Ernest Vovk

3697928

CS 4795

Final Paper for Sudoku Solver using Deep Learning

December 8, 2024

Introduction

– Why did you choose an idea to implement

I chose this idea as I thought I would be able to solve it or really come close to solving it by using Deep Learning because I was interested in this approach and the idea, but not everything goes as planned.

– What did you learn from it

Main lesson that I learnt is that this kind of a problem is not solved if it is not 100% correct. The approach that I chose only came close as 84% with starting at 81%. Therefore, my approach to this project failed and I should have done a different approach using the A* search algorithm as it is a better fit for this problem.

The problem

– Explain the problem to a 7th-grade student

I have tried implementing the code that would solve sudoku in a way that it would train itself on the sudokus and its corresponding solutions given in the excel file. Basically, the idea of my code so that it trains itself by going through a given dataset multiple times, which would help the code to increase the percentage of solving like problems in the future with the aim of total correctness, but it didn't quite work out.

Key idea

– Explain the idea to your friend

The core idea is to feed the prompted sudokus and their solutions to the program. Then the program goes through the whole dataset multiple times, in other words the program trains itself to boost its predictions of solving the sudoku's given faster in the future.

The details

– What is the overall performance

The overall performance of the code is that it uses CNN to solve sudoku puzzles. The model was trained on the dataset of sudoku puzzles and their solutions and evaluated on a separate test dataset. The performance of the model was based on its accuracy and loss during training and validation.

– What were the most striking successes?

The most striking success was that after running 5 epochs the code's accuracy went up by slightly more than 3%.

– What were the most striking failures? (show pictures if possible)

The most striking failure is that the problem like this we cannot have 86% accuracy, so it would actually predict the sudoku puzzles wrong even after the training was done. (The examples of that going to be included in the notebook from Colab)

Related works

– What other ideas are there and the reason why you didn't choose them.

First idea is Backtracking. I didn't choose it as it was going to be something special like project because it's straight brute force approach and it wouldn't count.

Second idea is using A* algorithm. Only after completing the whole program model of Deep Learning I realized that it wasn't the right approach, and A* would have been the approach that would result in correct output because it uses the step by step approach by predicting one number by one. Simply didn't choose it in the beginning because I thought the model that I would train would be a Rockstar (no it wasn't).

Conclusion

– Summarize your work

Although the model that I implemented and trained and provided the datasets for, didn't work out to be successful for the problem I was trying to solve for. But in the process I learned more about Deep Learning approach and what kind of problems should it be applied to and Sudoku definitely isn't one of them. It was also interesting to train the model and compare the results. I feel like I gained better understanding of this concept even though the program failed to be great.

– State your contribution (What would be useful for other students)

I'd recommend the students to actually make a better research on the problem they are about to make the project for, and not just pick the problem based on their personal preferences but rather get it confirmed with your professor first and see what they recommend to you.

– Suggest future works

I'd suggest for the future work to try implement a model that would solve sudoku puzzles with A* algorithm and even make it a CNN model but based on A* algorithm.

Bibliography

– Include the bibliography you found relevant to your project.

- Chatgpt prompts
- https://towardsdatascience.com/solving-sudoku-with-convolution-neural-network-keras-655ba4be3b11
- https://www.kaggle.com/code/yashchoudhary/deep-sudoku-solver-multiple-approaches#Deep-Learning-Approach

Appendix

– The source code

I will attach the Notebook with code and output to the submission. It will also be on my github.

– The data set

Will be attached with this file

– The failed dataset

It’s in the Notebook