Script to Speech Conversion for Hindi Language by Using Artificial Neural Network

Author – Mrs. Prajakta S. Rathod, Institute of Technology, Nirma University

Abstract--**This paper represents the technique for converting the text written in Hindi language into speech by using artificial neural network. Text to speech conversion software can have many applications in day to day life. It can be useful for blind people to read the document. It is useful in giving education to the students by listening what is written in books. If someone doesn't have time to read emails, he can listen the contents of email while doing other work. The document containing Hindi text is scanned and given as an input to the system, it is considered as an image. Then pre-processing is done for the document to obtained clear image. In this system neural network is used for character recognition, so the system can be enhanced to work with letters written in different style or fonts. After the letters in the document are successfully identified by neural network, the text in the document is converted into speech.**

I. INTRODUCTION

The script to speech conversion system takes scanned document as an input which is in image form. This document contains text which is to be converted into speech. There are three main modules of the system as shown in Fig. 1.

- Pre-processing the document to obtain clear image and identify each character.
- Character is given to the neural network for identification.
- Converting identified character into sound.

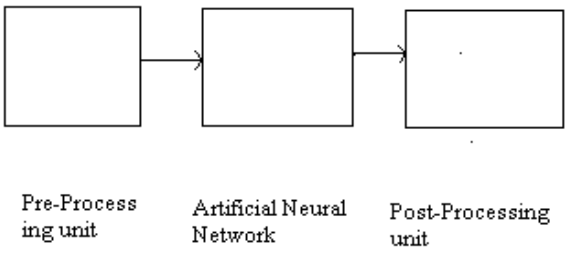

Fig. 1 Sub-modules of the System

II. PRE-PROCESSING

In preprocessing part, all the pixel values are converted into 0 and 1. Pixel having gray value above threshold is converted into 1 and pixel having gray value below threshold is converted into 0. Then thinning is done to remove noise and obtained clear image of the document.

After getting clear image, first lines in the document are separated. After that words are separated from lines and then characters are separated from the words. These characters may have heads and tails with them, so again heads and tails are separated from these characters. Now we have separated characters with their heads and tails.

For separating lines in the document, horizontal histogram of the document is taken. The blank space between the two lines is considered for making the separation. The space is identified in horizontal histogram. Fig. 2 shows the horizontal histogram of the document for separating the lines.

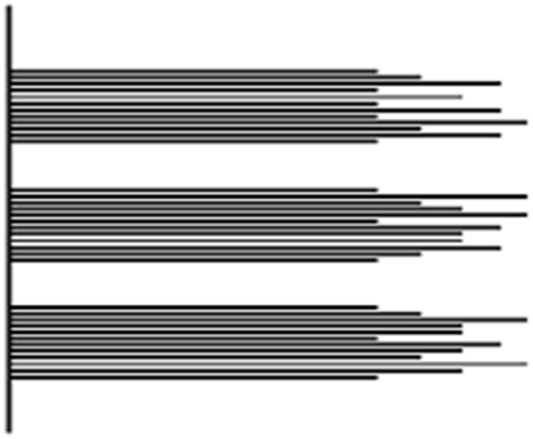

Fig. 2 Horizontal Histogram of Lines in the Document

After lines get separated, vertical histogram of the line is taken for separating the words. For separating the words from the lines, the blank space between two words is used. Fig. 3 shows the vertical histogram for separating words from line.

Once the words are separated, the head portion and tail portion associated with characters is separated. For separating head portion, again horizontal histogram of the word is taken. Horizontal line above the word is taken as a separation line of head and character. The portion above the horizontal line is head portion and portion below horizontal line is the character body. Then tail portion is separated from the character body. For separating tails, character width is taken into consideration. Also the point at which tail is associated with the character has very few black pixels. Using this point and the horizontal histogram of word, the tail portion is separated from the character body.

Thus we have separate head portion, character body and tail portion.

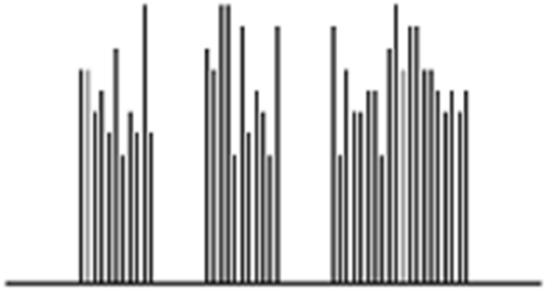

Fig. 3 Vertical Histogram of Words in the Line

Now for separating individual characters from character body portion of the word, the gap between the characters is considered. The horizontal line above the word is taken as a minimum threshold value for character separation, and by using vertical histogram, individual characters are separated from word. Fig. 4 shows the vertical histogram of the word for separating characters.

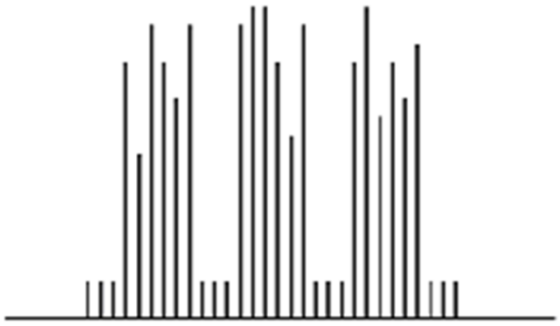

Fig. 4 Vertical Histogram of Characters in the Word

The separated characters, heads and tails are identified by using neural network. All the characters in Hindi are divided into four groups depending on vertical line present in the character. First group is the group of characters having vertical line in the middle. Second group is the group of characters having vertical line at the end of character. Third group contains characters having two vertical lines. Fourth group contain characters having no vertical line. For identifying the group to which character belongs, vertical histogram of the character is taken. Vertical histogram indicates whether character has single vertical line in middle or single vertical line at the end or two vertical lines or no vertical line. Again two groups are made. One is for head and another is for tail. Now we have six groups in all. Six different neural networks are used for these six groups.

Hindi characters are classified in different groups as follows [6], [2].

1. Group of characters having vertical line in the middle

क फ

Fig. 5 Groups of Characters Having Vertical Line in the Middle

2. Group of characters having vertical line at the end.

अ ख घ च ज झ त थ ध न ष य

ल व श स क्ष ज्ञ

Fig. 6 Group of Characters Having One Vertical Line at the End

3. Group of characters having two vertical lines

ए ग प भ म ष ण

Fig. 7 Group of Characters Having Two Vertical Lines

4. Group of characters having no vertical line

इ उ छ ट ठ ह ढ र ह

Fig. 8 Group of Characters Having No Vertical Line

5. Group of heads

ि ी etc.

Fig. 9 Group of Heads

6. Group of tails

ॅ ॅ etc.

Fig. 10 Group of Tails

When character is separated, it is first decided to which group it belongs. i.e. whether it is a character having one vertical line in the middle, or having one vertical line at the end. Whether it has two vertical lines or no vertical lines. After identification of the group of character, the character is send to the appropriate neural network for identifying exactly which character it is. All the characters in one group are given to one neural network. Neural network is trained to identify all the characters in one group.

A 10X10 matrix is taken around character. Each pixel will have a value either 1 or 0. In fig. 11, the Hindi alphabet has been digitized into 10X10=100 digital cells, each having a single color, either black or white. It becomes important for us to encode this information in a form meaningful to a computer. For this, we assign a value 1 to each black pixel and 0 to each white pixel and create the binary image matrix which is shown in the Fig. 12. Digitization of an image into a binary matrix of specified dimensions makes the input image invariant of its actual dimensions. Hence an image of whatever size gets transformed into a binary matrix of fixed pre-determined dimensions. This establishes uniformity in the dimensions

of the input and stored patterns as they move through the recognition system, as in [7], [2].

Fig. 11 10×10 Matrix is Taken for Hindi Character

0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1	1	0
0	0	0	1	1	0	0	1	1	0	0
0	0	0	1	1	0	0	1	1	0	0
0	0	1	1	1	1	1	1	1	0	0
0	0	1	1	1	0	0	1	1	0	0
0	0	0	1	0	0	0	1	1	0	0
0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0

Fig. 12 Pixel Values in fig. 11 Converted into 0 and 1. 0 for White Pixel and 1 for Black Pixel.

When the character identification is over, then heads and tails associated with character are given to the respective neural network for identification. Now we have done character identification for all the characters in the document. Heads and tails associated with the characters are also identified.

III ARTIFICIAL NEURAL NETWORK

An artificial neural network (ANN), usually called neural network (NN), is a mathematical model or computational model that is inspired by the structure and/or functional aspects of biological neural networks. A neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

An artificial neuron is a device with many inputs and one output. The neuron has two modes of operation, the training mode and the using mode. In the training mode, the neuron can be trained to fire for particular input patterns.

In the using mode, when a taught input pattern is detected at the input, its associated output becomes the current output. If the input pattern does not belong in the taught list of input patterns, the firing rule is used to determine whether to fire or not. Fig. 13 shows the working of neuron, as in [3].

The commonest type of artificial neural network consists of three groups, or layers, of units.

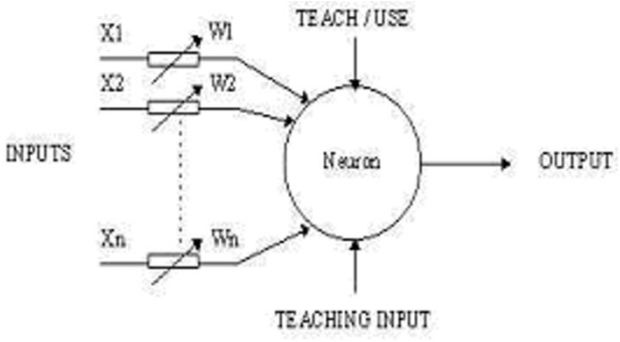

Fig. 13 An MCP (McCulloch and Pitts model) Neuron

$$Y_j = X_1W_1 + X_2X_2 + \cdots\cdots\cdots X_nW_n \qquad (1)$$

A layer of input units is connected to a layer of hidden units, which is connected to a layer of output units, as shown in fig 14.

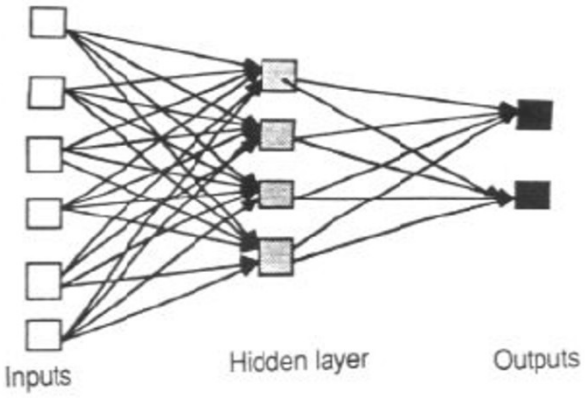

Fig. 14 Layers of Artificial Neural Network

- The activity of the input units represents the raw information that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
- The behaviour of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

A unit in the output layer determines its activity by following a two step procedure.

- First, it computes the total weighted input Xj, using the formula:

$$X_j = \sum_i Y_i W_{ij} \qquad (2)$$

where Y_i is the activity level of the i^{th} unit in the previous layer and W_{ij} is the weight of the connection between the i^{th} and the j^{th} unit.

- next, the unit calculates the activity Y_j using some function of the total weighted input. Typically we use the sigmoid function:

$$Y_j = \frac{1}{1+e^{-X_j}} \qquad (3)$$

Once the activities of all output units have been determined, the network computes the error E, which is defined by the expression:

$$E = \frac{1}{2}\sum_j (Y_j - d_j)^2 \qquad (4)$$

where y_j is the activity level of the j^{th} unit in the top layer and d_j is the desired output of the j^{th} unit, as in [1], [3], [4].

Backpropagation Algorithm

Training an artificial neural network is nothing but providing it with a training set and allowing it to learn by adjusting weights of its synapses. A training set is a collection of training samples.

Training Set = Set of training samples

A training sample is a pair of a sample input vector and a desired output vector. The length of input vector should be same as the number of neurons in the input layer, and the output vector length should be equal to the number of neurons in output layer.

Training Sample = (input vector, desired vector)

Backpropagation algorithm is a commonly used supervised algorithm to train feed-forward networks. It was first introduced by Paul Werbos in his book 'The Roots of Backpropagation'. The basic idea is to determine how the neural network behaves for a sample input, compare how different it is from the desired behavior and then adjusting the weights of synapses to minimize the difference. This process is repeated for all training samples in the set multiple times to ensure proper training.

Backpropagation training algorithm:

- Initialize the weights of networks
- Choose a random training sample, and assign input vector to the input neurons
- Propagate all neurons in the forward direction to obtain output at the output layer
- Evaluate error values at each neuron in the output layer as the difference between obtained output and the desired output of the training sample chosen
- Evaluate Mean Squared Error value. This value reflects the effectiveness of training done so far.
- Backpropagate the errors, all the way upto the input layer
- Calculate delta (weight update) for all synapses
- Update the weights of all synapses such that the sum-squared value of error is minimized.
- Now, choose another random sample and repeat the process. In this fashion, train all samples in some random order. A training epoch is a cycle through all the samples in the training set. Typically, many training epochs are required to train a backpropagation network.
- Check if the stopping criterion has reached. If not, continue with the next training epoch.

The stopping criterion is usually a limit to acceptable mean squared error or a limit to the number of training cycles to use. Here maximum number of training cycles used are 3000, as in [5].

IV. POST PROCESSING

A three digit code is generated for each of the character. That code represents the character and its associated head or tail.

Consider following Hindi word

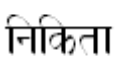

Fig. 15 Example Word

Three digit code generated for each letter in this word is

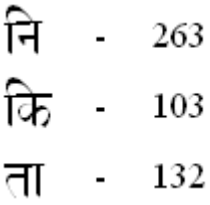

Fig. 16 Three Digit Code Given to Each Letter

The database containing sound files for pronunciation of each letter is prepared. Database contains entries for sound file and its corresponding three digit code. After the three digit code is generated by the system, its entry is searched in the database. If its entry is found then, the corresponding sound file name is return to the system. Then the system plays the sound file for the pronunciation of the letter. This is done for all the letters present in the word,

and finally text in entire scanned document is converted into speech.

V. RESULT

A text to speech conversion software for Hindi Language is implemented using Artificial Neural Network. Currently this system is working for Hindi font shivaji01. The database of words contains four hundred and fifty different words which are sufficient to process any common document. The success rate of the system is 75%.

VI. FUTURE ENHANCEMENT

- At present this system is working only for scanned document i.e. it takes image as an input. In future it can be enhanced to work with typed text.
- Now it is specific to Hindi font shivaji01, in future it can be implemented for any Hindi font.
- In future it can be improve to work with different type of handwritten text also.
- Currently it is working only for Hindi language, in future it can be enhanced to work with other languages also.
- At present the system does not identify connected letters. In future the database of words can be improved by adding connected letters.

VII. CONCLUSION

A system for converting text written in Hindi language into speech is described. This system involves preprocessing of image to obtained clear text. After that individual characters are separated from the text. Then character recognition is done using artificial neural network. Finally post processing is done for identified character to convert it into speech. This system has many applications in day to day life. Also new features can be added to the system in future to make it more powerful.

VIII. REFERENCES

[1] Artificial Neural Network by Robert J. Schalkoff. McGRAW-HILL INTERNATIONAL EDITIONS, Computer Science Series.

[2] Artificial Neural Network by B. Yenanarayana, Published by Asoke K. Ghosh, PHI learning Pvt. Ltd.

[3] http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html

[4] http://www.learnartificialneuralnetworks.com/

[5] http://neurondotnet.freehostia.com/manual/bpnalgo.html

[6] http://www.hindilearner.com/hindi_alphabet.html

[7] http://www.codeproject.com/KB/dotnet/simple_ocr.aspx