Speaker-Specific Pronunciation for Speech Synthesis*

Lukas Latacz[1,2], Wesley Mattheyses[1], and Werner Verhelst[1,2]

[1] Vrije Universiteit Brussel, Dept. ETRO-DSSP, Brussels, Belgium
[2] iMinds, Dept. of Future Media and Imaging, Ghent, Belgium
{llatacz,wmatthey,wverhels}@etro.vub.ac.be

Abstract. A pronunciation lexicon for speech synthesis is a key component of a modern speech synthesizer, containing the orthography and phonemic transcriptions of a large number of words. A lexicon may contain words with multiple pronunciations, such as reduced and full versions of (function) words, homographs, or other types of words with multiple acceptable pronunciations such as foreign words or names. Pronunciation variants should therefore be taken into account during voice-building (e.g. segmentation and labeling of a speech database), as well as during synthesis.

In this paper we outline a strategy to automatically deal with these variants, resulting in a speaker-specific pronunciation. Based on a labeled speech database, the pronunciation lexicon is pruned in order to remove as much as possible pronunciation variation from the lexicon. This pruned lexicon can be used to train speaker-specific letter-to-sound rules. If the speaker has uttered a word in different ways, then these variants are not pruned. Instead, decision trees are trained for each of those words, which are used to select the most suitable pronunciation during synthesis. We tested our approach on five speech databases, and two lexicons per speech database. The automatic selection of pronunciation variants yielded a small improvement over the baseline (selecting always the most common variant).

Keywords: speech synthesis, lexicon, pronunciation variants, speaker-specific.

1 Introduction

In a typical modern speech synthesizer, the input text needs to be converted in a phonemic or phonetic sequence before the actual speech can be produced. This conversion is usually done using a pronunciation lexicon in which the phonemic transcription of each word can be looked up.

Ideally, the transcriptions in a pronunciation lexicon are tailor-made for a specific speaker. In practice, they are made for a specific regional accent of a language. Some lexicons are so-called meta-lexicons, allowing to generate pronunciations in different regional accents (e.g. Unisyn [1]), or different speaking styles (e.g. FONILEX [2]). These rules could also be adapted for a particular speaker, but this requires a significant amount of (manual) work. Another approach to generate speaker-specific pronunciation

* The research reported in this paper was partly supported by the projects IWT-SPACE, iMinds-RAILS, iMinds-SEGA and EC FP7 ALIZ-E (FP7-ICT-248116).

I. Habernal and V. Matousek (Eds.): TSD 2013, LNAI 8082, pp. 501–508, 2013.

is to use a phone-recognizer to detect variants uttered by the speaker as proposed in [3]. This approach also requires some additional manual work, as the accuracy of the recognizer is not high enough for a fully automatic approach.

Taking pronunciation variations into account during synthesis can improve synthesis quality as mentioned in [4], but their approach cannot easily be applied in most speech synthesizers as multiple pronunciations for the same word need to be supported. Most speech synthesizers use a single pronunciation for each word during synthesis.

In case multiple pronunciation variants of a certain word are present in the lexicon, the speech synthesizer should have a strategy to select the best pronunciation variant. Sometimes the pronunciation variant depends on the part-of-speech (e.g. in Dutch: een (article) versus een (numeral)) or on the meaning of the word (e.g. in English: bass (music) versus bass (fish)). This information can be added to the pronunciation lexicon, and allows selecting the correct variant. This approach can not be used in all cases, as some variants have the same meaning or part-of-speech tag (e.g. the full and reduced versions of function words). An option in the Multisyn synthesizer [5] allows setting whether full or reduced vowels should be chosen, but this affects the whole utterance. Bennett and Black proposed to train decision trees to select the correct pronunciation variant, but applied their approach only to predict reduced or full function words [6].

Our strategy to generate a speaker-specific pronunciation is fully automatic and is also able to deal with pronunciation variants beyond function words. The most suitable pronunciation is selected during voice-building and synthesis. Our approach requires that the same pronunciation lexicon is being used for both labeling the speech database and synthesis, and that post-lexical rules are not used. The latter has the advantage that the risk of a potential mismatch between the segmentation labels and the output of the front-end is reduced.

Instead of modifying existing lexicon entries or adding new entries to the lexicon, we assume that the lexicon contains enough pronunciation variants to match the speech of the speech database sufficiently. We realize that such assumption is difficult to test, but - based on our own experience with building voices - adding additional pronunciations for known words is not often needed if a pronunciation lexicon suitable for speech synthesis is being used. We therefore propose to remove entries (unnecessary variants) from the lexicon, and to automatically select the correct pronunciation variant out of multiple options which were uttered by the speaker.

Table 1. Pronunciation variants in analyzed lexicons. Words are words with a unique orthography (disregarding capitalization). The *unresolved* column refers to the number of unique words which have multiple pronunciation variants which could not be disambiguated using information present in the lexicon. Lexical stress and syllable boundaries are not taken into account.

Lexicon	Entries	Words with multiple pronunciations	Unresolved words
CMUDict	105901	168	6
OALD	72301	428	111
Unisyn-RP	167617	1003	268
Unisyn-GA	167617	1084	330
KUNLEX	312232	100	64
FONILEX	188550	7785	7760

This paper continues as follows. In section 2, we have analyzed the pronunciation variants in several pronunciation lexicons. Even though the number of variants in these lexicons is quite limited in comparison to the total number of lexicon entries, we still need to cope with these variants during labeling and synthesis. In section 3 we describe how we perform labeling and segmentation. In section 4 we describe how we have generated a speaker-specific pronunciation lexicon. Our approach to deal with pronunciation variation during synthesis is described in section 5 and evaluated in section 6 using several speech synthesis databases and lexicons.

2 Lexicon Analysis

We have analyzed the following pronunciation lexicons in order to find out how many words with multiple pronunciations are present in each lexicon. The CMU lexicon [7] is a relatively widely-used American English lexicon. We have used CMUDict 0.4 included in the Festival speech synthesis system. It contains part-of-speech tags for disambiguation. The Unisyn lexicon [1] allows generating lexicons for many English accents. It contains part-of-speech tags for each entry, and an optional field for disambiguation. This optional field contains either the meaning of the word, or indicates whether the word is (un)reduced. We have used both the (British English) Received Pronunciation and General American accent in our experiments. The (un)reduced option was ignored in our experiments. OALD [8] is the computer-useable version of the Oxford Advanced Learner Dictionary and contains British English pronunciation. We have used the version included in Festival, which contains part-of-speech tags for each entry. KUNLEX is a Dutch lexicon which covers standard Northern-Dutch (i.e. the standard Dutch spoken in The Netherlands). It is part of NeXTeNS [9], a Dutch extension for Festival. FONILEX [2] is a Flemish pronunciation lexicon. We are using a modified version of this lexicon, which contains a high-level phonemic transcription (the pronunciation of each morpheme is transcribed as it would be pronounced in isolation) and the default speaking style (normal pronunciation). FONILEX does not contain part-of-speech tags, but these can be obtained from the CELEX lexicon [10] as each FONILEX entry contains a reference to its corresponding CELEX entry.

All these pronunciation lexicons contain some pronunciation variants that cannot be disambiguated using part-of-speech or other information in the lexicon, as shown in table 1. The amount of these variants varies between the lexicons. The largest number of variants can be found in the FONILEX lexicon. We assume that this is because this lexicon was originally constructed for speech recognition purposes. The other lexicons contain far fewer pronunciation variants.

Some of the words with pronunciation variants are homographs, proper names or foreign words with multiple acceptable pronunciations. Reduced and full versions of common function words can also be frequently found. The use of other variants is more questionable. We assume that these could be speaker- or region-depended, or simply exist because of the difficulty of accurate transcription. The lexicon developer needed to make hard decision during the construction of the lexicon, and sometimes none of the options was entirely satisfying: the phonemic transcription is intrinsically an approximation of the continuous nature of speech sounds.

3 Segmentation and Labeling

A large speech database is required for high-quality speech synthesis, and needs to be segmented into phonemes. The word sequence of each utterance of the database is normally known beforehand, as the speaker is instructed to read a recording script. Currently, the most popular segmentation technique is to use hidden Markov models: a large vocabulary continuous speech recognizer is run in forced-alignment mode - the most likely sequence of states is selected for each utterance of the database, given input speech signals and phonemic sequences. If multiple phonemic sequences are possible (due to optional silence insertions or multiple pronunciations), the best phonemic sequence - in terms of likelihood - is automatically chosen using the Viterbi algorithm. Our approach for segmenting and labeling is based on the open source HMM-based speech recognizer SPRAAK [11], but can probably also be extended to other recognizers such as HTK or Sphinx.

The output of the (HMM-based) segmentation algorithm does not include which variant is actually chosen, as only a phoneme sequence including timing information is typically given. We have solved this problem by constructing a word lattice. The correct sequence of pronunciation variants as selected by the segmentation algorithm can be obtained by traversing through the lattice using the output phoneme sequence. If at some point the output sequence does not match the selected variant, a different variant needs to be chosen and backtracking might be necessary. Optional silences (i.e. silences between words) are also taken care off. Since no additional post-lexical rules are being used in our case, a matching word sequence will always be found. Words with the same pronunciation but different lexical stress or syllable boundaries cannot be distinguished using this approach, as the segmentation algorithm is not able to take these differences into account. In our current implementation, the first suitable variant is always chosen.

4 Lexicon Pruning

If a word has multiple possible pronunciations in the lexicon, we need to select the most suitable option during synthesis. As discussed previously, several such words can be found in the lexicons we examined. Part of the variation can be reduced by using part-of-speech. Our approach to deal with the remaining variation is to base the selection of the pronunciation variants on the speech database, and focus on these variants that have been uttered by the speaker. Using this information, we can create a new version of the pronunciation lexicon which contains as few pronunciation variants as possible, resulting in a speaker-specific lexicon.

Ideally, each word in the pronunciation lexicon should have exactly one pronunciation in the lexicon, unless additional information is present in the lexicon to disambiguate the variants. In this section, we refer to a word and this additional information as an unambiguous word in the lexicon. These unambiguous words could also have multiple pronunciations, as previously demonstrated. The goal of the lexicon pruning is to keep a single pronunciation variant for each unambiguous word. If a word is present in the speech database, then all variants which are pronounced by the speaker are kept. Words with multiple pronunciation variants can still be present in the pruned lexicon.

Unambiguous words that are not present in the speech database need to be dealt with differently. We do not know which variant will give the best synthesis quality before actually synthesizing these variants. Choosing the best variants by hand though iterative listening is a potential solution, but not very practical for texts longer than a few utterances. We therefore propose the following solution.

Words that are covered by common diphones can typically be synthesized better than words which contain rare diphones. A very simple quality estimation is to calculate the average diphone coverage of a word, using the diphones present in the speech database. We therefore keep only the variant that has the highest average diphone coverage. If multiple variants have the same coverage, we select the first variant. A probably better solution would be then to look at larger elements such as triphones.

The resulting pruned pronunciation lexicon is a speaker-specific lexicon, tailored to a specific speech database. As such, it becomes possible to train speaker-specific letter-to-sound rules using the pruned lexicon.

5 Dealing with Multiple Pronunciation Variants

The most straight-forward approach is to always select the most common pronunciation for each so-called unambiguous word, i.e. the variant that is chosen most frequently. An alternative is to select a variant based on the average diphone frequency, in order to maximize diphone coverage (as explained previously). Both approaches ignore all non-selected variants from the lexicon. As such, they are less suitable if two or more variants of the same word have similar frequency in the speech database.

Decision trees can also be used to select the most suitable pronunciation. Our approach is quite similar to the approach proposed by Bennett and Black [6] They used decision trees to select either the full or the reduced versions of function words. The trees were trained using linguistic data extracted from an annotated speech database. We have generalized their approach for all words with pronunciation variants in a speech database. We have used wagon part of the Edinburgh Speech Tools to train the decision trees. A separate tree is constructed for each distinct word. The training data is based on

Table 2. Results of the lexicon pruning. *Unresolved words* are words which could not be disambiguated using information in the lexicon.

Database	Lexicon	Entries in pruned lexicon	Unresolved words
SLT	CMUDict	105894 (-0,01%)	0
	Unisyn GA	167167 (-0,27%)	26
AWB	CMUDict	105894 (-0,01%)	0
	Unisyn GA	167172 (-0,27%)	29
RJS	OALD	72002 (-0,41%)	3
	Unisyn RP	167172 (-0,27%)	26
AVKH	FONILEX	178262 (-5,46%)	21
	KUNLEX	311696 (-0,17%)	2
AWDC	FONILEX	178251 (-5,46%)	11
	KUNLEX	311694 (-0,17%)	0

the instances uttered by the speaker. Linguistic features are extracted for each instance. Our current features take part-of-speech, phrase breaks, and phonemes and graphemes of the surrounding words into account. If the pronunciation of each word is processed sequentially (which is normally the case), then it is not possible to use features below the word level for words which succeed the current word, as this information is not yet known. The best stop size (a parameter to control the size of the tree) can be selected using n-fold cross-validation (e.g. n=5).

6 Evaluation

6.1 Databases

The following databases were used in our experiments: SLT and AWB [12], RJS [13], the audio part of the AVKH [14] database, and AWDC [15]. A subset of the AWDC database was used, containing about 2.5 hours of speech and consisting of sentences and paragraphs selected from childrens stories. Each database was segmented twice with SPRAAK, with two different lexicons. The SLT and AWB databases were segmented with the CMUDict and Unisyn (General American) lexicons. We chose to use US English lexicons for the Scottish-accented AWB database, because the CMUDict lexicon is the default lexicon for this database in the AWB voices supplied with Festival. The RJS database was segmented with the OALD and Unisyn (Received Pronunciation) lexicons. The Flemish databases were segmented with the KUNLEX and FONILEX lexicons. All databases contained some out-of-vocabulary words. These words were manually transcribed, based on the existing entries of the lexicon. These additional transcriptions were needed during segmentation. As we are not expert English phoneticians, words were only transcribed if the transcription was straight-forward, such as in case of compound words. Utterances which contained non-transcribed out-of-vocabulary words were not used in our experiments.

6.2 Lexicon Pruning

Result of the lexicon pruning can be seen in table 2. As expected from table 1, the impact of the lexicon pruning is quite small for all lexicons except FONILEX. As FONILEX was originally constructed for speech recognition, many alternative pronunciations were added by the lexicon developers in order to improve recognition.

6.3 Prediction of Pronunciation Variants

We evaluated the algorithms to select the best pronunciation variants using 5-fold cross-validation. Words with less than 6 instances in the database are ignored in this evaluation. The results in table 3 indicate that the best approach is to use decision trees. This indicates that the selection of the best variant can be generalized up to a certain extend - from data in the speech database. The worst approach is to select a variant based on diphone frequency. This is not unexpected, as this does not directly use information about which of the variants have been uttered by the speaker. Its performance is

Table 3. Evaluation of the automatic selection of pronunciation variants. Accuracy (% correct) is shown. The results of other voices are not included because not enough instances were available for evaluation.

Database	Lexicon	Instances in evaluation	Decision tree	Majority voting	Diphone selection
SLT	Unisyn GA	1660	**81%**	80%	60%
AWB	Unisyn GA	1914	**76%**	74%	58%
RJS	Unisyn RP	6962	**86%**	83%	75%
AVKH	FONILEX	2192	**90%**	82%	75%
AWDC	FONILEX	1273	**86%**	85%	52%

still better than randomly selecting a variant. This is an indication that our approach to prune words which do not occur in the speech database, is justified.

If we compare the results across speech databases, we can see that in general larger databases yield better performance. Performance across the RJS, AWDC and AVKH databases is quite similar for the decision tree-based method. The difference in accuracy between the AWB and SLT database might be explained by the Scottish accent of the speaker in the AWB speech database. We have not examined whether using the Scottish variant of the Unisyn lexicon would yield better performance.

7 Conclusion

In this paper we presented an approach to automatically generate a speaker-specific pronunciation for speech synthesis based on lexicon pruning and decision tree-based variant selection. Our approach allows the fully automatic prediction of these variants, without the need of the synthetic voice developer to know the language in particular. The impact of the lexicon pruning is quite limited if the original lexicon contains few words with multiple pronunciations, and seems more appropriate for lexicons with a relatively large number of variants, such as the FONILEX or Unisyn lexicons. Our approach may lead to fewer pronunciation differences between the front-end and back-end of the synthesizer, as the speaker-specific pronunciation is generated using the actual segmentation labels in the speech database. A lower amount of these differences can result in an improved speech synthesis quality [4].

Several improvements are still possible. As relatively standard features were used to construct the decision trees for prediction, a potentially larger improvement can be obtained by the use of more advanced linguistic features. Our current segmentation algorithm could be extended to take lexical stress (e.g. [16]) and syllable boundaries into account. Furthermore, the success of the lexicon pruning and pronunciation variant selection depends partly on the accuracy of the segmentation (i.e. whether the correct variant is actually chosen in the speech database). We have not yet examined the influence of the segmentation accuracy in our experiments. Our approach can also be implemented differently: the existing lexicon is kept and decision trees are constructed for each word with multiple pronunciations in the original lexicon. This has the advantage that the same lexicon can be reused for multiple speakers, while still allowing a speaker-specific pronunciation.

References

1. Fitt, S.: Unisyn multi-accent lexicon, version 1.3, http://www.cstr.ed.ac.uk/projects/unisyn
2. Mertens, P., Vercammen, F.: FONILEX manual. Technical report, K.U.Leuven CCL (1998)
3. Kim, Y.J., Syrdal, A., Conkie, A.: Pronunciation lexicon adaptation for TTS voice building. In: Proceedings Interspeech 2004, Jeju Island, Korea, pp. 2569–2572 (2004)
4. Hamza, W., Eide, E., Bakis, R.: Reconciling pronunciation differences between the front-end and the back-end in the IBM speech synthesis system. In: Proceedings Interspeech 2004, Jeju Island, Korea, pp. 2561–2564 (2004)
5. Clark, R.A.J., Richmond, K., King, S.: Multisyn: Open-domain unit selection for the festival speech synthesis system. Speech Communication 49, 317–330 (2007)
6. Bennett, C., Black, A.: Prediction of pronunciation variations for speech synthesis: A data-driven approach. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing 2005 (ICASSP 2005), Philadelphia, PA, USA, vol. 1, pp. 297–300 (2005)
7. Weide, R.L.: The carnegie mellon university pronouncing dictionary, version 0.4 (1995)
8. Mitton, R.: A description of a computer-usable dictionary file based on the oxford advanced learner's dictionary of current english. Technical report, Oxford Text Archive (1992)
9. Kerkhoff, J., Marsi, E.: NeXTeNS: a new open source text-to-speech system for dutch. In: 13th Meeting of Computational Linguistics in the Netherlands (2002)
10. Baayen, R.H., Piepenbrock, R., Gulikers, L.: The CELEX lexical database (CD-ROM). Technical report, Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA (1995)
11. Demuynck, K., Roelens, J., Compernolle, D.V., Wambacq, P.: SPRAAK: an open source "SPeech recognition and automatic annotation kit". In: Proceedings Interspeech 2008, Brisbane, Australia, p. 495 (2008)
12. Kominek, J., Black, A.W.: The CMU arctic speech databases. In: Proceedings Fifth ISCA Workshop on Speech Synthesis (SSW5). ISCA (2004)
13. King, S., Karaiskos, V.: The blizzard challenge 2010. In: Blizzard Challenge Workshop 2010 (2010)
14. Mattheyses, W., Latacz, L., Verhelst, W.: Auditory and photo-realistic audiovisual speech synthesis for dutch. In: Proceedings International Conference on Auditory-Visual Speech Processing 2011 (AVSP 2011), Volterra, Italy, pp. 55–60 (2011)
15. Duchateau, J., Kong, Y.O., Cleuren, L., Latacz, L., Roelens, J., Samir, A., Demuynck, K., Ghesquière, P., Verhelst, W., et al.: Developing a reading tutor: Design and evaluation of dedicated speech recognition and synthesis modules. Speech Communication 51, 985–994 (2009)
16. Van Dalen, R.C., Wiggers, P., Rothkrantz, L.J.M.: Lexical stress in continuous speech recognition. In: Proceedings Interspeech 2006, Pittsburgh, PA, USA (2006)