VAWKUM Transactions on Computer Sciences
http://vfast.org/journals/index.php/VTCS@ 2024, ISSN(e): 2308-8168, ISSN(p): 2411-6335
Volume 12, Number 2, July-December 2024 *pp: 253-263*

Bit Pattern based Sindhi Character Recognition using Neural Network

Deepa Jai[1*], **Saima Siraj**[2], **Shamshad Lakho**[1], **Waqas Ali**[2], **Baqar Ali**[2], **Akhtar Hussain**[1]

[1]Department of Computer Science, Quaid-e-Awam University of Engineering Science and Technology Nawabshah, Pakistan; [2]Department of Information Technology, Quaid-e-Awam University of Engineering Science and Technology Nawabshah, Pakistan

Keywords: *Bit Patterns, Sindhi Alphabets, Pattern Recognition, Artifical Neural Network, Multi-Layer Perceptron*

Journal Info:
Submitted:
October 30, 2024
Accepted:
December 16, 2024
Published:
December 30, 2024

Abstract

In this paper, "bit pattern" based character recognition for Sindhi language has been presented. The characters of sindhi language are very much complicated and hard to recognize for particular domain. Although,a numerous studies had been carried out for recognition but precious studies were based on image recognition. Developing with innovation and different studies involves the use of bit patterns for characters and anticipate outcome on the basis of provided input pattern. A data set with nine no. of inputs and six outputs for each character were framed. In this study, patterns were used due to their computational complexity constant. The matrix 3x3 for input patterns that were uniquely framed for all characters and output were generated in form of binary pattern for the particular character sequence numbers. The system reads the 3X3 matrix in clock wise pattern to get input pattern and system matches with already framed data set. The data set was trained by using Neural Network Model, Multi-Layer Perceptron (MLP) with significant number of hidden layers to get measurable results. The accuracy of 82.6% has been achieved in the study.

***Correspondence author email address:** Saimasiraj@quest.edu.pk
DOI: 10.21015/vtcs.v12i2.1957

1 Introduction

The Sindhi language is one of the oldest languages of the world having fifty-two alphabetical letters and space to adopt several other languages lexicons. This language is written, read, and spoken all over the world. The Sindhi language is complex grammatically and rich morphologically.

The grammar of Sindhi language is not the same as the grammar of English and other languages even the meaning and sense of understating of Sindhi lexicons are different. The diacritics used in Sindhi text is changing the meaning, number, and gender of the Sindhi lexicons. In this modern era of automation and technology,

Artificial Intelligence being considered as giant in field of computer Science. It is the field that encompasses on concept and development for performing tasks that are based on human intelligence. The field includes different sub area like visual perception, speech recognition, language translation and pattern recognition. In this study we have worked on pattern recognition of Sindhi language character with the help of Artificial intelligence model Artificial Neural Networks (ANNs). Pattern Recognition is basically the process of matching fundamental components of text, such as words, syntax and a particular pattern they use to follow. It is being utilized in many fields including classification of contents, machine translations and for checking grammar.

This technique is frequently taught using supervised learning methods in which we train the data with proper number of inputs and outputs and on the basis of that the ANN identifies the output for given input patterns. In contrast to a brief introduction to artificial intelligence, pattern recognitions and artificial neural networks, Authors discussed the Braille system and typical methods used by blind people to read and write it. It also reviews network training, testing, and outcomes for these two procedures. In this work, an artificial neural network (ANN) will be created to recognize photographs of Cyrillic characters. To replicate the surroundings of the actual world, some of them will be noised in some way. To be utilized in identifying them, ANN will be trained and put through testing [1].

With the idea of Pattern Recognition and Artificial Neural Networks ANNs, we have experimented an idea of matching patterns of Sindhi Characters like Braille System for recognition. There are multiple researches that have been done on different languages to recognize their characters and words. To give enhancement to Sindhi Language we have worked for character recognition of Sindhi language using pattern matching. With the review of Machine Learning techniques, we have worked on supervised learning, for the training and testing we have used Artificial Neural Networks and made our data base by creating unique input codes based on nine inputs with matrix of 3 X 3 for all 52 characters of language as given for braille language [1]. For reading of the 3X3 matrix we used method of Local Binary Pattern (LBP) that gave a measurable result [2]. In the similar way we have also designed binary code outputs for all the characters of Language Sindhi [3].

2 Related Work

In the past years many researchers have worked on this field of pattern matching using image processing and image segmentation. Our goal for this research is to mainly recognize the characters on the basis of bit patterns and machine learning techniques. Few researchers have used ML and NLP techniques to create chatbots for predicting writing and speaking by using previous knowledge and training .

According to the previous related work one of the authors has proposed a system for visually impaired people to recognize Braille language characters. For that they have used ANN model to train and test their designed data base based on bit patterns [4]. As Braille is also a part of pattern recognition so the work has also been done using back propagation method used in Artificial Neural Networks and the results of speech recognition has also been translated from Braille to English Language [5]. With reference of Image processing to recognize characters of any language there are researchers who have worked on Sindhi alphabets with the help of Convolution Neural Network (CNN) model along with use of max-pooling layers. According to the authors using max-pooling layer along with CNN increases the performance and gives measurable results [6]. After reviewing various methods in the field of character recognition we came across that Local Binary Pattern (LBP) have also been used by many researchers to identify their characters of their languages. One of the researchers worked on Persian/Arabic characters by smoothing, turning original image in to Grey scale image segmented the image in to multiple parts and then had trained through multi-layer perceptron to extract features using LBP operators.

Ultimately this model gave better results than already proposed models on data set of HODA which contains almost 60000 train samples and 20000 test samples [7]. Going through usage of LBP and its variants in reference

with pattern matching work has also been done with help of five LBP-based techniques like LBP, CS-LBP, CS-LTP, CS-LMP, and U2LBP. These techniques along with Support Vector Machine SVM as classifier have been used on Arabic digits data set of MAHDBase. These combinations of techniques had given remarkable result in this field, also they have proved that if we use sliding window along with LBP we don´t need to use any of its extension as it gives fine results too [8].

As we can see that in Urdu language isolated characters are used to create any word. A training system is developed to identify character in those isolated words with the help of image pre-processing, character and word segmentation by making an xml file to be trained [9]. A CNN approach usually called summation along with deep residual network with short cut connections are given to extract features of handwritten Sindhi letters and classify them [10].

A system for identifying patterns in English Braille, Using the English Braille Grade-1 data set and reliable machine learning techniques like SVM, DT and KNN (K-Nearest Neighbor) along with RICA(Reconstruction Independent Component Analysis) and PCA(Principal Component Analysis) based feature extraction techniques [11]. Optical Braille Identification has also been done using Object detection convolution neural network, the method successfully recognize Braille texts that have been shot by a smart phone camera under normal circumstances with a remarkable accuracy [12]. We can work on the same picture collection, preparation (noise reduction, image alignment), and cells segmentation stages as used in most of the implementations of braille recognition that use standard image processing techniques. After that the segmented braille cells can be connected to related alphabets in a variety of ways. One of the most common methods is to turn the data points into 3x2 binary matrices or 6-bit binary values and then recognize the cells by comparing those values to records already in a database. Instead of commonly used methods ANNs model can be used to match the binary values [13].

Earlier suggested deep CNNs' significant computational cost made them unsuitable for real-time applications. In contrast to other methods, they concentrated on the problem of network computational cost and suggested a novel method for Braille character identification that makes use of the best picture pre-processing methods and a compact deep convolution neural network. The original CNN model uses a dynamic type CNN modification approach employing the inverted residual block (IRB) module to address the issue of network computing cost [14]. A technique of pattern recognition to recognize braille letters with the help of their code images using web cam. Real time input images needed to be captured for data set and then the data set has been trained and tested in artificial neural networks [15] .They have compared their data with data set made using old scanner. This method is used for sight impaired to recognize braille characters with more Accuracy than the old scanner provides. They have set the codes for each character and used those code images to be trained in their data set [16].

As we have been gone through the matrix reading so there are multiple ways researchers have used to read a 3X3 matrix. At some place by going through description of Binary Patterns it was given that by threshold a 3x3 neighborhood with the center pixel value and using the result as a binary integer, the operator provides a label to each pixel of an image [17]. The circular 0 and 1 resulting numbers might be read either clockwise or anticlockwise in different publications. As can be seen in the accompanying image, the binary result will be produced in this study by reading the values clockwise, starting at the top left corner [18]. On the other place one of the researchers, while providing a unique method for the detection of hand motions under occlusion with face. For that feature extraction they have used Sobel-LBP [19].

The resultant force field image is split into nine areas, and each region is then applied to SLBP to extract feature vector histograms by performing a 3X3 neighborhood operation on each pixel using a center point in the form of binary representation in a non-uniform pattern. To retain the inputs of those nine areas they had also read the 3X3 matrix in clock wise order [20].

3 Proposed Framework

With reference of above research papers and their content we come across that Sindhi character recognition is done using image recognition and in printed form, but for that we should have images of characters. To overcome the problem of images and reduce the cost of matching we can make unique binary codes for all characters to make it more efficient and easy to match according to number of character, therefore there is need of such research that use Pattern Matching using Machine learning methods for character recognition of Sindhi Language.The main aim of this research work is to recognize Sindhi language characters with pattern matching by using Neural Networks Model. To make our system work properly we used Neural Network Model, Multi-Layer Perceptron (MLP) to train and test our data set to get the proper outcome. As our system work for Sindhi Language and it is consist of 52 letters so we have created our data for all these characters.

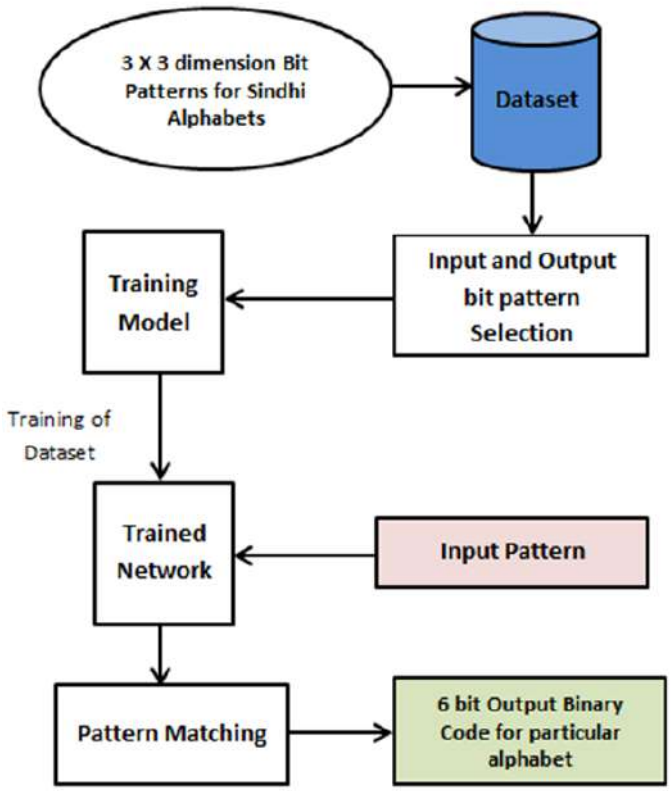

Figure 1. Research Methodology

Figure 1 describes the flow of system that initially we created the data set and after finalizing that our system separates input and output patterns for all characters for pattern matching, with the same data set we trained the MLP (Multi-Layer perceptron) Neural Network model and in the result we got our trained network that, then we used for getting our test result for all character patterns. The testing occurs when we provide the Input Pattern to network and by pattern matching it provides us particular binary code for the sequence number of the letter.

4 Model Explanation

As given in Figure 1 we have divided our system in different parts, each system provides some result and transfers it to other next one. Further explanation is given below:

4.1 Bit Patterns

For working on pattern matching, we have created patterns for Sindhi Characters in form of 0 and 1 in matrix dimension of 3 X 3 that are uniquely defined for all 52 characters.

4.2 Training Data Set

For data set we used the patterns that have been created in bit patterns for all 52 characters, input patterns were in the form of given figures 2,3 . Each character is being mapped in nine inputs in matrix dimension of 3X3.

Sindhi Character	Input Patterns	Output Patterns	Data Set Patterns	Sindhi Character	Input Patterns	Output Patterns	Data Set Patterns
ا	0 0 1 0 0 1 0 0 1	2(000001) = 1	001110000 000001	ڄ	1 0 1 1 0 0 1 1 1	2(010001) = 17	101011110 010001
ب	1 0 1 1 1 1 0 1 0	2(000010) = 2	101101011 000010	ڃ	0 0 0 0 1 1 0 0 0	2(010010) = 18	000100001 010010
پ	1 0 1 1 1 1 0 1 1	2(000011) = 3	101111011 000011	ڊ	1 1 0 0 0 1 1 1 0	2(010011) = 19	110101100 010011
ت	0 0 0 0 0 0 0 0 0	2(000100) = 4	000000000 000100	ڋ	1 1 1 0 0 1 1 1 0	2(010100) = 20	111101100 010100
ٹ	1 1 0 1 0 1 1 1 1	2(000101) = 5	110111110 000101	ڌ	1 1 1 0 0 0 0 0 0	2(010101) = 21	111000000 010101
ث	1 1 1 1 1 1 1 1 1	2(000110) = 6	111111111 000110	ڍ	1 1 0 0 0 1 1 1 1	2(010110) = 22	110111100 010110
ٹ	1 1 0 1 1 1 1 1 1	2(000111) = 7	110111111 000111	ڏ	0 0 0 0 0 0 0 1 1	2(010111) = 23	000011000 010111
ٹ	0 1 0 1 1 1 1 1 1	2(001000) = 8	010111111 001000	ڈ	1 0 1 0 0 0 0 0 0	2(011000) = 24	101000000 011000
ٹ	0 1 1 1 1 1 1 1 1	2(001001) = 9	011111111 001001	ڙ	0 0 1 0 0 1 0 1 0	2(011001) = 25	001101000 011001
ٹ	1 0 1 1 1 1 1 1 1	2(001010) = 10	101111111 001010	ژ	1 1 1 1 1 1 0 1 0	2(011010) = 26	111101011 011010
ٹ	1 1 1 1 1 0 1 1 1	2(001011) = 11	111011111 001011	ڑ	1 0 1 0 0 1 0 1 0	2(011011) = 27	101101000 011011
ٹ	0 0 0 0 1 0 0 0 0	2(001100) = 12	000000001 001100	ں	0 1 1 1 1 0 1 1 0	2(011100) = 28	011001111 011100
ٹ	1 0 0 1 0 0 1 1 0	2(001101) = 13	100001110 001101	ڻ	1 1 1 1 1 0 1 1 0	2(011101) = 29	111001111 011101

Figure 2. Data Description

Whereas for output we have mapped our letter according to specific sequence of character for instance 1st character is represented by 000001, 2nd as 000010 and so on till last character 52 as 110100.

4.3 Input and Output Bit Pattern Selection

As we have designed our system in a way where input patterns are going to be matched to provide output pattern so for that the data is being separated into two portions input and output first 9 patterns in the data are input patterns and next six patterns are showing the number of the alphabet against that input pattern.

4.4 Training Phase

After creation of data set we trained the data set in Multi-Layer Perceptron that included Input Layer Output Layer and Hidden Layers we have trained almost 80per random data patterns to supervise our data.

4.5 Testing Phase

When the training phase was completed and our model was trained, we tested our data set to know whether our system is working fine or not. For that we used 100% data, we provided input patterns and the system matched the input pattern and return output pattern according to the sequence number of characters as given in results and discussion.

ج	0 0 1 0 0 0 0 0 1	2(001110) = 14	001010000 001110	ص	0 1 1 1 1 1 1 1 0	2(011110) = 30	011101111 011110
ح	0 0 1 0 0 0 0 0 0	2(001111) = 15	001000000 001111	ض	1 1 1 1 1 1 1 1 0	2(011111) = 31	111101111 011111
خ	0 0 1 0 1 1 0 0 0	2(010000) = 16	001100001 010000	ط	1 0 0 1 1 0 1 1 0	2(100000) = 32	100001111 100000
ظ	1 0 1 1 1 0 1 1 0	2(100001) = 33	101001111 100001	گھ	0 1 0 0 0 0 1 1 1	2(101011) = 43	010011100 101011
ع	1 1 0 1 0 0 1 1 0	2(100010) = 34	110011110 100010	گ	1 1 1 1 0 1 1 1 1	2(101100) = 44	111111110 101100
غ	1 1 1 1 0 0 1 1 1	2(100011) = 35	111011110 100011	ل	0 0 1 0 0 1 1 1 1	2(101101) = 45	001111100 101101
ف	0 0 1 1 1 1 1 1 1	2(100100) = 36	001111111 100100	م	0 1 1 0 1 1 0 1 0	2(101110) = 46	011101011 101110
ق	1 0 0 0 1 0 0 0 0	2(100101) = 37	100000001 100101	ن	0 0 0 1 1 1 1 1 1	2(101111) = 47	000111111 101111
ق	0 1 1 1 0 0 1 1 1	2(100110) = 38	011011110 100110	ڈ	0 0 0 1 0 1 1 1 1	2(110000) = 48	000111110 110000
ے	0 0 1 0 1 0 1 1 1	2(100111) = 39	001011101 100111	و	0 1 1 0 1 1 1 1 1	2(110001) = 49	011111101 110001
ک	0 0 1 1 0 1 1 1 1	2(101000) = 40	001111110 101000	ھ	1 0 0 1 0 0 0 0 0	2(110010) = 50	100000010 110010
گ	0 1 1 1 0 1 1 1 1	2(101001) = 41	011111110 101001	ء	1 0 1 0 1 0 1 0 1	2(110011) = 51	101010101 110011
گھ	1 0 1 1 1 1 1 1 0	2(101010) = 42	101101111 101010	ي	1 1 1 1 1 1 0 1 1	2(110100) = 52	111111011 110100

Figure 3. Data Description

5 Results and Discussion

5.1 Neural Network Description

As already discussed, that we have used artificial neural network model for the training and testing of our system.

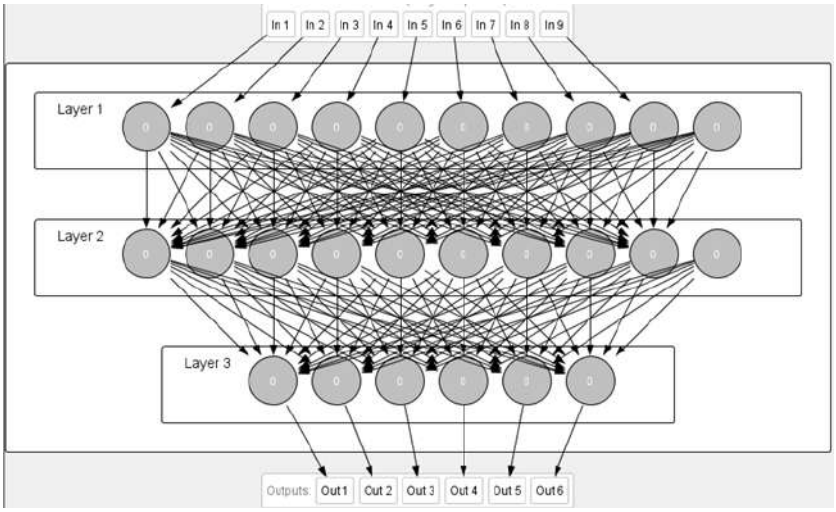

Figure 4. Neural Network Model

Figure 4 shows the model of neural network that is based on three layers at the very first layer we have nine inputs as we have input patterns for each character that are uniquely defined in the created data set. While using multilayer perceptron model, second layer is hidden layer where 9 neurons are working to calculate the output. At the end the last layer is of output, where the binary value will be displayed according to the number of characters in the sequence of alphabets. After creating the data set, we trained our model on the data set with following

parameters given in Table 1 . By giving the below given parameters we have trained our data set with 30,000 numbers of iterations. After training when we tested our data set it gives Mean square Error of 0.035.

Table 1. Training Parameters

NN Model Input Parameter	Value
Learning Rate	0.2
Momentum	0.7
Maximum Error	0.01

6 Experimental Results

After doing the training of data set we tested patterns of different characters.

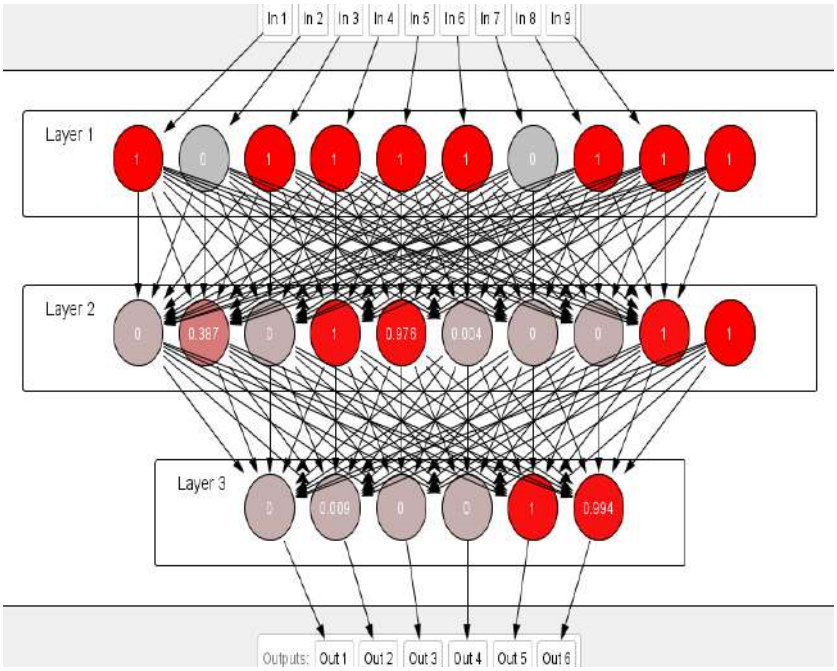

Figure 5. Experimental Result 1

Figure 5 shows the test result of character which is on 3rd number in Sindhi alphabet sequence. We have created its input pattern that is after reading clockwise is 1 0 1 1 1 1 0 1 1 and as according to sequence we got its output 000011 that is also given in Figures 2,3

After doing the first test, we then tested the patterns of character with input pattern 0 0 0 1 0 0 0 0 1 and it is at number 18 according Sindhi alphabet sequence, so because of that its output should be binary pattern of (010010)2 = .18

The third experiment we did on pattern of character for which we have designed input pattern after reading it clock wise as 1 1 1 1 0 1 0 1 1 and it is at 26 number in alphabetical sequence for that we have got its output 011010 as given in Figure 7 , that is correct output according to binary code of its sequence number which is 26.

Further testing our system, we tested character number 32, and according to its sequence number we got its result is 100000 as given in Figure 8, however its designed input pattern was 1 0 0 0 0 1 1 1 1. This pattern is also matched correctly as per required input.

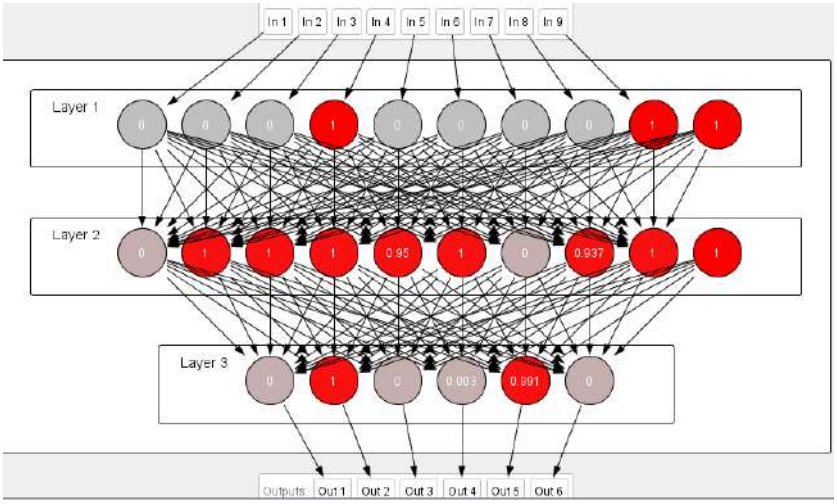

Figure 6. Experimental Result 2

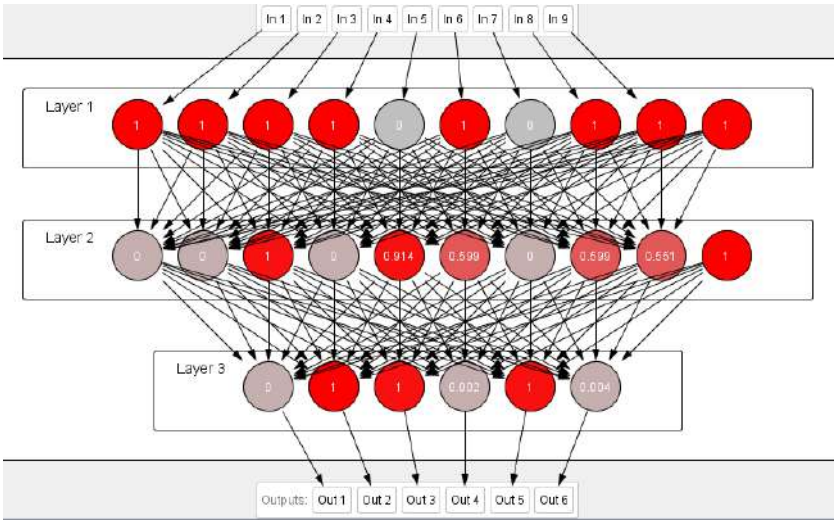

Figure 7. Experimental Result 3

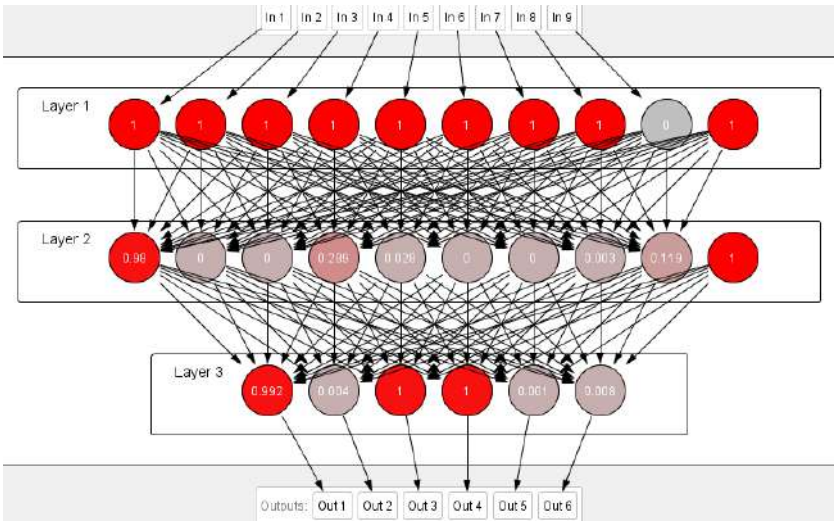

Figure 9. Experimental Result 5

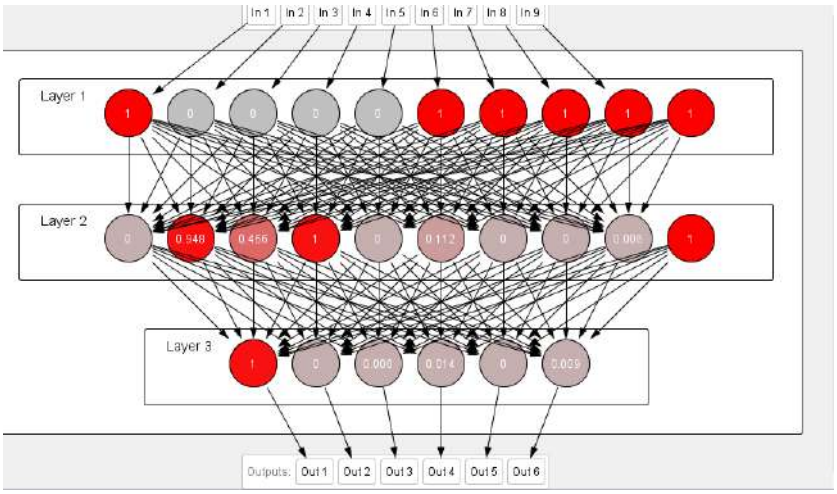

Figure 8. Experimental Result 4

Next testing pattern is also from the Sindhi Language characters which has input pattern 1 1 1 1 1 1 1 1 0 and it is at 44th number in its sequence. Figure 9 shows its output 101100 that is also binary pattern for .44 Neural Network model MLP was very useful to train and test our system. We have tested all patterns, also some patterns that were not trained. With the testing results we have generated a confusion matrix which shows that there are 86 True positive values in overall testing and our systems gives 82.6% accuracy to pattern matching as the system successfully matches 43 out of 52 characters of Sindhi language. It gives precision; recall and F1 score 0.83 that is also a measurable result for the applied model.

7 Confusion Matrix Results

TP = 86

Overall Accuracy = 82.69%

Table 2. Confusion Matrix Results

Class	n(truth)	n(classified)	Accuracy	Precision	Recall	F1 Score
1	52	52	82.69	0.83	0.83	0.83
2	52	52	82.69	0.83	0.83	0.83

8 Conclusion and future Recommendations

Our system is pattern recognition system that recognizes Sindhi Language characters; in this system we have used Artificial Neural Networks model multi-layer perceptron. The model shows us a very good performance rate. The results of our Designing and implementation have proven success to our system. We have achieved our objectives from creation of data set to finally getting measurable results. According to experimental results of recognition of Sindhi characters using Neural Network our testing accuracy is almost 82.6% that is a remarkable result for the method. On the basis of results, it can be concluded that, using Neural Network model for recognition and pattern matching is very efficient and easy to use as it gives programming network that makes easy to train and test a large number of data set of patterns and images. In future, to further exceed this system the experimental results can be improved from the current results and also, we can add the new objective to recognize character present in combined words with the same methods of bit patterns. Educational Tools: Development

of educational applications that teach the Sindhi language or facilitate learning in Sindhi can support language preservation and promote literacy.

Studies that compare Sindhi NLP techniques with those of related languages can provide insights and improve methodologies. Cultural and Contextual Awareness: Ensuring that NLP applications are culturally aware and consider the social and contextual factors surrounding language. Collaborating with academic institutions, NGOs, and tech companies can foster innovation and deployment of Sindhi NLP technologies.

By focusing on these areas, future research and development in Sindhi NLP can significantly enhance the usability and functionality of technology for Sindhi speakers, ultimately contributing to the preservation and growth of the language in the digital age.

Author Contributions

Deepa Jai: Conceptualization, Methodology, Software. **Saima Sira**: Supervision, Data curation.**Shamshad Lakho**: Visualization, Investigation. **Waqas Ali**: Software, Validation. **Baqar Ali**: Writing- Reviewing and Editing **Akhtar Hussain** Writing- Original draft preparation.

Compliance with Ethical Standards

Declare any potentially competing interests, financial or otherwise see the example It is declared that all authors don't have any conflict of interest. It is also declared that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

References

[1] K. Smelyakov, A. Chupryna, D. Yeremenko, A. Sakhon, and V. Polezhai, "Braille character recognition based on neural networks," in *2018 IEEE Second International Conference on Data Stream Mining Processing (DSMP)*, pp. 509–513, 2018.

[2] L. Sánchez López, "Local binary patterns applied to face detection and recognition," Master's thesis, 2010.

[3] S.-C. Ng, C.-P. Kwok, S.-H. Chung, Y.-Y. Leung, and H.-S. Pang, "An intelligent banknote recognition system by using machine learning with assistive technology for visually impaired people," in *2020 10th International Conference on Information Science and Technology (ICIST)*, pp. 185–193, 2020.

[4] A. A. Chandio, M. Leghari, M. Leghari, and A. H. Jalbani, "Multi-font and multi-size printed sindhi character recognition using convolutional neural networks," *Pakistan Journal of Engineering and Applied Sciences*, 2019.

[5] M. Biglari, F. Mirzaei, and J. G. Neycharan, "Persian/arabic handwritten digit recognition using local binary pattern," *International Journal of Digital Information and Wireless Communications (IJDIWC)*, vol. 4, no. 4, pp. 486–492, 2014.

[6] E. Al-wajih and R. Ghazali, "An enhanced lbp-based technique with various size of sliding window approach for handwritten arabic digit recognition," *Multimedia Tools and Applications*, vol. 80, pp. 24399–24418, 2021.

[7] T. Nawaz, S. Naqvi, H. ur Rehman, and A. Faiz, "Optical character recognition system for urdu (naskh font) using pattern matching technique," *International Journal of Image Processing (IJIP)*, vol. 3, no. 3, p. 92, 2009.

[8] A. A. Chandio, M. Leghari, A. Orangzeb Panhwar, S. Zaman Niza-mani, and M. Leghari, "Deep learning-based isolated handwritten sindhi character recognition," *Indian Journal of Science and Technology*, vol. 13, no. 25, pp. 2565–2574, 2020.

[9] S. Shokat, R. Riaz, S. S. Rizvi, I. Khan, and A. Paul, "Characterization of english braille patterns using automated tools and rica based feature extraction methods," *Sensors*, vol. 22, no. 5, p. 1836, 2022.

[10] I. G. Ovodov, "Optical braille recognition using object detection neural network," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1741–1748, 2021.

[11] C. Li and W. Yan, "Braille recognition using deep learning," in *Proceedings of the 4th International Conference on Control and Computer Vision*, pp. 30–35, 2021.

[12] T. Kausar, S. Manzoor, A. Kausar, Y. Lu, M. Wasif, and M. A. Ashraf, "Deep learning strategy for braille character recognition," *IEEE Access*, vol. 9, pp. 169357–169371, 2021.

[13] J. Subur, T. A. Sardjono, and R. Mardiyanto, "Braille character recognition using find contour and artificial neural network," *JAVA Journal of Electrical and Electronics Engineering*, vol. 14, no. 1, 2016.

[14] A. Hussain, A. R. Abbasi, and N. Afzulpurkar, "Detecting & interpreting self-manipulating hand movements for student's affect prediction," *Human-Centric Computing and Information Sciences*, vol. 2, pp. 1–18, 2012.

[15] S.-B. Cho, "Neural-network classifiers for recognizing totally unconstrained handwritten numerals," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 43–53, 1997.

[16] Y. Boulid, A. Souhar, and M. Elyoussfi Elkettani, "Handwritten character recognition based on the specificity and the singularity of the arabic language," 2017.

[17] C. Saha, R. H. Faisal, and M. M. Rahman, "Bangla handwritten character recognition using local binary pattern and its variants," in *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pp. 236–241, IEEE, 2018.

[18] A. Gattal and F. Abbas, "Isolated handwritten digit recognition using lpq and lbp features," in *Proceedings of the 10th International Conference on Information Systems and Technologies*, pp. 1–5, 2020.

[19] R. M. Mohammadpoor, "Recognition of persian handwritten numbers using lbp-hog descriptor," *Bull. Environ. Pharmacol. Life Sci*, vol. 4, no. 1, pp. 284–290, 2015.

[20] M. F. Kader and K. Deb, "Neural network-based english alphanumeric character recognition," *International Journal of Computer Science, Engineering and Applications*, vol. 2, no. 4, p. 1, 2012.