Self-Adjustable Neural Network for speech recognition

Hua-Nong Ting *, Boon-Fei Yong, Seyed Mostafa Mirhassani

Biomedical Engineering Department, Faculty of Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia

ARTICLE INFO

Article history:
Received 13 February 2012
Accepted 10 June 2013
Available online 13 July 2013

Keywords:
Neural Network
Variable input length
Malay vowels
Isolated words
TIMIT
Children speech

ABSTRACT

Neural networks with fixed input length are not able to train and test data with variable lengths in one network size. This issue is very crucial when the neural networks need to deal with signals of variable lengths, such as speech. Though various methods have been proposed in segmentation and feature extraction to deal with variable lengths of the data, the size of the input data to the neural networks still has to be fixed. A novel Self-Adjustable Neural Network (SANN) is presented in this paper, to enable the network to adjust itself according to different data input sizes. The proposed method is applied to the speech recognition of Malay vowels and TIMIT isolated words. SANN is benchmarked against the standard and state-of-the-art recogniser, Hidden Markov Model (HMM). The results showed that SANN was better than HMM in recognizing the Malay vowels. However, HMM outperformed SANN in recognising the TIMIT isolated words.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

The non-linear classification ability of the Artificial Neural Network (ANN) has made it advantageous and popular for use in speech recognition. ANNs such as Multi-Layer Perceptron (MLP), Time-Delay Neural Network (TDNN) and Recurrent Neural Network (RNN) require fixed length input vector in order for the speech signals to be trained and tested under the same ANN. In normal practice, once a proper input layer size of ANN is selected based on the maximum length of the speech signals, speech signals with shorter lengths are then padded with zero before they are fed into the ANN for training and testing. Due to the dynamic characteristics of speech, speech signals come with different types and lengths; thus, in practice, ANN is not sufficiently flexible to handle the timing variability of speech signals without adjustment to the speech length, analysis windows or analysis frames.

Popular speech feature extraction methods such as Mel-Frequency Cepstral Coefficient (MFCC) (Davis and Mermelstein, 1980), Perceptual Linear Predictive (PLP) (Hermansky, 1990), and Linear Predictive Cepstral Coefficient (LPCC) (Makhoul, 1975; Papamichalis, 1987) have been used in the short time spectral analysis to represent spectral information. In order to deal with speech signals of variable lengths, various methods have been proposed to provide fixed length input vectors. Yamauchi et al. (1995) proposed a speech recognition module that can control the speech signal length by introducing delay lines towards the input data. However, the approach was too static and it showed limited range of variance in real practical conditions where speech is more spontaneous. In another method, time scaling was proposed to expand and compress speech signals (Duchen-Sanchez et al., 2009). Under this time scaling, short speech signals are expanded to longer signals while long speech signals are shortened. However essential information of the speech may be lost during the compression of speech signals.

Rozman and Kodek (2007) proposed the use of asymmetric analysis window to extract the speech features according to the temporal and spectral functions. However, it was hard to determine the part of the speech signal that should be extracted with longer or shorter lengths. Variable-length frame overlap between two consecutive frames was proposed by Ding (2010) to handle the transition between different speech analysis frames. A more accurate feature vector of speech signals was extracted and compared to that using conventional fixed-length frame overlapping. Tyagi et al. (2006) also proposed variable size analysis windows to improve the recognition rate of the Hidden Markov Model (HMM)-based ASR.

On the other hand, HMM, which is currently the state-of-the-art in ASR, does not have problems with variable speech lengths. HMM is able to handle both temporal and spectral variations statistically (OShaughnessy, 2008). However, HMM still has some major limitations for applicability in ASR. Thus, hybrid systems of ANN and HMM were proposed to improve flexibility and recognition performance (Trentin and Gori, 2001). The tandem connectionist method was further proposed to improve the recognition rate of the ANN/HMM system (Hermansky et al., 2000). The conventional HMM systems used likelihood assumption to predict how the phonations were produced while the Tandem method utilised the log posterior
probabilities from several acoustics models to perform speech recognition on a large scale.

TDNN was proposed by Waibel et al. (1989) to introduce delays to the input units. TDNN was able to learn the temporal structure of acoustic events and the temporal relationships between such events. Besides that, TDNN was translation invariant, which allows the acoustic features learned by the network to be insensitive to shifts in time. TDNN outperformed HMM in the recognition of the three Japanese phonemes. Since then, improvements have been studied to increase the robustness of TDNN (Hemphire and Waibel, 1990; Day and Davenport, 1993; Baldi and Atiya, 1994; Poo, 1997; Yazdizadeh and Khorasani, 2002; El-Bakry, 2006; Becerikli and Oysal, 2007; Ge et al., 2009). Although delay is introduced to the time series speech signals, TDNN still suffers from the fixed length problem and it is inflexible in handling speech signals with variable lengths.

In order to deal with the variable lengths of speech signals, a Self-Adjustable Neural Network (SANN) is proposed in this paper. The proposed SANN can automatically adjust its structure according to the input length and it enables different lengths of speech signals to be trained and tested together under the same network. This paper is organised as follows: Section 2 introduces the architecture of the SANN structures. Section 3 explains the speech feature extraction using Mel-frequency Cepstral Coefficients (MFCCs). Section 4 explains the database, as well as the training and testing of SANN and the state-of-the-art speech recogniser, HMM. Section 5 presents the results and discussion on the recognition rate of the SANN and HMM. The paper is then concluded in Section 6.

2. Self-Adjustable Neural Network

2.1. SANN architecture

The Self-Adjustable Neural Network (SANN) architecture is presented in this section. The explanation of the SANN architecture is based on the speech recognition of 6 different Malay vowels as an example. SANN is proposed based on the architecture of TDNN. TDNN is further modified so that SANN can allow speech signals with different lengths to be trained and tested under the same network.

The architecture of SANN is illustrated in Fig. 1. SANN consists of 4 layers, which are the input layer, first hidden layer, second hidden layer and output layer. The analysis window, which consists of a few analysis frames, is used in SANN so that the speech signals are fed into the SANN one segment at a time in order to be connected to one analysis frame at the next layer. The analysis window is then shifted or delayed by one analysis frame in order to be connected to the next analysis frame at the next layer. The analysis windows are shifted in a way that they overlap with each other with two analysis frames.

The maximum total input layer size is comprised of 15 analysis frames, with each analysis frame consisting of 39 MFCC coefficients. However, the input layer size of SANN at one time is comprised of an analysis window, with a size of 3 analysis frames. The number of input nodes at one time is equal to 117 MFCC coefficients. The first analysis window at the input layer is connected to the first analysis frame at the first hidden layer. The analysis window is shifted or delayed by one analysis frame in order to be connected to the second analysis frame at the first hidden layer. The shifting of the analysis window at the input layer is repeated until reaching the last analysis frame at the input layer. A total of 13 analysis frames are formed at the first hidden layer based on 15 analysis frames at the input layer. Every analysis window has its own copy of weights when it is connected to the analysis frame at the next layer. Every neuron in one analysis window at the input layer is fully connected to all neurons in one analysis frame at the first hidden layer while every neuron in one analysis frame at the first hidden layer is fully connected to all neurons in one analysis window at the input layer.

The first hidden layer consists of a maximum layer size of 13 analysis frames. The first hidden layer size at one time is again comprised of an analysis window, with a size of 5 analysis frames.

![Fig. 1. Architecture of SANN.](image-url)
The first analysis window at the first hidden layer is connected to the first analysis frame at the second hidden layer. The analysis window is shifted by one analysis frame in order to be connected to the second analysis frame at the second hidden layer. The shifting of the analysis window at the first hidden layer is repeated until reaching its last analysis frame. Thus, a total of 9 analysis frames are formed at the second hidden layer based on 13 analysis frames at the first hidden layer. Every analysis window has its own copy of weights when it is connected to the next layer. Every neuron in one analysis window at the first hidden layer is fully connected to all neurons in one analysis frame at the second hidden layer while every neuron in one analysis frame at the second hidden layer is fully connected to all neurons in one analysis window at the first hidden layer.

The second hidden layer consists of a maximum size of 9 analysis frames, with each analysis frame consisting of 6 hidden neurons. The hidden neurons across all the analysis frames at the second hidden layer are connected to the corresponding output neurons at the output layer. Thus, the hidden neuron number at the second hidden layer is set in a way that equals the output neuron number at the output layer. The first neuron at the second hidden layer across all the activated analysis frames is connected to the first output neuron at the output layer. The same approach is applied to the second until the last output neuron.

The number of analysis frames at the layers of input, first hidden layer and second hidden layers are determined respectively according to Eqs. below:

\[ N_1 = \frac{S}{L}, \quad (2.1) \]

\[ N_2 = N_1 - \text{AWS1} + 1, \quad (2.2) \]

\[ N_3 = N_2 - \text{AWS2} + 1, \quad (2.3) \]

where \( N_1 \) is the number of analysis frames at the input layer, \( N_2 \) is the number of analysis frames at the first hidden layer, \( N_3 \) is the number of analysis frames at the second hidden layer, \( S \) is the speech length in millisecond, \( L \) is the analysis frame length in millisecond, \( \text{AWS1} \) is the analysis window size at the input layer (in terms of number of analysis frames) and \( \text{AWS2} \) is the analysis window size at the first hidden layer (in terms of number of analysis frames). The \( \text{AWS1} \) and \( \text{AWS2} \) are set at 3 and 5 respectively. For speech sounds with any signal length that are fed into SANN, the size of the input, first hidden and second hidden layers will be determined according to Eqs. (2.1)-(2.3). The input layer of SANN consists of a minimum of 7 analysis frames and a maximum of 15 analysis frames. A minimum 7 analysis frames at the input layer are required in order to be connected to 5 analysis frames at the first hidden layer and subsequently one analysis frame at the second hidden layer. This is the most basic structure of SANN. In other words, if MFCC coefficients are extracted using analysis frame of 25 ms with a shift of 10 ms, then a minimum speech length of 85 ms is required for SANN. Thus, the bigger network size will be activated for longer speech lengths compared to that of shorter speech lengths.

### 2.2. The learning algorithm

In TDNN, different shifted or delayed analysis windows shared the same copy of the weights. The weights at the same layer were updated using the same average values (Waibel et al., 1989). In SANN, every analysis window and its shifted analysis window has its own copy of weights when it is connected to the next layer. The weights are updated for every analysis window based on the current weight values. SANN is trained using standard error back-propagation and the weights are updated using the standard gradient descent method.

The multiplication of weights and input nodes that are connected to each other is calculated, then the result is added with the biases and, lastly, multiplied by a non-linear sigmoid function to produce a neuron at the next layer. The same feed forward method is used for connections at all the layers. Results obtained at the output layer are compared with the desired output. For every training pattern, error information terms are then calculated and used to update the weights and biases in every layer as shown in Eqs. (2.4) and (2.5).

\[ w(t + 1) = w(t) + \Delta w(t + 1) \quad (2.4) \]

\[ \Delta w(t + 1) = a\Delta k + \beta \Delta w(t) \quad (2.5) \]

where \( w(t+1) \) is the new updated weight, \( w(t) \) is the current weight, \( \Delta w(t+1) \) is the updated weight correction term, \( \Delta w(t) \) is the current weight correction term, \( a \) is the learning rate, \( \beta \) is the momentum, \( \Delta \) is the error information term at the output layer, second hidden layer and first hidden layer; and \( k \) are the neurons at the second hidden layer, first hidden layer and input layer.

The weights of the SANN are initialised and connected according to the maximum length of the speech length, i.e. 165 ms or 15 analysis frames with each analysis frame length at 25 ms and a shift of 10 ms. Therefore, when the speech length is less than the maximum length, the weights will be activated to the speech length while other weights will not be activated. The weights that are not activated will remain the same in the forward propagation and only those activated weights will be multiplied with the input values and feed-forwarded by the non-linear function to obtain the output. When the weight values need to be updated at the stage of error back propagation, the weights that are not activated will not be updated and will remain the same. This allows SANN to learn the information about variable lengths of the speech signals.

### 3. Speech feature extraction

A pre-emphasis is used to spectrally flatten the speech tokens according to Eq. (3.1).

\[ x(n) = s(n) - 0.95s(n-1) \quad (3.1) \]

where \( s(n) \) is the input speech token and \( x(n) \) is the pre-emphasised speech token. Then, the speech tokens are blocked into analysis frames. A Hamming window is used to window the analysis frame in order to minimise the discontinuities at its boundary. The windowed speech token is

\[ h(n) = x(n)w(n), \quad 0 \leq n \leq N-1 \quad (3.2) \]

where \( w(n) \) is the Hamming window function (Webster, 1978), and \( N \) is the total sample points in an analysis frame. A typical Hamming window is defined in Eq. (3.3).

\[ w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \quad 0 \leq n \leq N-1 \quad (3.3) \]

MFCC is used to extract the speech features based on this windowed speech token. MFCC consists of 39 orders, which includes 13 mel-frequency cepstral coefficients (12 MFCC plus 1 log energy term using Discrete Cosine Transform), 13 first derivative and 13 second derivative features (Delta cepstrum). Fast Fourier Transform (FFT) is performed on the \( h(n) \) and the output spectrum \( f \), which is then changed to log mel-spectrum according to Eq. (3.4). Discrete Cosine Transform is then used to convert the log mel-spectrum to MFCC.

\[ \text{Mel}(f) = 2595 \log_{10} \left(1 + \left(\frac{f}{700}\right)^{0.5}\right) \quad (3.4) \]

MFCC coefficients were directly extracted by using the Hidden Markov Toolkit (HTK) (Young et al., 2006). MFCC were normalized between −1 and +1 before being fed into SANN. 39 MFCC
coefficients were extracted at every analysis frame, with each analysis frame length of 25 ms and a shift of 10 ms.

4. Methodology

4.1. Speech database

4.1.1. Malay vowels

Malay vowels consist of /a/, /e/, /i/, /o/ and /u/. A Malay vowel speech database was compiled from 360 Malay children (180 males and 180 females) aged between 7 and 12 years. The vowels were sustained for a duration of between 3 and 5 s. The speech database consisted of 2160 (360 speakers × 6 vowels) speech tokens. The speech samples were recorded using Goldwave audio editor, with a sampling rate of 20 kHz with 16 bit of resolution.

Three-fold cross validation was used for the Malay vowel speech database. Speech tokens of 240 and 120 speakers were used for the training and testing respectively each time. The experiments were conducted in three different training and testing sets. The speech tokens were extracted using MFCC. 39 MFCC coefficients were extracted at every analysis frame length of 25 ms with a shift of 10 ms. Variable lengths of between 85 ms and 165 ms per speech token were used for the training and testing sets. The speech tokens were extracted using MFCC. 39 MFCC coefficients were extracted at every analysis frame length of 25 ms and a shift of 10 ms.

4.1.2. TIMIT isolated words

A fully segmented and labeled TIMIT acoustic-phonetic continuous speech corpus (Garofolo et al., 1992) was used to evaluate and compare the performance of SANN with HMM. The TIMIT database was used because it is the standard database used in many studies and it contains speech sounds with variable lengths (Pols et al., 1996).

The isolated words were extracted from two sa sentences of TIMIT: “She had your dark suit in greasy wash water all year” and “Don’t ask me to carry an oily rag like that”. Ten isolated words were selected, which were “dark”, “wash”, “water”, “year”, “don’t”, “carry”, “greasy”, “like”, “oily” and “that”. The training and testing set consisted of 462 and 168 speakers, respectively. MFCC were extracted from the speech tokens with a minimum signal length of 85 ms in order to produce a minimum seven analysis frames. One testing speech token in “don’t” did not meet the minimum signal length, thus it was excluded. The details of the database are shown in Table 1. MFCC was extracted using an analysis frame length of 25 ms with a shift of 10 ms. A maximum frame number of 63 was extracted per speech token.

4.2. Experimental setting

MFCC was used to extract the speech features. 39 MFCC coefficients were derived per analysis frame. The total number of MFCC coefficients per speech token depends on the speech length. For Malay vowels, variable lengths of the speech token were used between 85 ms and 165 ms, with an increment of 10 ms. Analysis frame length of 25 ms with a shift of 10 ms (or an overlapping of 15 ms) was used for the speech feature extraction. Thus, a minimum of 7 analysis frames and a maximum of 15 analysis frames were extracted from these tokens. MFCC coefficients were extracted from the Malay vowels about 0.75 s from their start point while for TIMIT isolated words, the speech features were extracted based on the labelled start point.

For the training of SANN, all the weights were initialised randomly between −0.3 and +0.3. The training of the neural networks was stopped when they reached a maximum iteration of 100 for Malay vowels and 1000 for TIMIT isolated words, and a minimum root mean square error of 0.005. The root mean square error, \( E_{\text{rms}} \), is defined as

\[
E_{\text{rms}} = \sqrt{\frac{1}{PK} \sum_{p=1}^{P} \sum_{k=1}^{K} (t_{pk} - y_{pk})^2}
\]  

(4.1)

where \( P \) is the number of training patterns and \( K \) is the number of output neurons, \( t_{pk} \) is the target or desired output and \( y_{pk} \) is the output of the network.

The isolated words were extracted from two sa sentences of TIMIT: “She had your dark suit in greasy wash water all year” and “Don’t ask me to carry an oily rag like that”. Ten isolated words were selected, which were “dark”, “wash”, “water”, “year”, “don’t”, “carry”, “greasy”, “like”, “oily” and “that”. The training and testing set consisted of 462 and 168 speakers, respectively. MFCC were extracted from the speech tokens with a minimum signal length of 85 ms in order to produce a minimum seven analysis frames. One testing speech token in “don’t” did not meet the minimum signal length, thus it was excluded. The details of the database are shown in Table 1. MFCC was extracted using an analysis frame length of 25 ms with a shift of 10 ms. A maximum frame number of 63 was extracted per speech token.

Table 1

<table>
<thead>
<tr>
<th>Type of recogniser</th>
<th>Number</th>
<th>Malay vowels</th>
<th>TIMIT isolated words</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Training</td>
<td>Testing</td>
</tr>
<tr>
<td>SANN and HMM</td>
<td></td>
<td>240</td>
<td>120</td>
</tr>
<tr>
<td>Speaker</td>
<td>12,960</td>
<td>6480</td>
<td>4620</td>
</tr>
</tbody>
</table>

Table 2

Performance of various speech recognisers for isolated word recognition.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SANN</td>
<td>92.02</td>
</tr>
<tr>
<td>HMM</td>
<td>99.70</td>
</tr>
</tbody>
</table>

Table 3

Confusion matrix for Malay vowel recognition using SANN.

<table>
<thead>
<tr>
<th>Vowel</th>
<th>/a/</th>
<th>/e/</th>
<th>/i/</th>
<th>/o/</th>
<th>/u/</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>/a/</td>
<td>3170</td>
<td>0</td>
<td>38</td>
<td>9</td>
<td>23</td>
<td>0</td>
</tr>
<tr>
<td>/e/</td>
<td>9</td>
<td>3090</td>
<td>48</td>
<td>93</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/i/</td>
<td>79</td>
<td>69</td>
<td>3020</td>
<td>23</td>
<td>42</td>
<td>7</td>
</tr>
<tr>
<td>/o/</td>
<td>0</td>
<td>94</td>
<td>12</td>
<td>3134</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/u/</td>
<td>60</td>
<td>6</td>
<td>18</td>
<td>0</td>
<td>294</td>
<td>215</td>
</tr>
</tbody>
</table>

Table 4

Performance of various speech recognisers for isolated word recognition.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SANN</td>
<td>92.02</td>
</tr>
<tr>
<td>HMM</td>
<td>99.70</td>
</tr>
</tbody>
</table>
actual output. The momentum and learning rate were set at 0.9 and 0.1 respectively. For every vowel pattern, the target value is set in a way that only one output neuron is set to 0.9 while the rest of the output neurons are set to 0.1.

In SANN, the minimum and maximum number of analysis frames at the input layer was 7 and 15, respectively for the vowel speech tokens. Thus the minimum and maximum number of input neurons was 273 and 585, respectively. In recognition of Malay vowels, SANN was trained with first hidden neuron number between 10 and 50, in a step of 2. As for TIMIT isolated words, SANN was trained with first hidden neuron number between 10 and 20, in a step of 2. The second hidden neurons were set at 6 for the vowel recognition while 10 for the TIMIT isolated word recognition. The analysis window size was set at 3 and 5 for the input layer and first hidden layer, respectively. The output neurons of SANN were set at 6 for the recognition of Malay vowels and at 10 for the recognition of TIMIT isolated words.

The HMM was implemented using the HMM Toolkit (HTK), which was developed by Cambridge University, United Kingdom (Young et al., 2006). The whole word model was used for the vowel and TIMIT isolated word recognition. The HMM model consisted of 3 states of left to right HMM and each state consisted of 8 Gaussian mixtures. HInit and HRest tools were used for the training while for the testing, HCopy, HVite and HResults tools were used.

5. Results and discussion

The performance of SANN in Malay vowel recognition different first hidden neuron number is summarised in Table 2. SANN obtained the highest average recognition rate of 95.15% at the first hidden neuron number of 22. SANN needed to be trained with the proper first hidden neuron number. Lower or higher number of first hidden neuron number could not guarantee a optimal performance of SANN. On the other hand, HMM achieved an average recognition rate of 90.09% with the accuracy of set 1, set 2 and set 3 were 88.72%, 90.34% and 91.20% respectively. Obviously, SANN outperformed HMM in recognizing Malay vowels. The confusion matrix for Malay vowel recognition using SANN is shown in Table 3. Vowel /a/ was recognised with the highest accuracy, followed by /u/ and /i/, while /o/ was recognised with the lowest accuracy.

The performance of SANN and HMM for the TIMIT isolated word recognition is shown in Table 4. The results showed that HMM outperformed SANN in recognizing the isolated words. The confusion matrix for TIMIT isolated word recognition using SANN is shown in Table 5. Word “carry” was recognised with the highest accuracy while “wash” was recognized with the lowest accuracy.

The proposed SANN is able to deal with variable lengths of speech. SANN adjusts its network size according to the length of the speech. Though a maximum input layer size of 15 (equivalent to a speech length of 165 ms) was used for the vowel speech tokens, there is no limit on the input layer size of SANN. SANN can be extended to accommodate speech with longer length. In the recognition of TIMIT isolated words, the minimum input layer size was kept at 7, while the input layer size of SANN was extended to the maximum length of the speech tokens at 63. Larger network size will be activated for speech tokens with longer length while smaller network size will be activated for speech tokens with shorter length. This forces SANN to learn different lengths of the speech tokens. Compared to the conventional ANN such as MLP and TDNN, the input layer of the network needs to be initialized with the maximum signal length. Speech tokens with shorter signal length are padded with zero before they are fed into the ANN for training and testing.

6. Conclusion

A novel Self-Adjustable Neural Network is proposed in this paper with application in speech recognition. SANN is able to adjust its network size according to the speech length and allows speech tokens with variable lengths to be trained and tested under the same network. SANN has been successfully used to recognise Malay vowels and 10 TIMIT isolated words. SANN outperformed the state-of-the-art HMM in recognising Malay vowels but its accuracy was lower in recognizing the TIMIT isolated words.

Acknowledgements

The authors would like to thank the Ministry of Science, Technology and Innovation, Malaysia (MOSTI) and University of Malaya for funding this study under Science Fund (Project no. 06-01-03-SF0516) and University Malaya Research Grant (UMRG) (Project no. RG126/11AET).

References


