Connectionist Approaches to Large Vocabulary Continuous Speech Recognition

Hidefumi SAWAI, Member, Yasuhiro MINAMI†††, Nonmember, Masanori MIYATAKE††††, Member, Alex WAIBEL††††, Nonmember and Kiyohiro SHIKANO†††††, Member

SUMMARY This paper describes recent progress in a connectionist large-vocabulary continuous speech recognition system integrating speech recognition and language processing. The speech recognition part consists of Large Phonemic Time-Delay Neural Networks (TDNNs) which can automatically spot all 24 Japanese phonemes (i.e., 18 consonants /b, d, g, p, t, k, m, n, s, sh/ [j], h, z, zh, [j], ts, tr, w, y/ [j]) and 5 vowels /a, i, u, e, o/ and a double consonant /Q/ or silence by simply scanning among input speech without any specific segmentation techniques. On the other hand, the language processing part is made up of a predictive LR parser in which the LR parser is guided by the LR parsing table automatically generated from context-free grammar rules, and proceeds left-to-right without backtracking. Time alignment between the predicted phonemes and a sequence of the TDNN phoneme outputs is carried out by the DTW matching method. We call this hybrid integrated recognition system the TDNN-LR method. We report that large-vocabulary isolated word and continuous speech recognition using the TDNN-LR method provided excellent speaker-dependent recognition performance, where incremental training using a small number of training tokens is found to be very effective for adaptation of speaking rate. Furthermore, we report some new achievements as extensions of the TDNN-LR method: (1) two proposed NN architectures provide robust phoneme recognition performance on variations of speaking manner. (2) r speaker-adaptation technique can be realized using a NN mapping function between input and standard speakers and (3) new architectures proposed for speaker-independent recognition provide performance that nearly matches speaker-dependent recognition performance.

1. Introduction

In this paper, we describe recent progress in the connectionist large-vocabulary continuous speech recognition system we have developed at ATR. First, we review our research achievements: phoneme recogni-
matching method.

In Sect. 5, two kinds of recognition experiments i.e., large-vocabulary isolated word recognition and continuous speech recognition, were performed using the TDNN-LR method. In the large-vocabulary isolated word recognition, 5,240 common Japanese words are used. In the continuous speech recognition, 278 test phrases in the "ATR conference registration task" are recognized.

In Sect. 6, we propose some new connectionist approaches to robust speech recognition, speaker-adaptation and speaker-independent recognition as extensions of the TDNN-LR method: two proposed NN architectures provide robust phoneme recognition performance on variations of speaking manner; a speaker-adaptation technique can be realized using a NN mapping function between input and standard speakers; and new architectures proposed for speaker-independent recognition provide performance that nearly matches speaker-dependent recognition performance.

2. Phoneme Recognition Using TDNN

2.1 TDNN Architecture

For the recognition of phonemes, a three-layer net is constructed. Its overall architecture and a typical set of activities in the units are shown in Fig. 1 based on one of the phonemic subcategory tasks (BDG).

At the lowest level, 16 mel-scale spectral coefficients serve as input to the network. Input speech, sampled at 12 kHz, was hamming windowed and a 256-point FFT computed every 5 msec. Mel-scale coefficients were computed from the power spectrum and coefficients adjacent in time collapsed resulting in an overall 10 msec frame rate. The coefficients of an input token (in this case 15 frames of speech centered around the hand labeled vowel onset) were then normalized to lie between -1.0 and +1.0 with the average at 0.0. Figure 1 shows the resulting coefficients for the speech token "BA" as input to the network, where positive values are shown as black squares and negative values as grey squares. The detailed architecture is described in Ref. (1). The TDNN achieved a recognition rate of 98.5% averaged for three male speakers.

2.2 Consonant Recognition by Modular TDNN Design

Our consonant TDNN (shown in Fig. 2) was constructed modularly from networks aimed at the consonant subcategories, i.e., the bdg-, ptk-, mmn-, shsh-, chch- and the rsw-tasks. Each of these nets had been trained before to discriminate between the consonants within each class. In addition, an interclass discrimination net that distinguishes between the consonant subcategories was trained. This hopefully provides missing feature information for interclass discrimination. Three connections were then established to each of the 18 consonant output categories /b/, /d/, /g/, /p/, /t/, /k/, /m/, /n/, /n.s., sh/ (\{j\}), /n., /z/, /ch/ (\{\theta\}), /ts/, /r/, /w/ and /y (\{\j\}): one to connect an output unit with the appropriate interclass discrimination unit in hidden layer 2, one with the appropriate intraclass discrimination unit from hidden layer 2 of the corresponding subcategory net and one with the always-activated threshold unit (not shown in Fig. 2). The overall network architecture is illustrated in Fig. 2 for the case of an incoming test token (e.g., a /g/). The performance of the network yielded 96.0% correct consonant recognition over the test data (99-105). Furthermore, a fast back-propagation method later developed at ATR made it possible to train the consonant network from random weight values at the same time, and yielded a better recognition rate of 96.7% (98).

3. Phoneme Spotting Using TDNN

A large TDNN architecture for discriminating 24
Fig. 2  Modular construction of all consonant network.

Fig. 3  The large phonemc TDNN architecture.
Japanese phonemes (18 consonants: /b/, /d/, /g/, /p/, /t/, /k/, /m/, /n/, /n/, /s/, /sh/, /h/, /z/, /ch/, /tj/, /ts/, /r/, /w/, /y/; and 5 vowels /a/, /i/, /u/, /e/, /o/, and silence) was constructed as shown in Fig. 3. The TDNN is modulated by 6 inner-class subnetworks discriminating among “bgd”, “ptk”, “mnN”, “shsh”, “chch” and “rwy”, an inner-class subnetwork discriminating among consonant groups, a vowel network, and a silence network discriminating between silence and speech. These subnetworks are integrated into a third hidden layer which has 24 units so that their corresponding output units can be laterally inhibited.

Phoneme tokens for training the TDNN are classified into 24 phoneme categories, based on hand labeling, extracted from even-numbered words of the 5,240 common words uttered by a male speaker. The number of training tokens per phoneme category ranges up to 1,000, randomly selected from the extracted tokens. Tokens are duplicated when the number per category can not reach 1,000. Training the TDNN is performed using a fast back-propagation learning procedure.

Phoneme spotting outputs are obtained as recognition results by shifting the input layer among input speech frame by frame. This phoneme spotting method does not require any phoneme segmentation techniques and can get spotting results merely by scanning the network.

Table 1 shows spotting results for 2,620 test words when using up to 400 and 1,000 training tokens, respectively. It is demonstrated that 98.0% of the phonemes in the test words are correctly spotted for the latter case, yielding a false alarm rate of 23.2%.

Thus, the Large Phonemic TDNN is already trained by as many as 18,864 training tokens extracted from 2,620 training words. For the first experiment, spotting experiments in continuous speech were conducted using the TDNN. The initial correct phoneme spotting rate in 278 Japanese test phrases was 81.2% with a false alarm rate of 47.8%, as shown in Table 2. Because of the different co-articulatory effects of word speech and continuous speech, incremental TDNN training using a small number of tokens extracted from continuous training speech seemed to be needed. The number of tokens for incremental training is only 100/200 tokens per phoneme category (2,011/3,251 tokens are only 11%/17% of the original tokens extracted from the training words). The correct phoneme spotting rate was significantly improved from 81.2% to 89.0%/89.1% after the adaptive incremental training. More importantly, the false alarm rate decreased from 47.8% to 34.8%/25.8%. Figure 4 shows an example of phoneme spotting results in the phrase "toukoku-wo/". The lower layer shows an input spectrogram and the upper layer shows spotting outputs. We can also expect better phrase recognition rates in continuous speech after the incremental training.

Table 1 TDNN phoneme spotting results on large-vocabulary.

<table>
<thead>
<tr>
<th>Phonemes</th>
<th>No of phonemes</th>
<th>400 tokens/call</th>
<th>1,000 tokens/call</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Correct</td>
<td>Deletion</td>
</tr>
<tr>
<td>b</td>
<td>247</td>
<td>228</td>
<td>3</td>
</tr>
<tr>
<td>d</td>
<td>180</td>
<td>175</td>
<td>5</td>
</tr>
<tr>
<td>g</td>
<td>266</td>
<td>230</td>
<td>35</td>
</tr>
<tr>
<td>p</td>
<td>26</td>
<td>25</td>
<td>3</td>
</tr>
<tr>
<td>t</td>
<td>461</td>
<td>452</td>
<td>0</td>
</tr>
<tr>
<td>k</td>
<td>1,000</td>
<td>1,218</td>
<td>82</td>
</tr>
<tr>
<td>m</td>
<td>488</td>
<td>482</td>
<td>3</td>
</tr>
<tr>
<td>n</td>
<td>273</td>
<td>255</td>
<td>15</td>
</tr>
<tr>
<td>N</td>
<td>488</td>
<td>487</td>
<td>1</td>
</tr>
<tr>
<td>s</td>
<td>572</td>
<td>570</td>
<td>2</td>
</tr>
<tr>
<td>sh</td>
<td>387</td>
<td>285</td>
<td>2</td>
</tr>
<tr>
<td>h</td>
<td>212</td>
<td>212</td>
<td>2</td>
</tr>
<tr>
<td>z</td>
<td>315</td>
<td>310</td>
<td>5</td>
</tr>
<tr>
<td>zh</td>
<td>141</td>
<td>140</td>
<td>1</td>
</tr>
<tr>
<td>ts</td>
<td>220</td>
<td>219</td>
<td>1</td>
</tr>
<tr>
<td>r</td>
<td>760</td>
<td>709</td>
<td>51</td>
</tr>
<tr>
<td>w</td>
<td>61</td>
<td>80</td>
<td>1</td>
</tr>
<tr>
<td>y</td>
<td>573</td>
<td>531</td>
<td>42</td>
</tr>
<tr>
<td>a</td>
<td>172</td>
<td>170</td>
<td>2</td>
</tr>
<tr>
<td>i</td>
<td>1335</td>
<td>1282</td>
<td>51</td>
</tr>
<tr>
<td>u</td>
<td>1615</td>
<td>1496</td>
<td>119</td>
</tr>
<tr>
<td>e</td>
<td>1225</td>
<td>822</td>
<td>7</td>
</tr>
<tr>
<td>o</td>
<td>1952</td>
<td>1677</td>
<td>15</td>
</tr>
<tr>
<td>Total</td>
<td>1,574</td>
<td>1,518</td>
<td>456</td>
</tr>
</tbody>
</table>

Note: The numbers in parentheses indicate the false alarm rates.
4. The TDNN-LR Recognition System

To extend the high performance spotting results to large-vocabulary continuous speech recognition, a "hybrid" method combining a predictive LR parser\(^{17}\) with a DTW alignment technique was proposed. We applied this method to 5,240 common Japanese words and phrases\(^{16}\) uttered by the male speaker.

4.1 LR Parsers

LR parsing is well known in the field of program languages, and is applicable to a large class of context-free grammars. Generalized LR parsing\(^{16}\) is a kind of LR parsing, and has been extended to handle arbitrary context-free grammars. The LR parser is guided by the LR parsing table automatically created from context-free grammar rules, and proceeds left-to-right without backtracking. These parsing algorithms are very efficient for natural language processing.

An example of sentences in the LR parsing and an LR parsing table are shown in Figs. 5 and 6, respectively\(^{17}\). The LR table consists of an ACTION table and a GOTO table. In Fig. 6, lines show grammar symbols, and columns show parser status. The symbols "s" and "r" show "shift" and "reduce" actions, respectively.

---

**Fig. 4** An example of phoneme spotting results (phrase name is /tourokugo/).

**Fig. 5** An example of context-free grammar.

**Fig. 6** An example of ACTION and GOTO tables.
The figure on the right side of "S" is the next status in a "shift" action. The figure on the right side of "r" is the number of grammar rules. The right side shows a GOTO table where the figure indicates the next status value.

A predictive LR parsing method predicts the next phonemes in input speech based on the currently processed phonemes. An HMM continuous speech recognition system using a predictive LR parsing has been evaluated. This technique is also applicable to spotting results from the TDNN and a word or phrase grammar describing a large vocabulary or phrase database, respectively. Prediction can be easily realized by referencing an LR table such as Fig. 6. By way of analysis, when the predictive LR parser is in a status, possible phonemes served to this LR parser are the only phonemes described by "shift" and "reduce" on a line of the table. The predictive LR parser regards these phonemes as predicted phonemes.

4.2 Integration of the TDNN and the Parser

The basic structure of the recognition system which utilizes TDNN spotting and predictive LR parsing is shown in Fig. 7 (hereafter: TDNN-LR). First, an input speech is converted to outputs via TDNN phoneme spotting shown in the upper part of Fig. 4. Matching between these outputs and reference words is performed by the predictive LR parser according to a grammar rule. When plural phonemes are predicted, the predictive LR parser analyzes the phonemes in parallel. The predicted phoneme sequences are evaluated by a DP match between predicted phonemes and the TDNN phoneme spotting results. This procedure continues until input phonemes come to an end. However, since it takes considerable time to process all predicted phonemes, a beam search is used to take the first "B" candidates, where "B" is the width of the beam (ex.: B = 100).

The likelihood of a similarity between a predicted

![Fig. 7 The TDNN-LR speech recognition system.](image)

Table 2: TDNN phoneme spotting results on test phrases.

<table>
<thead>
<tr>
<th>Phoneme</th>
<th>ref phonemes</th>
<th>No adaptive training</th>
<th>Adaptive training (100 tokens)</th>
<th>Correct</th>
<th>Deletion</th>
<th>False alarm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Correct</td>
<td>Deletion</td>
<td></td>
<td></td>
<td>False alarm</td>
</tr>
<tr>
<td>h</td>
<td>16</td>
<td>16</td>
<td>2</td>
<td>19</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>e</td>
<td>68</td>
<td>68</td>
<td>4</td>
<td>25</td>
<td>62</td>
<td>7</td>
</tr>
<tr>
<td>a</td>
<td>34</td>
<td>34</td>
<td>15</td>
<td>36</td>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td>i</td>
<td>10</td>
<td>10</td>
<td>8</td>
<td>2</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>u</td>
<td>70</td>
<td>70</td>
<td>46</td>
<td>11</td>
<td>68</td>
<td>2</td>
</tr>
<tr>
<td>o</td>
<td>210</td>
<td>210</td>
<td>182</td>
<td>61</td>
<td>105</td>
<td>15</td>
</tr>
<tr>
<td>y</td>
<td>65</td>
<td>65</td>
<td>56</td>
<td>22</td>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td>n</td>
<td>74</td>
<td>74</td>
<td>56</td>
<td>38</td>
<td>53</td>
<td>41</td>
</tr>
<tr>
<td>N</td>
<td>34</td>
<td>34</td>
<td>24</td>
<td>10</td>
<td>27</td>
<td>7</td>
</tr>
<tr>
<td>s</td>
<td>73</td>
<td>73</td>
<td>65</td>
<td>7</td>
<td>72</td>
<td>2</td>
</tr>
<tr>
<td>sh</td>
<td>33</td>
<td>33</td>
<td>59</td>
<td>3</td>
<td>55</td>
<td>0</td>
</tr>
<tr>
<td>h</td>
<td>27</td>
<td>27</td>
<td>14</td>
<td>13</td>
<td>24</td>
<td>6</td>
</tr>
<tr>
<td>sh</td>
<td>32</td>
<td>32</td>
<td>22</td>
<td>0</td>
<td>32</td>
<td>0</td>
</tr>
<tr>
<td>nh</td>
<td>17</td>
<td>17</td>
<td>10</td>
<td>7</td>
<td>18</td>
<td>1</td>
</tr>
<tr>
<td>ts</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>0</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>r</td>
<td>66</td>
<td>66</td>
<td>53</td>
<td>13</td>
<td>41</td>
<td>5</td>
</tr>
<tr>
<td>w</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>0</td>
<td>23</td>
<td>2</td>
</tr>
<tr>
<td>y</td>
<td>96</td>
<td>96</td>
<td>80</td>
<td>16</td>
<td>99</td>
<td>6</td>
</tr>
<tr>
<td>a</td>
<td>210</td>
<td>210</td>
<td>238</td>
<td>41</td>
<td>272</td>
<td>7</td>
</tr>
<tr>
<td>i</td>
<td>192</td>
<td>192</td>
<td>160</td>
<td>22</td>
<td>179</td>
<td>13</td>
</tr>
<tr>
<td>u</td>
<td>97</td>
<td>97</td>
<td>99</td>
<td>7</td>
<td>267</td>
<td>85</td>
</tr>
<tr>
<td>e</td>
<td>127</td>
<td>127</td>
<td>107</td>
<td>20</td>
<td>121</td>
<td>6</td>
</tr>
<tr>
<td>o</td>
<td>256</td>
<td>256</td>
<td>234</td>
<td>22</td>
<td>237</td>
<td>19</td>
</tr>
<tr>
<td>total</td>
<td>1940</td>
<td>1940</td>
<td>1940</td>
<td></td>
<td>1940</td>
<td></td>
</tr>
</tbody>
</table>

| (21.2%) | (10.8%) | (27.8%) | (68.1%) | (10.8%) | (25.8%) |
phoneme and an input phoneme is defined as the logarithm of the activation value of TDNN output, where the likelihood is regarded as a posterior probability for each output. The length of the reference patterns (predicted phoneme patterns) is the average length of the training phoneme tokens extracted from the training words of the large vocabulary. The slope constraint in DTW alignment is 1/2 to 2. The detailed matching algorithm is described in Refs. (10), (14), (18).

5. Recognition Experiments

5.1 Large-Vocabulary Speech Recognition

In recognition experiments of large vocabulary, 5,240 common Japanese words were used. Among those words, another half of the large database which were not used for the network training were used as test words. The number of test words was incremented as 100, 500, 2,620 test words. On the other hand, the number of reference words was also incremented as 100, 500, 2,620 and 5,240 words, where in the former three cases, the reference words corresponded to the test words, and in the last case, the 5,240 reference words included the 2,620 test words as a subset. Therefore, note that this experiment is vocabulary-independent recognition.

Figure 8 shows the recognition rates of the n-th 
\(1 \leq n \leq 5\) top choices as a function of the vocabulary size of reference words from 100 to 5,240. In the case of the whole 5,240 words, a rate of 92.0% is obtained for the top choices, and rate of 97.6% and 99.1% are obtained for the second and fifth choices, respectively.

Recognition error in 5,240 common words is classified into the following three cases: (1) insertion of "i" or "k" at the beginning of a word (ex.: "aisuru" → "taisuru"). (2) a short word is misrecognized as a long word (ex.: "aa" → "hanahada"). (3) a double consonant is confused with a silence accompanied by an unvoiced stop (affricate) (ex.: "itai" → "ittai").

5.2 Continuous Speech Recognition

The Large Phonemic TDNN is already trained by as many as 18,864 training tokens extracted from 2,620 training words. As an initial attempt, continuous speech recognition experiments were conducted using the trained TDNN and an LR-parser describing general phrase grammar rules (its phoneme-perplexity is 5.9). Table 3 shows the features of the ATR “Conference Registration” task we used. The initial phrase recognition rate for 278 Japanese test phrases was 55.0% for the top choices and 82.7% for the top 5 choices, respectively. Because of different co-articulatory effects between word speech and continuous speech, incremental training of the TDNN using a small number of training tokens extracted from continuous training speech seemed to be needed.

The number of training tokens for incremental training is only 100 tokens per phoneme category (2,011 tokens in total are only 11% of the original tokens extracted from the training words). We then increased the number up to 200 tokens per category (3,251 tokens in total). The phrase recognition rates are shown in Table 4 as compared with the rates before the incremental training. A phrase recognition rate of 65.1% for the top choices and 88.8% for the top 5

Table 3 Features of the task

<table>
<thead>
<tr>
<th></th>
<th>1,005</th>
<th>2,011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of words</td>
<td>1,005</td>
<td>2,011</td>
</tr>
<tr>
<td>Number of phrases</td>
<td>1,856</td>
<td>3,645</td>
</tr>
<tr>
<td>Number of states in LR</td>
<td>5,035</td>
<td></td>
</tr>
<tr>
<td>Phoneme perplexity</td>
<td>5.9</td>
<td></td>
</tr>
<tr>
<td>Entropy/phoneme</td>
<td>2.6 bit</td>
<td></td>
</tr>
<tr>
<td>Average number of tokens per phrase</td>
<td>7.22</td>
<td></td>
</tr>
</tbody>
</table>

Table 4 Phrase recognition rates (%)

<table>
<thead>
<tr>
<th>Rank</th>
<th>Before adaptive training</th>
<th>Before adaptive training with duration context</th>
<th>After adaptive training 100 tokens</th>
<th>After adaptive training 200 tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>52.9</td>
<td>55.0</td>
<td>64.4</td>
<td>66.1</td>
</tr>
<tr>
<td>2</td>
<td>70.1</td>
<td>70.1</td>
<td>79.5</td>
<td>78.4</td>
</tr>
<tr>
<td>3</td>
<td>77.7</td>
<td>76.6</td>
<td>84.2</td>
<td>87.1</td>
</tr>
<tr>
<td>4</td>
<td>81.7</td>
<td>81.3</td>
<td>86.0</td>
<td>88.1</td>
</tr>
<tr>
<td>5</td>
<td>92.4</td>
<td>92.7</td>
<td>98.8</td>
<td>98.8</td>
</tr>
<tr>
<td>6-10</td>
<td>90.3</td>
<td>87.4</td>
<td>93.2</td>
<td>91.0</td>
</tr>
<tr>
<td>11-15</td>
<td>97.4</td>
<td>97.4</td>
<td>98.5</td>
<td>92.4</td>
</tr>
<tr>
<td>16-</td>
<td>99.6</td>
<td>92.6</td>
<td>96.5</td>
<td>76.6</td>
</tr>
</tbody>
</table>
choices were obtained. Therefore, the efficiency of adaptive incremental training using a small number of training tokens extracted from continuous speech was confirmed through this experiment.

Typical errors are as follows:

1. Substitution errors between /n/ and /m/.
   - ex.: /sanka-no/ → /sanka-mo/.
   - /sotet-no/ → /sotet-mo/.

   These errors occurred due to the fact that the number of /m/ and /n/ phoneme tokens for incremental training was too small (17 tokens for /m/ and 13 tokens for /n/), compared with the original training tokens extracted from training words (1,000 for /m/ and 460 for /n/).

2. Phoneme insertion errors.
   - ex.: /zyuusho/ → /zyuusho-o/.
   - /happyou/ → /happyou-o/.

   These errors occurred due to difficulty of precise duration control at the end of the utterances.

6. Extensions

   In this section, we describe extensions in the TDNN-LR speech recognition system: robustness for variations of speaking manner, speaker-adaptation and speaker-independent phoneme recognition.

6.1 Neural Network Architectures for Robust Speech Recognition

   Until now, Time-Delay Neural Networks (TDNN) architecture has been applied to several speaker-dependent recognition stages, such as phoneme recognition (described in Sect. 2), Japanese phoneme spotting (in Sect. 3), and the TDNN-LR large-vocabulary continuous speech recognition system with integrated training for spotting Japanese phonemes (in Sect. 4). If we extend these recognition methods based on TDNN to a continuous, speaker-independent speech recognition system, a novel robust recognition strategy should be developed. This section introduces several novel TDNN architectures for robust speaker-independent, continuous speech recognition.

   One novel architecture for a Frequency-shift-invariant TDNN (FTDNN) is based on the frequency-time-shift-invariance as well as the time-shift-invariance by constructing the same weighting values between the input layer and the hidden layers of the TDNN. Speech features from the input layer of the FTDNN are individually extracted along the time-axis and the mel-scaled frequency-axis by each corresponding first hidden layer. The extracted features are then integrated into a single second hidden layer. The final decision is made based on the activation patterns whose property is invariant from both the time and frequency-shift of input phoneme tokens.

   Another novel architecture is a Block-Windows NN (BWNN), based on windowing each layer of the NN with local time-frequency windows. This architecture makes it possible for the NN to capture global features from the upper layers as well as precise local features from the lower layers, because the local windows in the upper layers can integrate more global features than those in the lower layers. A five layer BWNN is constructed for a phoneme recognition experiment. These architectures yield significantly better recognition performance than the original TDNN.

6.2 Speaker-Adaptation Using Neural Networks

   Speaker-adaptation is one good approach to a speaker-independent recognition problem. It is necessary to use a small amount of training data uttered by an input speaker to adapt a speech recognition system. A speaker-adaptation technique using neural networks have been proposed. It is also possible to use segmental speech for speaker-adaptation by building a mapping function from an input speaker to a standard speaker. We proposed a segmental approach using neural network identity mapping as a supervised learning method. In this approach, segmental speech including a phoneme or syllable can be mapped between two speakers through a neural network and DTW matching method. This mapping network can be used as a front end of the TDNN-LR speech recognition system. This technique is being applied to other phoneme categories including all consonants and phonemes. Also, an unsupervised speaker-adaptation technique using neural networks is being investigated.

6.3 Speaker-independent Recognition

   In this section, we compare several TDNN architectures applied to speaker-dependent and multi-speaker's phoneme recognition with respect to their capabilities in a speaker-independent recognition problem.

   We verified performance of several architectures:
   (1) single TDNN, (2) SID (Stimulus Identification) network, (3) Meta-Pi network, (4) Modular TDNN and (5) Modular Speaker ID network, where the single TDNN is an original architecture, the SID network is constructed by both each speaker's module and a speaker ID module which selects outputs in each speaker's module, the Meta-Pi network is reported as the network most suitable for multi-speaker phoneme recognition. However, it has not been demonstrated how the Meta-Pi network is effective for a speaker-independent phoneme recognition problem. Furthermore, two novel modular TDNN architectures (4) & (5) are proposed to improve the performance. The modular TDNN is a network which is constructed by
integrating each speaker's module (i.e., a single TDNN) trained on the first stage, and retrained on the second stage to recognize each phoneme, regardless of training speakers. The Modular Speaker ID network comprises of a speaker ID module in addition to the Modular TDNN, thus explicitly classifying each speaker ID as in the Meta-Pi network.

**Speaker-independent** phoneme experiments for recognizing voiced stops /b, d, g/ using six and twelve training speakers showed high recognition rates of 92.1\% for the modular TDNN and 95.6\%, respectively for the Modular Speaker ID network. These results are significantly better than the rates of 82.0\% and 85.9\%, respectively for the Meta-Pi network. As a result, it is found that the Meta-Pi architecture suitable for multi-speaker recognition is not necessarily robust for a speaker-independent recognition task. The recognition rate for the Modular Speaker ID network nearly matches the speaker-dependent recognition rate of 98.0\% for the single TDNN model.

7. Conclusion

We described an integration of speech recognition and language processing. The speech recognition part consists of the Large Phonemic Time-Delay Neural Networks (TDNNs) which can automatically spot all 24 Japanese phonemes with an excellent spotting rate of 98.0\% by simply scanning among an input speech along with it. The language processing part is made up of a predictive LR parser which predicts subsequent phonemes based on the currently processed phonemes. The TDNN-LR hybrid recognition system provides large-vocabulary and continuous speech recognition. Two kinds of recognition experiments i.e., large-vocabulary isolated word recognition and continuous speech recognition were performed using the TDNN-LR method. Speaker-dependent recognition rates of 92.6\% for the first choices and 97.6\% for the top two choices were obtained for 5,240 Japanese common words, and rates of 65.1\% for the first choices and 88.8\% within the fifth choices were attained for phrase recognition. In the case of continuous speech recognition, adaptive incremental TDNN training using a small number of continuous training tokens is found to be very effective for adaptation of speaking manners.

We also proposed several new connectionist approaches as extensions of the TDNN-LR speech recognition system: (1) two proposed NN architectures (FTDNN and BWN) provided robust phoneme recognition performance on variations of speaking manner, (2) a speaker-adaptation technique can be realized using a NN mapping function between input and standard speakers and (3) the Modular Speaker ID architecture provided high phoneme recognition performance that nearly matches speaker-dependent recognition performance. These techniques should be implemented in the TDNN-LR method in the future.

Acknowledgement

The authors would like to express their gratitude to Dr. Akira Kurematsu, president of ATR, Interpreting Telephony Research Laboratories, for his support, and to Mr. Shigeki Sagayama, Head of Speech Processing Department for his valuable discussions on this research. They are also indebted to the members of the Speech Processing Department at ATR, for their constant help in various stages of this research.

References

Hidefumi Sawai was born in Kobe on Aug 23, 1954. He graduated from Dep. of Electrical Engineering in Faculty of Engineering of Keio University in 1977, and also graduated from Doctor course of Keio University in 1982. He joined Research Company in 1983 and has been engaged in research and development of speech recognition, pattern processing in the Research & Development Center. From 1988, he has been working for ATR Interpreting Telephony Research Laboratories where he is studying speech recognition using neural networks. He is currently a senior researcher and Dr. of Engineering. Dr. Sawai was a visiting researcher for Carnegie Mellon University, P. A., in the United States in 1989 and 1990. He is a member of the Editorial Committee of the Information Processing Society of Japan, currently a member of Acoustical Society of Japan, IEEE, Acoustics, Speech and Signal Processing Society, Neural Network Society and Japan Neural Network Society.

Yasuhiro Minami graduated from Dep. of Electrical Engineering in Faculty of Science and Technology of Keio University in 1986. Currently he is in a doctoral course of the same university and at the same time, he was an intern student of ATR Interpreting Telephony Research Laboratories. His interest mainly includes speech recognition. He is a member of Acoustical Society of Japan.

Masanori Miyatake graduated from Dep. of Electrical Engineering in Faculty of Engineering of Kobe University in 1980. He joined Sanrio Electric Company in the same year. From 1986 to 1989 he worked for ATR Interpreting Telephony Research Laboratories. Currently he is working for Information and Communication Systems Research Laboratories of Sanrio Electric Company. Since joining Sanrio Company, he has been engaged in speech recognition and synthesis. He is currently a senior researcher and a member of Acoustical Society of Japan.
Alex Waibel received the B.S., M.S. and Ph.D. degrees in Electrical Engineering and in Computer Science in 1979, 1980 and 1986, from MIT, and from Carnegie Mellon University, respectively. In 1986, he joined the faculty of the Computer Science Department as a Research Associate and is now a Research Computer Scientist. From May, 1987 to July, 1988, he has worked as Invited Research Scientist as the ATR Interpreting Telephony Research Laboratories in Japan. His current research interests include Speech Recognition and Synthesis, Neurocomputing, Machine Learning and Machine Translation. He is currently a member of the Technical Committee of the IEEE ASSP Society, a member of the IEEE ASSP Society, Computer Society, the Acoustical Society of America, the Association for Computational Linguistics and the International Neural Network Society.

Kiyohiro Shikano received the B.S., M.S. and Ph.D. degrees in electrical engineering from Nagoya University in 1970, 1972 and 1980, respectively. From 1972, he has been working at NTT Laboratories where he has been engaged in speech recognition research. He is currently the Head of Speech Information Processing Department at NTT Human Interface Laboratories, where he is managing the research of speech recognition and speech coding. During 1984-1986, he was a visiting scientist in Carnegie Mellon University, where he was working on distance measures, speaker adaptation by codebook mapping, and statistical language modeling. During 1986-1990, he was the Head of Speech Processing Department at ATR Interpreting Telephony Research Laboratories where he was directing speech recognition and speech synthesis research for interpreting telephone systems. He received the Yonezawa Prize from IEICE in 1975. He is a coauthor of “Speech, Auditory and Neural Network Models” (Series of Neuro Science & Technology, Ohm Press). He is a member of the Institute of Electrical and Electronics Engineers, Information Processing Society of Japan, and the Acoustical Society of Japan.