Designing and Implementation of Neural Network Architecture Using Phonetic Identification Algorithms for Obtaining Performance Improvement of Recognizing English Homophone Words

Vimal P.Parmar [*], Dr. CK Kumbharana [**]

[*] Research Scholar, Department of Computer Science, Saurashtra University, Rajkot, Gujarat, INDIA
[**] Head, Guide, Department of Computer Science, Saurashtra University Rajkot, Gujarat, INDIA

Abstract- Neural network architecture is widely used in computer science for solving complex problems arise in various research applications. One of the popular applications of neural network is pattern recognition. Neural network can be used to solve real world complex problems that can be solved by human being among which some require less effort while others require tremendous efforts. If neural network is incorporated with phonetic algorithms that recognize whether the given two words are phonetically similar or not yields improved outcome. The integrated approach discussed here results in performance improvement over any single algorithm implementation. Different phonetic algorithms are developed to identify phonetic similarity with different set of rules and different set of output after processing English words. Homophones are the English words having similar pronunciation but different spelling and meaning. The number of such phonetic algorithms are studied and discussed briefly here and then another algorithm with phonetic rules is built. Each algorithm has different level of performance. The performance word used here is in the context of identifying the homophones but not the processing performance of algorithm performed by a computer system. Each algorithm has advantages and limitations but by incorporating these algorithms in neural network results in improvement compared to any single algorithm implementation. Neural networks are commonly used for solving pattern recognition problems and so the efforts are being made here to recognize similar English words pattern found in pronunciation of the English words.

Index Terms- Neural Network, Phonetic algorithms, Matching, Soundex, Metaphone, Comparison, Pattern matching, Pronunciation, Match Rating, Pattern recogniztion

I. Introduction

Identification of whether given English words are homophones meaning that whether they have similar pronunciation or not requires text processing through a computer. Many numbers of algorithms are developed for such purposes which are implemented in many applications development as well as database tools.

Each such algorithm has its own advantages and limitations. Here an effort is to build a neural network consisting of more than one algorithm to identify homophone. The integrated approach applied here proved better performance compared to an individual implementation. Neural network consists of a network of neurons in which each neuron performs some unique functionality and passes the result to its nearer another neuron. Each neuron receives result consisting of data as well as weight associated with it from more than one neuron and processes them. Thus it forms a complex network of neurons and forwarding the processed data to the next layer in the network. Each neuron is considered as to perform an individual processing. Network binds all neurons in a single unit. Here the algorithms that we have used for phonetic identification represented as a neuron. Weights of the neuron can be initially decided and updated using experiments and performance criteria. Once the acceptable result is achieved neural network can be used for other set of data. The prime goal of the research paper is to achieve a better performance of phonetic algorithms which if implemented individually fails for certain kind of data. By using cumulative weight from different neuron's output it can be determined whether the given words have similar pronunciation or not. Here two levels processing of neurons are applied. At first level two existing algorithms and one derived algorithm is applied. The result is feed forward to next level of neuron representing match rating algorithm. Further if the accepted result is obtained from first level than it doesn't require processing of second level. At last set of data are experimented as analyzed by comparing individual algorithm performance and applied as whole neural network performance. The neural network applied here can be considered as AND feed forward neural network.

II. Neural Network Introduction

Artificial intelligence, a branch of computer science which is one of the popular research area in solving complex problems which can be solved by human being. Human can think and solve logical, reasonable and innovative ideas due to the powerful mind which is composed of plenty of neurons. Complex structure of neural network can find out solution of almost any kind of complex problem efficiently with compared to any powerful computer. Computer is powerful device for manipulating numerical and textual data as well as storing and retrieving operations but it is more difficult for a computer to identify pattern from thousands of other patterns. But human mind can recognize the face seen many years ago within a fraction of a time. To solve such a problem using a computer requires massive data processing. The power of human brain lies in complex natural neural network which is a parallel processor and can solve such complex problems very quickly.

Following figure[7][9] describes the natural neural network of human brain. Each neuron has a nucleus with a cell body or soma and connected with other neurons through axon. Dendrite is the fiber which is an input to the neuron. Biochemical signal moves through axon from one neuron to another with weighted sum of all the input and must have value greater than or equal to some value called threshold value. Threshold value is determined by synapse. Here is decided whether the signal is to be exhibited or inhibited to the next neuron.

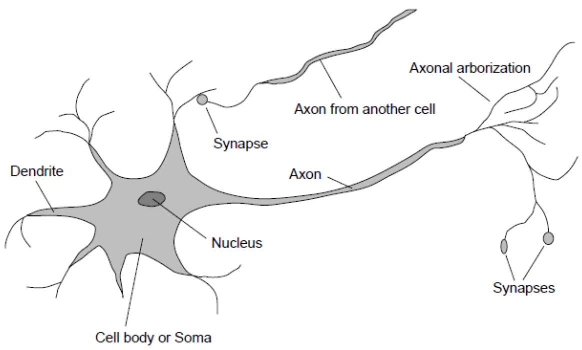

Figure 1 : Biological neural network

Key components of neural network are listed below[7].

Soma / Cell Body

Soma is the cell body of nucleus of an individual neuron.

Dendrite

Dendrite is connected with a neuron and is treated as the input to the neuron. A neuron may have multiple input dendrites.

Axon

Axon is treated as output from neuron as an electrical impulse. This axon is connected with synapses via boutons.

Synapse

Synapses are connected with axons and responsible for processing of activation of the signal to the next neuron. Synapses connect all the neurons as network and responsible for passing signal for the next level of processing.

The processing power of neural network is due to the powerful interconnections among the thousands of neurons. If signal passing from one neuron to another is delayed than it slows down the processing of neural network.

III. Artificial Neural Network

Artificial neural network idea is directly derived from the human brain nervous system. Artificial neural network consisting of a layered architecture where input layer neurons receives input, perform processing at its own and then forwarding the result to its next layer which is a hidden layer[9]. Neural network may have more than one hidden layers which are responsible for core processing of neural network. The output of the hidden layer is forwarded to the next level which is represented by output neuron. Output layer results the possible outcome. Input data including weight given to neuron is processed by the neuron using weight sum of all inputs and output the result to next neuron. The simplest form of neural network is feed forward neural network which controls the processing in one direction[9]. The neural network represented here in figure 2 is a feed forward neural network.

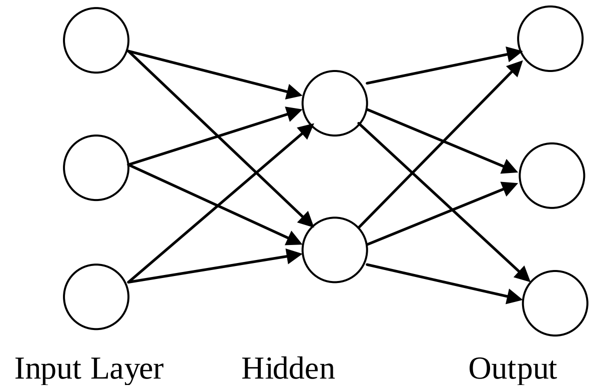

Figure 2 : Feed forward neural network

Each layer has input and output connections. A cumulative effect of all the input signals will direct the network neuron. In human brain neural network signals are of biochemical electrical signal pulses whereas in our artificial neural network signals are described as weight sum of real value. Each connecting arc is given a specific weight value along with the actual data. If this sum qualifies the expected threshold value then the neuron is activated. Threshold value and weights of the arc may be changed during the processing. At some level of time the threshold value is set to certain value once the training period of neural network has been over.

Following figure 3 represents [12] the working mechanism of a single neuron.

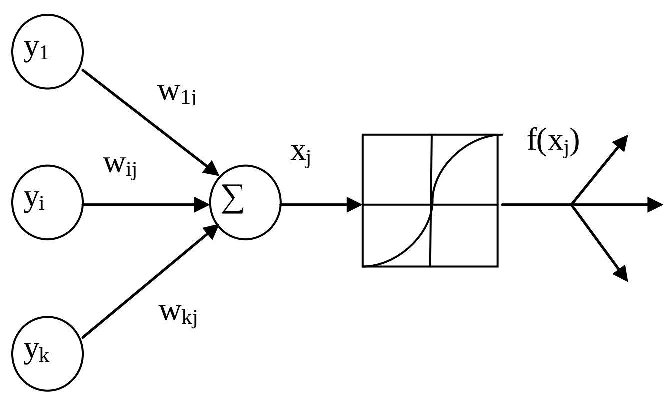

Figure 3 : Single neuron functionality

A neuron takes the input from other neurons with different weights and summation function is performed to calculate cumulative effect of its input neurons using $x_j = \sum w_{ij}\ y_i$. A nonlinear function $f(x_j)$ for example logarithmic or trigonometric is calculated and the result is forwarded to the next layer of neural network. The threshold value determines the activation of a neuron for acceptation or rejection. During neural network processing the threshold value is subject to changes until network becomes saturated.

IV. Existing Phonetic Algorithms

There exist many phonetic identification algorithms to determine similar pronunciation of different words. Following are the brief introduction of popular algorithms.

Soundex

The soundex algorithm is one of the oldest algorithms which was originally developed by Robert C. Russell and Margaret K. Odell in 1918[2][9]. The soundex algorithm returns an encoded string for the given word. The first character in return string is the starting alphabet of the given input word and remaining three characters are digits as per the defined rules.

Daitch-mokotoff soundex

By taking base of original soundex algorithm, Gary mokotoff developed D-M soundex in 1985 and its improved variation was designed by Randy Daitch to match surnames of Slavic and German languages. The D-M soundex algorithm returns the six digit numeric code for the given word[1,5].

Kolner phonetic

This algorithm was designed for German words [1].

Metaphone, Double metaphone and Metaphone 3

Lawrence Phillips Initially developed metaphone algorithm in 1990. Metaphone algorithm returns three characters encoded string for English word. Again he developed improved variation called double metaphone for other languages. He then released the third version of metaphone in 2009 with accuracy up to 99% of English words. This metaphone algorithms family formed the basis for many English spell checkers and dictionaries.

NYSIIS

New York state Identification and Intelligence System also known as NYSIIS phonetic algorithms developed in 1970 and achieved better accuracy over soundex algorithm.

Match Rating Approach

The match rating Approach MRA is a phonetic algorithm based on the distance among words developed by Western Airlines in 1977 for indexing and comparing homophonous names[1].

Caverphone

David Hood at the University of Otago in New Zealand developed the Caverphone phonetic algorithm in 2002 and revised in 2004. Primary aim of algorithm was data matching between late 19th century and early 20th century electoral rolls to commonly recognize the names[1].

By studying such algorithms researcher has developed a new algorithm[3]. The algorithm takes an English word as input text and produces the output text based on the pronunciation of the word. The output pattern is followed similar to which of metaphone algorithm. The process is simplified by applying phonetic identification rules to determine the outcome.

All these variety of algorithms can be used for spell correction and spell generators that can identify the words phonetically nearer to the given misspelled English words. Also these algorithms are the basis for determining the homophones means similar pronunciation with different spelling and different meaning. But each algorithm has specific advantages and limitations resulting in to unexpected outcome. For some set of words one algorithm is better while for other set another algorithm proves better. So, an effort of incorporating more than one algorithm has been made by constructing a neural network which require more processing of the given word but proves better performance for identifying homophones.

Three algorithms soundex, metaphone and user developed algorithms are used to form a neural network at one level and match rating at second level are implemented in Java to test the performance.

V. Phonetic Algorithms Incorporated With Artificial Neural Network

A neural network of phonetic algorithms can be formed to realize better performance. More than one algorithm in form of neural network is used to determine phonetic similarity. The resultant outcome is more accurate than any single such algorithm.. Neural network can be constructed in which input layer receives two English words. These input words are handed over to different neurons with some initial weight. Each neuron represents the separate processing of individual algorithm. We have used three different algorithms as three neurons. Each algorithm will try to determine whether the given words are homophone or not. If either of the algorithms identifies homophones then result is sent to the output neuron which will display the message. Weight can be calculated based on the recognition performance of each algorithm. If algorithms responses success then based on probability, weight can be increased otherwise decreased by calculating total experiments data and success cases. If all the algorithms are failed then only encoded strings from two algorithms are passed over to second level processing which identifies homophones liberally. So imposing constrains of both the outcome one from first encoded string to second level and second from second encoded string to second level identifies homophones then only the word are said as homophone.

The above described processing looks tricky and somehow complex but improves result far better. For calculating threshold value for individual neuron sine or other functions can be useful. Probability calculation is useful for determining the weight.

VI. Neural Network Architecture For Determining Homophones

Neural network approach and phonetic algorithms can be integrated to determine homophones. Following diagram figure 4 depicts the neural network approach for homophone identification consisting of four layers. Here for simplify the calculation the weight is taken as 1 and 0 meaning homophone recognize or not. This value along with the encoded output string is forward to second layer when first layer failed to recognize homophone. If first layer successfully identifies homophones then the processing of second layer is skipped.

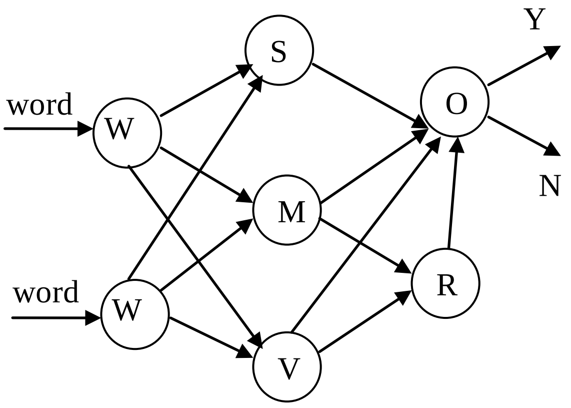

Figure 4 : Neural network of phonetic algorithms

The neural network represented consists four layers. Input layer takes two input English words this word is passed to first hidden layer consisting of three neurons namely S representing soundex, M representing metaphone and V representing user's developed algorithm to determine phonetic similarity. Initially weights can be set equally or based on the experimented set of data. The outcome of these neurons are sent to output neuron represented by O which is responsible for displaying whether the given words are homophones or not based on cumulative effect of three algorithms. If all these three algorithms are failed then encoded strings from neurons M and V are sent to neuron R which represents the Match-rating algorithm. If this algorithm identifies homophones from both the neurons M and V then result is yes otherwise no displayed through output neuron.

Homophone Identifier Neural Network Processing Mechanism

Input layer takes two English words as input.

This words are passed to all the three neurons representing separate processing by different algorithms.

The result from each algorithm passed over to the output layer. For simplicity 1 is taken for identifying homophone and 0 for failure.

If either of the neuron identifies homophone, then the output neuron displays the success message of identifying homophone andprocess terminate.

But if neither of the neuron identifies homophone then the encoded output strings from neuron M and neuronV are passed to the another neuron in hidden layer R.

Now neuron R takes two sets of encoded strings from each neuron and performs processing by appling match rating algorithm to obtain further encoded string.

If neuron R can identify as homophone for both sets of the encoded input strings then only it will pass acceptance noation to output neuron other wise failure signal.

Output layer displays wether homophones arerecognize or not either taking input from three nurons or from R neuron .

The output layer neuron O calcuates summation of all three input layers. If it reaches at thresold value which is taken here as 1 for simplicity, outputs as homophone words otherwise not. The output of hidden layer neurons may be 0, 1, 2 or 3 for constant weight. If homophone match found then it is 1 otherwise it is 0. If real weights are taken in calculation then based on the experimental data, for each algorithm separate weight is calculated and it should be multiplied with 1 or 0 depending on the matching and passed over to the output layer. Output layer may also take input in form of 1 or 0 from neuron R.

Thus the proposed neural network consisting of massive processing of textual comparison and encoding strings but proves better performance compared to performing the same task using any individual algorithm used in neural network.

VII. Testing Approach Using Sample Data

Following table represents the test experiments performed on sample data set of homophones. Here the effort is to determine the performance of each individual algorithm with compared to one in which all are united through a neural network approach.

Table 1

Testing and Evaluation of neural network with sample data set

Sr. No.	***English Input Word***	***M***	***S***	***V***	***R***	***Result***
1.	WEEK WEAK	1	1	1	-	✓
2.	PIECE PEACE	1	1	1	-	✓
3.	BED BAD	1	1	1	-	✓
4.	WOULD WOOD	0	0	1	-	✓
5.	SUN SON	1	1	1	-	✓
6.	SHIP SHEEP	1	1	1	-	✓
7.	LATER LETTER	1	1	1	-	✓
8.	LOW LAW	1	1	1	-	✓
9.	CASE CASH	0	1	1	-	✓
10.	OF OFF	1	1	1	-	✓
11.	LIVE LEAVE	1	1	1	-	✓
12.	SIGN SINE	1	0	1	-	✓
13.	SIN SEEN	1	1	1	-	✓
14.	BY BYE	0	1	1	-	✓
15.	REACH RICH	1	1	1	-	✓
16.	SORT SHORT	0	1	1	-	✓
17.	CENTER CENTRE	1	1	1	-	✓

18.	FULL FOOL	1	1	1	-	✓
19.	THEN THAN	0	1	1	-	✓
20.	FILL FEEL	1	1	1	-	✓
21.	FOUR FOR	1	1	1	-	✓
22.	MAT MET	1	1	1	-	✓
23.	MERRY MARRY	1	1	1	-	✓
24.	WEATHER WETHER	0	1	1	-	✓
25.	BALL BAWL	1	1	1	-	✓
26.	CELL SELL	1	0	1	-	✓
27.	CEIL SEAL	0	0	1	-	✓
28.	WAIT WEIGHT	1	0	1	-	✓
29.	HIGHER HIRE	0	0	0	1	✓
30.	KNIGHT NIGHT	1	0	1	-	✓
31.	DEAR DEER	1	1	1	-	✓
32.	ROOT ROUTE	1	1	1	-	✓
33.	DESCENT DISSENT	1	1	1	-	✓
34.	KNOW NO	1	0	0	-	✓
35.	POLE POLL	1	1	1	-	✓
36.	TIDE TIED	1	1	1	-	✓
37.	DIE DYE	0	1	1	-	✓
38.	MAIL MALE	1	1	1	-	✓
39.	SOME SUM	1	1	1	-	✓
40.	BITE BYTE	1	1	1	-	✓
41.	ROSE RAWS	1	1	1	-	✓
42.	ROTE WROTE	1	0	1	-	✓
43.	BRAKE BREAK	1	1	1	-	✓
44.	VICE VISE	1	1	1	-	✓
45.	I EYE	0	0	0	-	×

46.	FAIR FARE	1	1	1	-	✓
47.	CUE QUEUE	1	0	1	-	✓
48.	HOUR OUR	0	0	0	1	✓
49.	CATTELE KETTLE	1	0	1	-	✓
50.	CANVAS CANVASS	1	1	1	-	✓
51.	LESSON LESSEN	1	1	1	-	✓
52.	MEET MEAT	1	1	1	-	✓
53.	HAIR HARE	1	1	1	-	✓
54.	GATE GAIT	1	1	1	-	✓
55.	HEAL HEEL	1	1	1	-	✓
56.	ACCEPT EXCEPT	0	0	0	1	✓
57.	CAST CASTE	1	1	1	-	✓
58.	CHORD CORD	0	1	1	-	✓
59.	CHEQUE CHECK	1	1	1	-	✓
60.	CASH CASE	0	1	1	-	✓
61.	BIRTH BERTH	1	1	1	-	✓
62.	THERE THEIR	1	1	1	-	✓
63.	CLOCK CLOAK	1	1	1	-	✓
64.	SHE SEE	0	1	1	-	✓
65.	SEE SEA	0	1	1	-	✓
66.	TWO TO	0	1	0	-	✓
67.	TOO TO	1	1	1	-	✓
68.	MADE MAID	1	1	1	-	✓
69.	ALOUD ALLOWED	0	1	0	-	✓
70.	PLANE PLAIN	1	1	1	-	✓
71.	WHICH WITCH	0	0	0	0	×
72.	TAIL TALE	1	1	1	-	✓
73.	SAIL SALE	1	1	1	-	✓

74.	BLUE BLEW	1	1	0	-	✓
75.	SENT CENT	1	0	1	-	✓
76.	FAZE PHASE	1	0	0	-	✓
77.	FEAT FEET	1	1	1	-	✓
78.	HEAL HEEL	1	1	1	-	✓
79.	HI HIGH	1	0	0	-	✓
80.	HOLE WHOLE	1	0	0	-	✓
81.	KNOT NOT	1	0	1	-	✓
82.	PEEK PEAK	1	1	1	-	✓
83.	ROLE ROLL	1	1	1	-	✓
84.	RING WRING	1	0	1	-	✓
85.	SCENE SEEN	1	1	1	-	✓
86.	SOME SUM	1	1	1	-	✓
87.	STEAL STEEL	0	1	1	-	✓
88.	VARY VERY	1	1	1	-	✓
89.	WEE WE	1	1	1	-	✓
90.	ASSENT ASCENT	1	1	1	-	✓
91.	BEACH BEECH	1	1	1	-	✓
92.	BOARDER BORDER	1	1	1	-	✓
93.	BEE BE	1	1	1	-	✓
94.	CARRIER CAREER	1	1	1	-	✓
95.	DUAL DUEL	1	1	1	-	✓
96.	DISCRETE DESCREET	1	1	1	-	✓
97.	GREAT GRATE	1	1	1	-	✓
98.	HEAR HERE	1	1	1	-	✓
99.	LESSEN LESSON	1	1	1	-	✓
100.	PORE POUR	1	1	1	-	✓
101.	RAISE RAYS	1	1	1	-	✓
102.	SHEAR SHEER	1	1	1	-	✓
103.	TIED TIDE	1	1	1	-	✓
104.	WASTE WAIST	1	1	1	-	✓
105.	WRING RING	1	0	1	-	✓

Table – 1 (Sample Data Test)

Performance Analysis And Comparison Of Sample Test Data Result

Table 2

	Analysis and success percentage of each algorithm					
Sr. No.	**Result**	**M**	**S**	**V**	**R**	**Success**
1.	**Total**	85	83	93	3	103
2.	**Percentage**	80.95	79.04	88.57	2.86	98.09

Table – 2 (Analysis of result)

From the table 1 sample data experiment, Table 2 is derived. Table 2 represents the performance of each algorithm and cumulative performance of neural network as a whole. First column is serial number. Second column is total number of success from 105 experimental data and percentage of success. Third, fourth, fifth, and sixth column headed with M, S, V and R represent the individual algorithm success with percentage applied here on 105 homophones. Last column is the total success and percentage entirely of approach. The match rating algorithm represented by R, applied here invoked only when all the preceding algorithms are failed. Further the input to neuron R is the encoded pair of input words from both M and V. If it success in both, then only the result is treated as successful otherwise fails. By incorporating more than algorithms the result is achieved here 98% which is greater than any individual implementation.

VIII. Conclusion

From the testing result of Table – 1, the model gives acceptable performance to identify homophone compared to any individual algorithm. The approach can be further experimented on other data to see the effect. The neural network described here processes massive text data using multiple algorithms. The performance obtained here can be further improved by incorporating other phonetic algorithms based the application. The performance is obtained here due to the increased cost of processing. Further optimization on this text processing can be applied to achieve the desired performance in terms of processing and computer power[13]. The goal of neural network model represented here is to incorporate neural network with phonetic identification algorithms. To performance to identify homophone and processing are inverse proposals. But due to availability of powerful computer second issue can be resolved. The weights used here in calculation are just binary but actual real valued weight can be calculated by applying nonlinear function and success ration of each algorithm. The neural network can be trained by experimenting on homophone words which will reflect the weight of joining arcs. Each time a neural network weight can be updated or taken fixed after training is over. The text processing is performed on word by word bases but it can be processed character by character which requires more processing.

REFERENCES

[1] Phonetic Comparison Algorithms By BRETT KESSLER Washington University in St. Louis Volume 103:2 (2005) 243–260

[2] Analysis and Comparative Study on Phonetic Matching Techniques, Rima Shah, Dheeraj Kumar Singh, IJCA Volume 87 – No.9, February 2014

[3] Name and address matching strategy – White Paper December 2010

[4] Vimal P Parmar and C K Kumbharana. Article: Study Existing Various Phonetic Algorithms and Designing and Development of a working model for the New Developed Algorithm and Comparison by implementing it with Existing Algorithm(s). International Journal of Computer Applications 98(19):45-49, July 2014

[5] Vimal P Parmar, C K Kumbharana and Apurva K Pandya - Determining the Character Replacement Rules and Implementing Them for Phonetic Identification of Given Words to Identify Similar Pronunciation Words. Futuristic Trends on Computation Analysis and Knowledge Management (ABLAZE) 2015 International Conference at Greater Noida, India Pages : 272-277 Print ISBN : 978-1-4799-8432-9 DOI : 10.1109/ABLAZE.2015.7155010

[6] http://www.massey.ac.nz/~wwpapajl/evolution/assign2/BC/ANN.htm

[7] http://www.ent.mrt.ac.lk/~ekulasek/ami/PartA.pdf

[8] http://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7

[9] Multi layer neural network and soundex algorithms for text recognition model. - Mandeep Kaur, Sheetal Mahajan – IJCSET ISSN 2229-3345 Vol 4 No. 08 Aug 2013, 1190.

[10] Artificial Neural Network by B. Yegnanarayana.

[11] Basics of Artificial Neural Network - Sakshi Kohli, Surbhi Miglani, Rahul Rapariya IJCSMC, Vol. 3, Issue. 9, September 2014, pg.745 – 751 ISSN 2320-088X

[12] Artificial neural network and its application Girishkumar Jha I.A.R.I. New Delhi.

[13] Use of neural network in pattern recognition – Jayanakumar Basu, Debnath Bhatacharya, Tai-hoon kim International Journal of software engineering and its applications Vol. 4, No.2, April 2010.

AUTHORS

First Author – Vimal P.Parmar, Research Scholar, Department of Computer Science, Saurashtra University, Rajkot, Gujarat, INDIA, parmarvimal1976@yahoo.co.in
Second Author – Dr. CK Kumbharana, Head, Guide, Department of Computer Science, Saurashtra University Rajkot Gujarat, INDIA, ckkumbharana@yahoo.com