

Nepali Consonant Character Classification Using Artificial Neural Network

Team:

Ashok Subedi

Giridhari Paudel

Indra Paudel

Pitambar Mahato

Contents

- 1) Introduction
- 2) Literature Review
- 3) Requirement Analysis and Feasibility study
- 4) Methodology
- 5) Structuring System Requirements
- 6) Testing
- 7) Result and Analysis

Introduction

- Nepali Character is one of the most spoken language in Nepal.
- Artificial Neural Network provides classification and regression capabilities to the machine.
- Backpropagation is the algorithm that was used during the training of the neural networks.

Problem Statement

- To change the hardcopy into softcopy form is the time consuming task.
- Without the classification or recognition of the characters it is not possible to make the OCR system.
- So we are here to classify the nepali characters

Objective

- To build a neural network model that can classify the scanned or handwritten isolated Nepali characters.
- To classify the Nepali consonant characters.

Project Questions

- Does ANN gives the optimum result for the classification of nepali characters?
- What is the main purpose of that project?
- What methods is going to be used for the achievement of the results in this project?

Scope and Limitations

- Our system recognizes the nepali isolated scanned or handwritten character but in case of the joined set of characters our system can't work.
- Our system only works for the images of size 32×32 but in case of the images greater than that the system can't work.

Applications

- Banking
- Legal
- HealthCare
- Other sectors

Literature Review

- Related Works

- Character Recognition (CR) is somewhat limited until 1980 due to lack of powerful hardware and data perception devices.
- The periods from 1980-1990 witness a growth in CR system development due to rapid growth in information technology.
- Research progress on the offline and on-line recognition during 1980 -1990.
- After 1990, image processing technique and pattern recognition were combined using artificial intelligence.

Requirement Analysis and Feasibility Analysis

- Operational Feasibility Analysis
- Technical Feasibility Analysis
- Economic Feasibility Analysis
- Schedule Feasibility Analysis

Use Case Diagram



Methodology



Model Training

- Input nodes: 1024 input neurons
- Output nodes: 36 output nodes
- Activation Function: Sigmoid
- Number of Hidden layer: 1 hidden layer with 340 nodes
- Learning Rate: 0.075
- Training algorithm: Error Back Propagation(Mean Squared Error)

Algorithm of The System

Step 1: Start

Step 2: Train the Neural Network Model

Step 3: Load the model into the system

Step 4: Draw the character into the canvas

Step 5: Save the Drawn character into the .png(image) format

Step 6: Input the saved image into the system

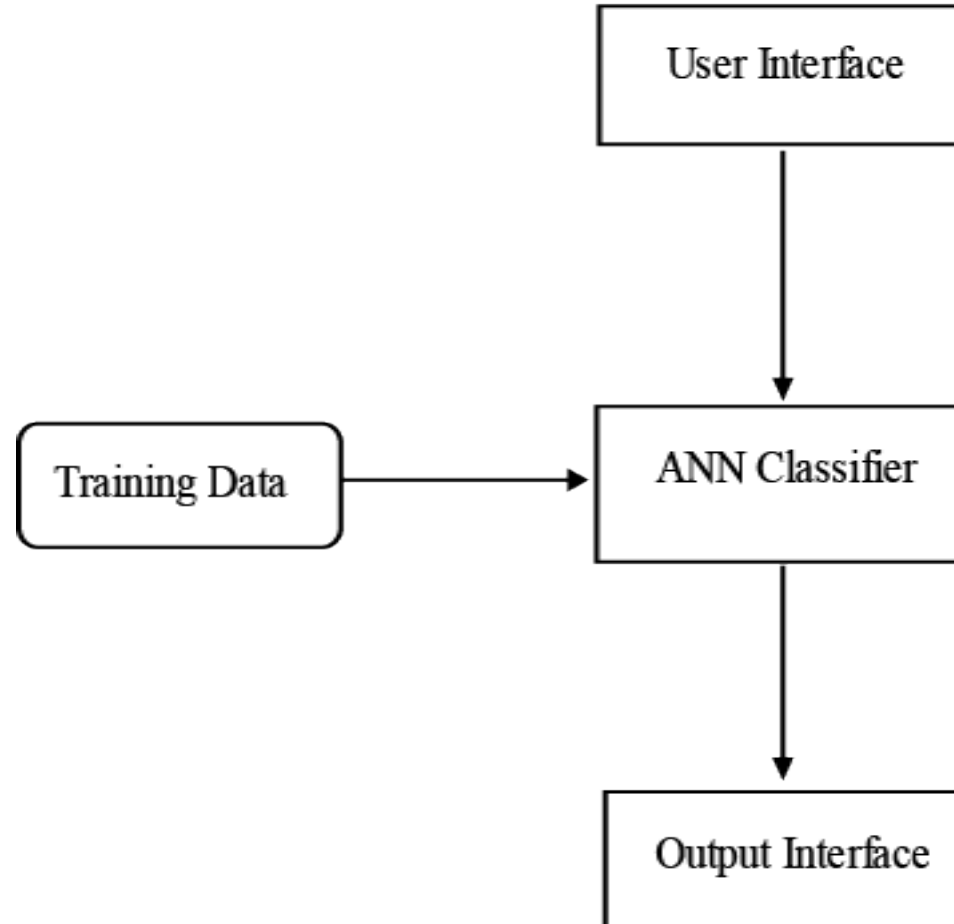
Step 7: Feed the character to the Neural Network Model

Step 8: Display Output

Step 9: Stop

Structuring System Requirements

System Architecture



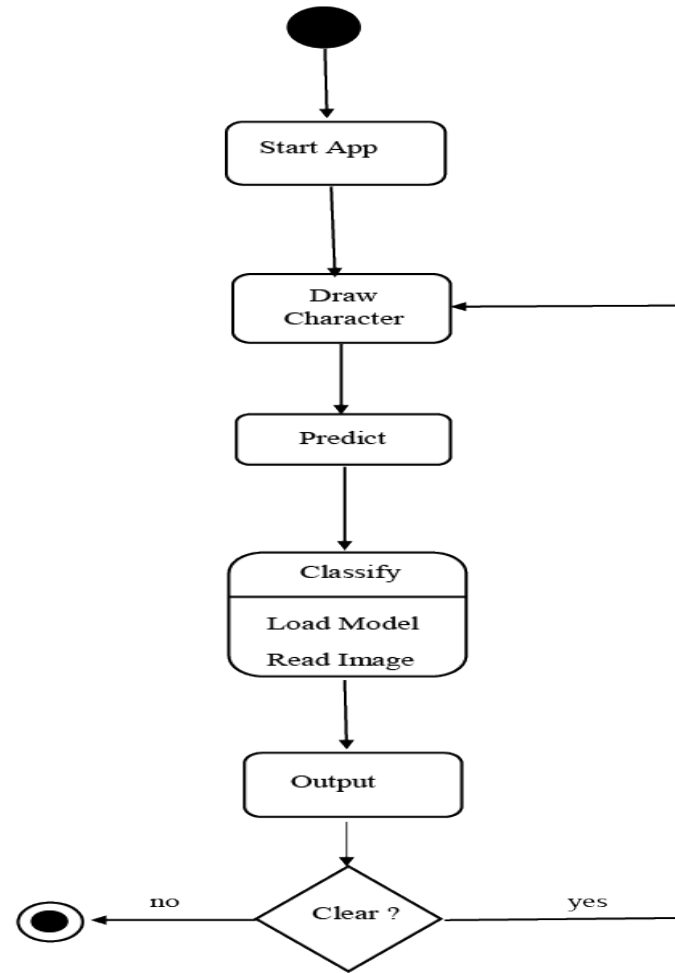
Data Flow Diagram



Class Diagram



Activity Diagram



Sequence Diagram



Testing

Unit Testing

| S.no. | Test Case | Input | Expected Result | Test Result |
|-------|----------------------------|--------------------|-------------------------------------|-----------------------------------|
| 1. | Drawing Canvas | Character | Character Drawn | User Can draw on canvas |
| 2. | Clear Button | Press clear button | Canvas clear | Canvas is clear |
| 3. | Prediction label | Draw character | Drawn character should be predicted | The drawn character was predicted |
| 4. | Preprocess data | Image Dataset | Reshape Dataset into 32*32 Image | Dataset was reshaped successfully |
| 5. | Training Model | Training Dataset | Model Should be trained. | The model was trained. |
| 6. | Recognize Nepali Character | Test Data | Predict the drawn character. | The character was predicted. |
| 7. | Model Performance | Nepali character | Predicted in a second. | Got result in a second. |

Integration Testing

| S.no. | Test Cases | Input | Expected Result | Test Result |
|-------|------------|----------------------------------|------------------------------|--|
| 1. | Load Model | Input test and training datasets | Model must load successfully | The model was loaded |
| 2. | Prediction | Draw Character | Predict the drawn character | The character was predicted successfully |

System Testing

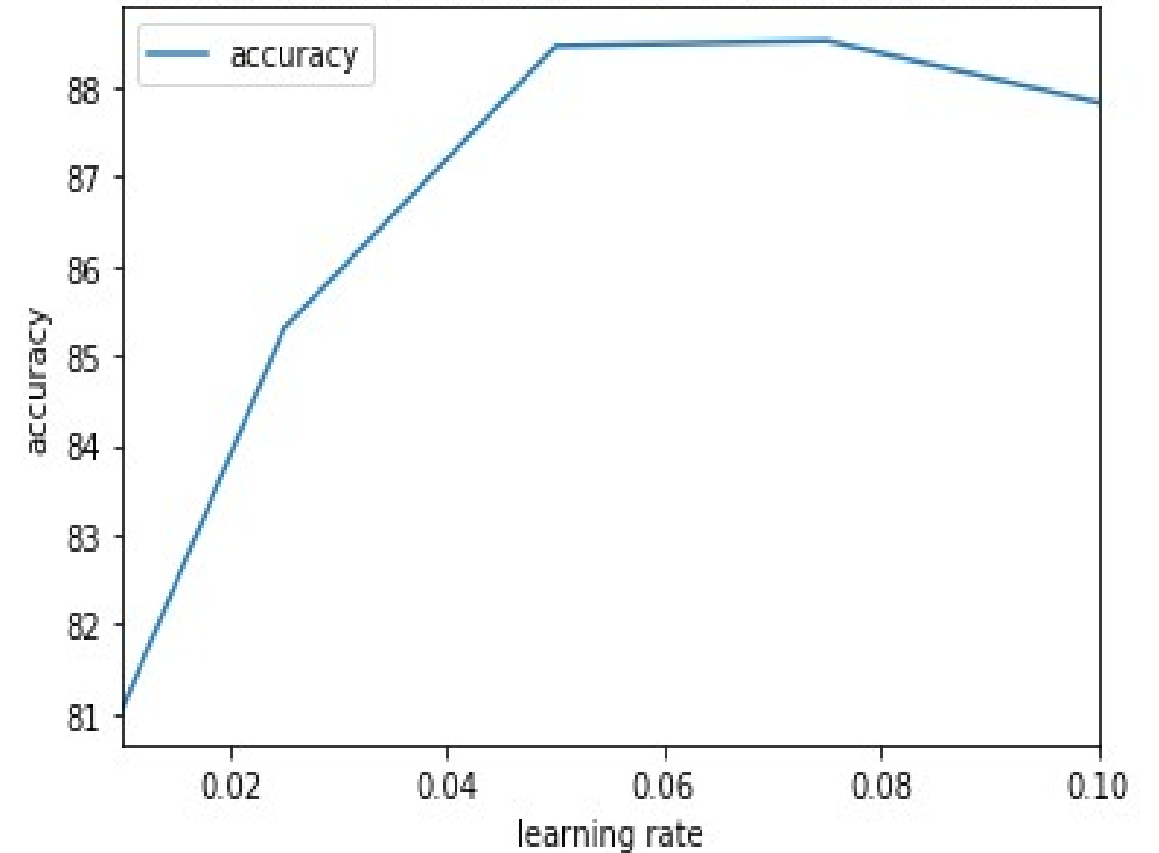
| S.no. | Test Case | Input | Expected Result | Test Result |
|-------|-------------------------|------------------------------|---|--------------|
| 1. | Predict Drawn Character | Draw Character on the canvas | Prediction Label filled with the drawn character. | As expected. |

Analysis

Accuracy Graph According to epochs



Accuracy Graph based on learning rate at 5 epochs





Output



Time Frame



References

- [1] S. Acharya, "Deep Learning Based Large Scale Handwritten Devnagari Character Recognition," Kathmandu, 2015.
- [2] Pranjali Pohankar, Namrata Taralkar, Snehalate Karmare, Smita Kulkarni, "Character Recognition using Artificial Neural Network," April, 2014.
- [3] Anna Tigunova, "Detection of Textual Information Using Convolutional Neural Networks," November 10, 2017.
- [4] Azmi Can Ozgen, "Text Detection in Natural and Computer-Generated Images," Istanbul, Turkey, 2011.
- [5] "Skymind," [Online]. Available: <https://skymind.ai/wiki/backpropagation>.
- [6] "MachineLearningMastery,"[Online].Available: <https://machinelearningmastery.com/learning-rate-for-deep-learning-neuralnetworks/>.
- [7] Bal Krishna Bal, Rajesh Pandey, Samir Tuladhar, Shanti Shakya, "INTERIM REPORT ON NEPALI OCR," Kathmandu, Nepal.
- [8] S. Shakya, A. Basnet, S. Sharma, A. Gurung, "Optical Character Recognition for Nepali, English Character and Simple Sketch Using Neural Network," Kathmandu, Nepal, 2016.
- [9] Ahmed Muaz, "Urdu Optical Character Recognition System," Lahore, Pakistan .
- [10] Chris J Maddison, Aja Huang, Ilya Sutskever, David Silver, "Move evaluation in go using deep convolutional neural networks," 2014.
- [11] Vasu Negi, Suman Mann, Vivek Chauhan, "Devanagari Character Recognition Using Artificial Neural Network".

Thank You