*, pages 1381–1384, Korea, 2004.

[20] T. Lambert, A. P. Breen, B. Eggleton, S.J. Cox, and B. P. Milner. Unit Selection in Concatenative TTS Synthesis Systems Based on Mel Filter Bank Amplitudes and Phonetic Context. In *8th European Conference on Speech Communication and Technology, Eurospeech 2003*, pages 273–276, Geneva, Switzerland, 2003.

[21] J. Cattell McKenn, editor. *The Popular Science Monthly*, volume LXXXVI, July-September. 1915. Oxford Text Archive, http://ota.ahds.ac.uk.

[22] P. Mertens, F. Beaugendre, and d'Alessandro C. F. Comparing Approaches to Pitch Contour Stylization for Speech Synthesis. In P. H. van Santen, R. W. Sproat, J. P. Olive, and J. Hirschberg, editors, *Progress in Speech Synthesis*, pages 347–364. Springer-Verlag, New York, 1997.

[23] J. Pierrehumbert. Synthesizing Intonation. *Journal of the Acoustical Society of America*, 70:4:985–995, 1981.

[24] M. Schröder. Emotional Speech Synthesis: a Review. In *Proceedings of Eurospeech*, volume 1, pages 561–564, Aalborg, Denmark, 2001.

[25] K. Stöber, T. Portele, P. Wagner, and W. Hess. Synthesis by Word Concatenation. In *Proceedings of Eurospeech*, volume 2, pages 619–622, Budapest, Hungary, 1999.

[26] M. Tatham and E. Lewis. SPRUCE - High Specification Text-to-Speech Synthesis. http://www.essex.ac.uk/speech/research/spr-1.html, 1997. University of Essex and University of Bristol.

[27] P. Taylor, A. W. Black, and R. Caley. Heterogenous Relation Graphs as a Formalism for Representing Linguistic Information. *Speech Communication*, 33:153–174, 2001.

[28] J. P. H. van Santen. Methods for Optimal Text Selection. In *Proceedings of Eurospeech*, pages 553–556, Rhodes, Greece, 1997.

[29] J. R. W. Yi and J. R. Glass. Natural-Sounding Speech Synthesis Using Variable-Length Units. In *Proceedings of the International Conference on Spoken Language Processing*, pages 1167–1170, Sydney, Australia, 1998.