AUTOMATIC GENERATION OF A PRONUNCIATION DICTIONARY BASED ON A PRONUNCIATION NETWORK

Toshiaki Fukada Yoshinori Sagisaka

ATR Interpreting Telecommunications Research Laboratories
2–2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02 Japan
Tel: +81 774 95 1301, FAX: +81 774 95 1308, E-mail: fukada@itl.atr.co.jp

ABSTRACT

In this paper, we propose a method for automatically generating a pronunciation dictionary based on a pronunciation neural network that can predict plausible pronunciations (*alternative pronunciations*) from the canonical pronunciation. This method can generate multiple forms of alternative pronunciations using the pronunciation network for words that only occur a few times in the database and even for unseen words. Experimental results on spontaneous speech show that the automatically-derived pronunciation dictionaries give consistently higher recognition rates and require less computational time for recognition than a conventional dictionary.

1. INTRODUCTION

In spontaneous speech, word pronunciation varies more than in read speech, but in most spontaneous speech recognition systems, actual pronunciation variations are disregarded and only standard pronunciations in citation form (*canonical pronunciations*) are used. It has been confirmed that an appropriate pronunciation dictionary constructed by hand or by a rule-based system improves recognition performance [1], but such dictionaries require time and expertise to construct. Consequently, research efforts have been conducted to construct a pronunciation dictionary that is automatically trained with real speech data [2] $\sim$ [6]. In these previous approaches, the pronunciation of each word is individually generated using words spoken by many different speakers. Although these methods can take into account pronunciation variations for each word, they have the following disadvantages: (1) alternative pronunciations generated from a small amount of word utterances are unreliable; (2) in spontaneous speech such as a Switchboard corpus, it cannot be guaranteed that a database has a sufficient number of word utterances; (3) it is difficult to construct a pronunciation for unseen words such as proper nouns.

In this paper, we propose a method for automatically generating a pronunciation dictionary based on a pronunciation neural network that can predict plausible pronunciations (*alternative pronunciations*) from the canonical pronunciation. As the pronunciation network is not trained for each word but is trained using all training data, the network can generate multiple forms of alternative pronunciations for words that only occur a few times in the database and even for unseen words.

Research approaches based on a phoneme confusion matrix [7][8] or automatic phonological rule generation [9] often use all training data to cope with pronunciation variability. These approaches, however, only consider the phonetic context of the preceding phone and the following phone (i.e. triphone) because it is difficult to make use of larger phonetic context windows (e.g. quinphone) due to the problem of sparse data or an increase in computational complexity. This would be disadvantageous to generating pronunciation dictionaries for spontaneous speech whose coarticulation is much larger than those of read speech. On the other hand, in the proposed method, a larger phonetic context can be readily considered because pronunciation variabilities are incorporated into a single network.

2. AUTOMATIC GENERATION OF A PRONUNCIATION DICTIONARY

2.1. Generating Alternative Pronunciation Strings

First, alternative pronunciation string generation and mapping to the canonical pronunciation are performed as follows.

1. Conduct phoneme recognition using speech training data for dictionary generation. Recognized phoneme strings are taken as an alternative pronunciation.
2. Align the canonical pronunciation to the alternative pronunciation using a dynamic programming algorithm.

For example, if the result is

```
a r a y u r     u   (canonical pron.)
a w a     u r i u   (alternative pron.),
```

the correspondence between the canonical pronunciation and the alternative pronunciation is as follows: /a/→/a/, /r/→/w/, /a/→/a/, /y/→/ /(deletion), /u/→/u/, /r/→/r i/(i is inserted), /u/→/u/. These results are used as input and output data for the pronunciation neural network training described in the following section.

2.2. Pronunciation Network Training

A pronunciation network is trained using a multilayer perceptron to predict alternative pronunciation $A(m)$ from the five phonemes (i.e. quinphone) of canonical pronunciations $L(m-2), \ldots, L(m+2)$. Figure 1 shows the network structure, which has a structure similar to that employed in NETtalk [10].

$L(m-2), \ldots, L(m+2)$ are given for the pronunciation network inputs; $A(m)$ aligned to $L(m)$ are given for the outputs. A total of 130 units (26 Japanese phoneme sets times 5 contexts) are used in the input layer. The representation of alternative pronunciations at the output layer is localized, with one unit representing deletion, 26 units

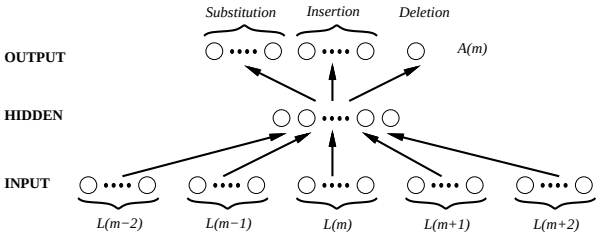

Figure 1. Pronunciation network.

for substitution and 26 units for insertion, providing a total of 53 output units.

In the previous example, / /(deletion), which corresponds to the 4-th canonical string /y/, is used as $A(m)$, and /r a y u r/ are used as $L(m-2), \ldots, L(m+2)$. Here, 1.0 is given as the output unit for deletion and as the input units for the /r/ in $L(m-2)$, /a/ in $L(m-1)$, etc.; 0.0 is given for the other units.

If 100 hidden units are used, the total number of network weights becomes about 18,000. This is much fewer than the number of confusion matrix based weights, 1.4×10^6, needed for the quinphone case under the Japanese phonotactic constraint.

2.3. Automatic Dictionary Generation

By using the pronunciation network, a variety of pronunciation dictionaries can be automatically generated. In this paper, the following three types of dictionaries are constructed based on the output from the pronunciation network.

1. An alternative pronunciation derived from the maximum number of outputs is stored in the dictionary (here denoted as **N(single)**).
2. Both **N(single)** and the canonical pronunciation are stored. If these pronunciations are the same, a single pronunciation is registered (**N(single)+C**).
3. A maximum number of N-best candidates based on the output values of the network are stored as multiple forms of alternative pronunciations (**N(multi)**).

The number of maximum candidates, N, in **N(multi)** was set to 8. Additionally, a threshold (0.03) was used in **N(multi)** to prevent storing implausible candidates. These procedures for dictionary generation are applied to word lexicons of more than five phonemes. For the beginning or end of two phonemes, the canonical pronunciations are used. Flowcharts of the procedures for generating a pronunciation dictionary with methods 1, 2 and 3 are shown in Figs. 2, 3 and 4, respectively.

2.4. Pronunciation Dictionary for Spontaneous Speech Recognition

Japanese spontaneous speech uttered by one male speaker (1,530 utterances; 100,000 phonemes) and contained in ATR spontaneous speech database [11] was used as the training data. First, shared-state context-dependent HMMs were generated from the training data [12]. Then, phoneme recognition was performed using the HMMs under the Japanese phonotactic constraint. The recognition results were used as alternative pronunciations. The pronunciation network, whose number of hidden units was set to 100, was trained as described in **2.2.**. Training was done using 200 iterations.

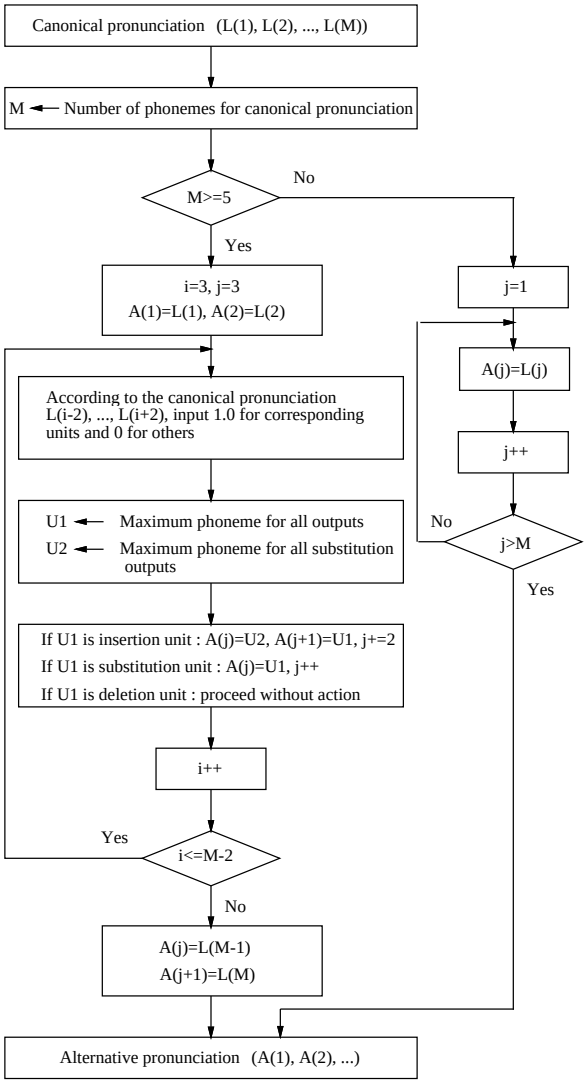

Figure 2. Pronunciation dictionary generation based on maximum output value of the network (method 1).

Three types of pronunciation dictionaries for the 6,635 words used in section **3.** were constructed. Table 1 shows examples of substitution, insertion and deletion. Table 2 shows examples of alternative pronunciations derived from the canonical pronunciation /k a m o g a w a r y o k a ng/.

3. EXPERIMENTS

3.1. Conditions

To compare the three kinds of dictionaries, we performed continuous speech recognition experiments for Japanese spontaneous speech using a 6,635-word recognizer [13]. The dictionary with canonical pronunciation **C** was also used to obtain baseline results. These four dictionaries were tested for the following three cases.

- speaker-dependent model generated from one male speaker (SD)
- speaker-independent model (SI)
- speaker-adapted model from SI model (SA) [14]

Table 1. Examples of pronunciation network output.

	Canonical pronunciation (input)	Alternative pronunciation (output)
substitution	k a ng *z* e e (*customs*)	k a ng *d* e e
insertion	w a r *i* a i (*rate*)	w a r *i y* a i
deletion	g o *y* o y a k u (*reservation*)	g o o y a k u

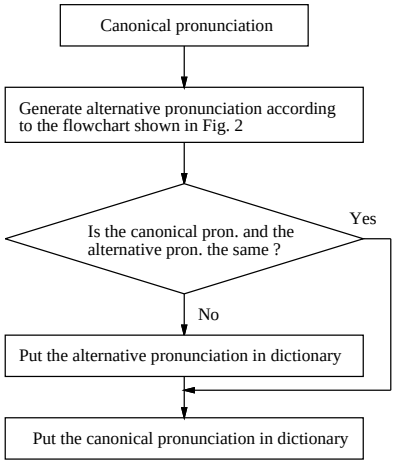

Figure 3. Pronunciation dictionary generation based on maximum output value of the network and canonical pronunciation (method 2).

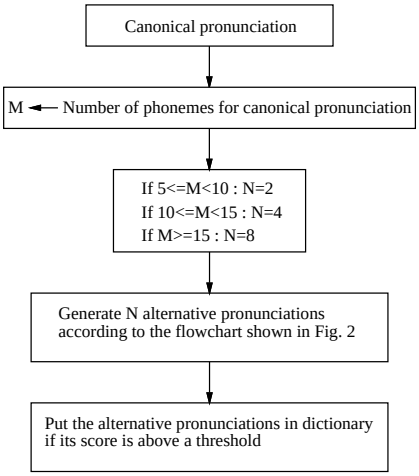

Figure 4. Pronunciation dictionary generation based on N-best candidates of the network (method 3).

In each of the experiments, context-dependent HMMs [12] and an n-gram language model [15] were used.

41 utterances for SD (one male speaker) and 98 utterances (7 speakers) for SI and SA were used as test data. In each case, decoding parameters (e.g. language model scale factor or beam width) were set to fixed values that gave the best recognition performance with dictionary **C**. Note that three types of pronunciation dictionaries (**N(single)**, **N(single)+C** and **N(multi)**) were generated from the speech data of one male and were commonly tested for the three cases. Dictionary sizes are listed in Table 3.

Table 2. Examples of automatically-derived pronunciation dictionaries.

Dictionary	Alternative pronunciation
N(single)	k a m o a a r y o k a ng
N(single)+C	k a m o a a r y o k a ng
	k a m o g a w a r y o k a ng
N(multi)	k a m o a a r y o k a ng
	k a m o a w a r y o k a ng
	k a m o a m a r y o k a ng

Table 3. Dictionary size.

C	N(single)	N(single)+C	N(multi)
6,635	6,635	7,854	14,324

Table 4. Recognition results (word accuracy %).

Dictionary	C	N(single)	N(single)+C	N(multi)
SD	19.98	20.82	21.07	24.46
SI	12.19	12.89	16.20	19.37
SA	27.39	28.16	32.41	32.56

3.2. Results

Recognition results are shown in Table 4. From these results, it can be seen that the proposed dictionaries (**N(single)**, **N(single)+C** and **N(multi)**) gave better performances than the conventional dictionary (**C**). Table 5 shows recognition time. Although the multiple pronunciation dictionaries (**N(single)+C** or **N(multi)**) had more word entries than **C**, these dictionaries did not require greater recognition time. For SI and SA, actual decreases of about 20∼30% could be achieved by using **N(single)+C** or **N(multi)**. One reason might be that the acoustic likelihoods for words represented by the pronunciations of **N(single)+C** or **N(multi)** were higher than those of **C**. As a result, many hypotheses could be pruned from the beam during recognition. This indicates that the automatically-derived pronunciation dictionaries effectively represent pronunciation variations in spontaneous speech.

3.3. Comparison with Confusion Matrix Based Approach

To compare the previous recognition performances with other approaches, we generated pronunciation dictionaries based on a phoneme confusion matrix based approach as follows.

3.3.1. Phoneme confusion matrix based pronunciation dictionary generation

First, a context-dependent phoneme confusion matrix was constructed according to the phoneme recognition results. Second, for each context, the most frequent result was taken as the pronunciation variation rule. Then, the pronunciation variation rules were applied to a 6,635 word

Table 5. Recognition time (sec.). Normalized time with respect to first column in brackets.

Acoustic model	Utterances	C	N(single)	N(single)+C	N(multi)
SD	195.5	104.1 (1)	103.8 (1.00)	106.7 (1.02)	104.1 (1.00)
SI	320.7	3,650 (1)	2,932 (0.80)	3,021 (0.83)	2,530 (0.69)
SA	320.7	1,497 (1)	1,530 (1.02)	1,196 (0.80)	1,138 (0.76)

Table 6. Recognition results using a confusion matrix derived pronunciation dictionary (word accuracy %, triphone/quinphone).

Dictionary	C	M(single)	M(single)+C
SD	19.98	18.60/21.71	22.40/23.09
SI	12.19	10.73/13.04	14.04/12.27
SA	27.39	25.54/25.39	29.48/30.86

lexicon to generate a phoneme confusion matrix based pronunciation dictionary **M(single)**. Canonical pronunciations were used for unseen contexts. These procedures were applied to word lexicons of more than five phonemes and the canonical pronunciations were used for the beginning or end of two phonemes. Similar to the proposed dictionary **N(single)+C**, dictionary **M(single)+C** was also generated by registering both **M(single)** and the canonical pronunciation.

Two types of context, triphone and quinphone, were considered. The dictionary sizes became 6,635 for dictionary **M(single)**, 8,498 for triphone context dictionary **M(single)+C** and 7,566 for quinphone context dictionary **M(single)+C**.

3.3.2. Recognition results

By using the phoneme confusion matrix dictionaries **M(single)** and **M(single)+C**, speech recognition experiments were performed under the same conditions as in **3.1.**. Recognition results are listed in Table 6.

Compared to the results shown in Table 4, the proposed dictionaries (**N(single)** and **N(single)+C**) gave better recognition performances than those of the phoneme confusion matrix based dictionaries (**M(single)** and **M(single)+C**), case SD being the exception. The reason might be that in the proposed approach, degrees of freedom are controlled by the structure of the pronunciation network or the number of hidden units, while the confusion matrix approach uses all of the recognition results as pronunciation variation rules without any constraints. As a result, the proposed dictionaries mainly represent speaker-independent pronunciation variations.

4. CONCLUSION

In this paper, we have proposed a method for automatically generating a pronunciation dictionary based on a pronunciation neural network. This method can generate multiple forms of alternative pronunciations even for unseen words. Experimental results on spontaneous speech showed that the automatically-derived pronunciation dictionaries gave consistently higher recognition rates and required less recognition time than the conventional dictionary. We expect the multiple pronunciation dictionary to be a useful resource for acoustic model retraining by realigning the training data [4][6].

REFERENCES

[1] L. Lamel and G. Adda : "On designing pronunciation lexicons for large vocabulary, continuous speech recognition," *Proc. ICSLP-96*, pp. 6–9, 1996.

[2] P. Schmid, R. Cole and M. Fanty : "Automatically generated word pronunciations from phoneme classifier output," *Proc. ICASSP-93*, pp. II-223–II-226, 1993.

[3] C. Wooters and A. Stolcke : "Multiple-pronunciation lexical modeling in a speaker independent speech understanding system," *Proc. ICSLP-94*, pp. 1363–1366, 1994.

[4] T. Sloboda : "Dictionary learning : performance through consistency," *Proc. ICASSP-95*, pp. 453–456, 1995.

[5] J. Humphries, P. Woodland and D. Pearce : "Using accent-specific pronunciation modelling for robust speech recognition," *Proc. ICSLP-96*, pp. 2324–2327, 1996.

[6] E. Fosler : "Automatic learning of word pronunciation from data," *Proc. ICSLP-96*, pp. 28–29 (addendum), 1996.

[7] A. Ito and S. Makino : "A new word pre-selection method based on an extended redundant hash addressing for continuous speech recognition," *Proc. ICASSP-93*, pp. II-299–II-302, 1993.

[8] D. Torre, L. Villarrubia, L. Hernandez and J. Elvira : "Automatic alternative transcription generation and vocabulary selection for flexible word recognizers," *Proc. ICASSP-97*, pp. 1463–1466, 1997.

[9] T. Imai, A. Ando and E. Miyasaka : "A new method for automatic generation of speaker-dependent phonological rules," *Proc. ICASSP-95*, pp. 864–867, 1995.

[10] T. Sejnowski and C. Rosenberg : "NETtalk: a parallel network that learns to read aloud," The Johns Hopkins Univ. Electrical Engineering and Computer Science Tech. Report JHU/EECS-86/01, 1986.

[11] A. Nakamura, S. Matsunaga, T. Shimizu, M. Tonomura and Y. Sagisaka : "Japanese speech databases for robust speech recognition," *Proc. ICSLP-96*, pp. 2199–2202, 1996.

[12] J. Takami and S. Sagayama : "A Successive State Splitting Algorithm for Efficient Allophone Modeling," *Proc. ICASSP-92*, pp. 573–576, 1992.

[13] T. Shimizu, H. Yamamoto, H. Masataki, S. Matsunaga and Y. Sagisaka : "Spontaneous dialogue speech recognition using cross-word context constrained word graphs," *Proc. ICASSP-96*, pp. 145–148, 1996.

[14] K. Ohkura, M. Sugiyama and S. Sagayama : "Speaker adaptation based on transfer vector field smoothing with continuous mixture density," *Proc. ICSLP-92*, pp. 369–372, 1992.

[15] H. Masataki and Y. Sagisaka : "Variable-order n-gram generation by word-class splitting and consecutive word grouping," *Proc. ICASSP-96*, pp. 188–191, 1996.