Sudoku Solver

Ragul VR, Niraj Gupta, Aravindakumar VM

University of Colorado Boulder

Problem Space

Sudoku is a popular number puzzle which has different variants with respect to grid size.For this project, we are considering the grid size of 9x9. Some cells are pre-filled with numbers from 1 to 9 and others are left blank. A player is supposed to fill the blank cells such that each row, each column and each sub-grid of size 3x3 have all the numbers from 1 to 9. This project is focused on building an intelligent system which, when given a Sudoku image, detects the digits and uses previously acquired knowledge and tries to solve it.

	4		2		1		6	
9		5				3		7
5		7		8		1		4
	1						9	
		1				6		
			7		5			
6		8	9		4	5		3

Figure 1:Sudoku Puzzle

Data

We are using three datasets to solve the Sudoku problem.

- **Sudoku Image** : Digital image of a Sudoku puzzle
- **MNIST Data** : Handwritten image dataset for numbers.
- **Solved Sudoku**: 1 Million solved Sudoku problems.

Approach

We are using a CNN model to detect the digits from the Sudoku image puzzle.

- Pre-Processing Image
- Cell Extraction
- Digit Recognizer Model Training
- Cell Prediction

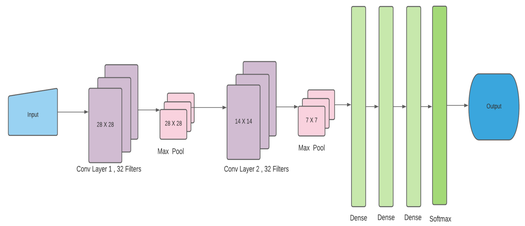

Figure 2:Digit Recognizer model

We are using three Neural Network models to solve the Sudoku problem.

- 3-Layer CNN (1 million dataset)
- 5-Layer CNN (2 million dataset)
- 15-layer CNN (1 million dataset)

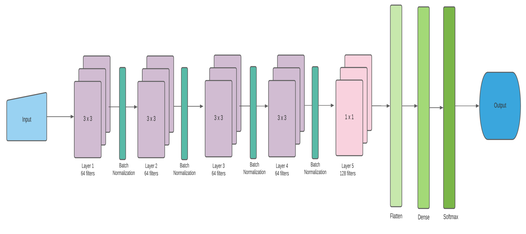

Figure 3:Five Layer CNN Model to solve the Sudoku problem

Results

From the below graph, we can observe that the model with 15 layer architecture takes the most time for predicting 100 Sudoku samples and the backtrack approach is the fastest. The time taken by the neural network models are greater because it is solving one cell at a time rather than solving entire Sudoku board.

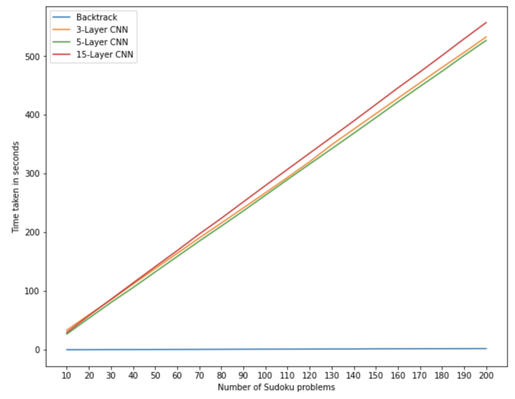

Figure 4:Time taken by various models

From the graph below, we can observe that the 5-layer model trained on Two Million dataset gives 100% accuracy whereas 3-layer model trained on one million dataset gives an accuracy of about 94% and 15-layer model trained on one million dataset yields an accuracy of 91%.

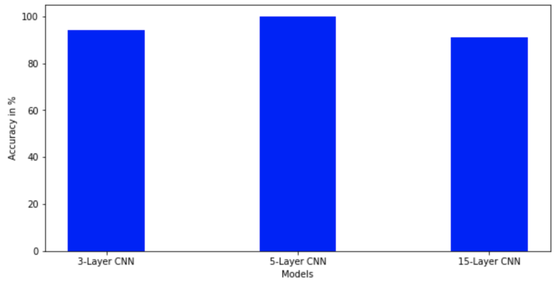

Figure 5:Accuracy of various models

Discussion

- One of the assumptions made while considering the image processing of Sudoku was that the puzzle is biggest element in the image. If the image has elements bigger than the Sudoku then detection of the contour would fail. Some of the other issues when doing this were:
  - The inner grid lines are very fine and were tough to recognize.
  - The lines are not perfect. if the lines have gaps then we would have to close it to detect contour.
- The resolution of the image can also change the cell prediction accuracy. From the results, we noticed that higher resolution images required network trained on more epochs whereas lower resolution images required less. (431x431 image had 100% accuracy on model trained on 10 epochs and 1525x1525 had 100% on model trained on 15 epochs).
- One of the first approach was to solve the entire Sudoku problem using Neural Network models at once. The input to the network was 1x81 array of the entire Sudoku. The results were unexpected as the model changed the pre-filled cells in Sudoku and accuracy took a massive hit. To improve the accuracy, we are currently using neural network models to solve cell by cell at a time rather than solving the whole puzzle at once. This improved the accuracy significantly but the time taken to solve the puzzle is also high. In future, we would like to explore how to reduce the time taken for solving a puzzle entirely at once without impacting the accuracy.

Code

Implementation details can be found here