Interactive Visualizer to Facilitate Game Designers to Understand Machine Learning

[1] Madhuri Bhidwe, [2] Rajiv Nagesh, [3] Kavish Harchandani, [4] Roshan Badarkhe, [5]Tannay Bagga, [6] Nitesh Chaudhari

[1]Assistant Professor, Information Technology, DY Patil College of Engineering, Akurdi, Pune.

[2,3,4,5,6]B.E., Information Technology, DY Patil College of Engineering, Akurdi, Pune.

ABSTRACT

Undergraduate research project work is very demanding of students, of tutors and of resources and many students and the transition from traditional practical work difficult. In particular, they have unrealistic expectations of what can be achieved. In order to prepare students for their project work, some third-year courses include mini-projects. This paper reports on a case study of one such mini-project: it was effective in preparing students for their project work but most students were unaware of this and as a result, many felt demoralized by their experience. A number of factors which might improve the effectiveness of mini-projects and reduce the students' negative feelings were identified including: making the aims and objectives unambiguous, achievable and explicit; recognizing the nature and difficulty of the demands which are being made of students; and providing sufficient time, support and guidance for students.

INTRODUCTION

Sudoku is one of the most popular games in the world nowadays and had been played by millions of people. This is because of its simple rules and complex problem-solving which give challenges and satisfaction for player that can complete the puzzle. Sudoku was introduced by Nikoli, Japanese Puzzle Company in 1986. It is a logic puzzle game that involves numbers and normally play with 9 x 9 grid. The sole goal for Sudoku puzzle is to write down a number from 1 to 9 in such a way that each number must appear once in every row, column and 3x3 block. The puzzle starts with a given clues or digit in various position

1.1 SUDOKU SOLVER

A. EXISTING SYSTEM (PEN AND PAPER)

People normally used pencil and eraser while playing with Sudoku puzzle. People often look at the cell and make note which numbe cannot be put into that cell. A packet act as reminder to the player and be representation that a number cannot be placed in a cell. For example, if a player determine that number 2, 5, 7 and 9 cannot be in the first cell in the first row, he/she need to put 4 symbol in the first cell.

B. PROPOSED SYSTEM (BRUTE FORCE ALGORITHM)

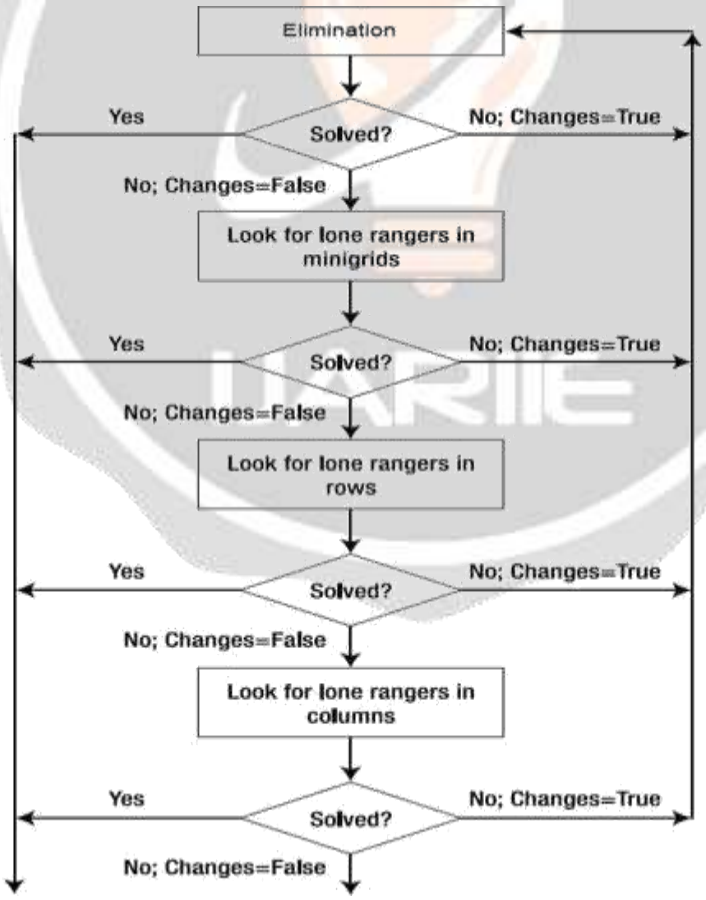
This technique is quite similar to Brute Force algorithm but the differences is Backtracking algorithm incrementally build clues for the solutions and abandon each partial clues as soon as it determine that candidate

clues is not the right solution. It will try with different number at first, if it fails, then it will backtrack and tries with another different numbers. However, this technique is much faster compared to Brute Force algorithm. It takes more than 90 minutes to find a Sudoku solution using Brute Force algorithm for three different level of difficulty which are easy, medium and hard. On the other hand, it takes 6miliseconds, 42miliseconds and 46miliseconds to solve each easy, medium and hard puzzle by using Backtracking algorithm.

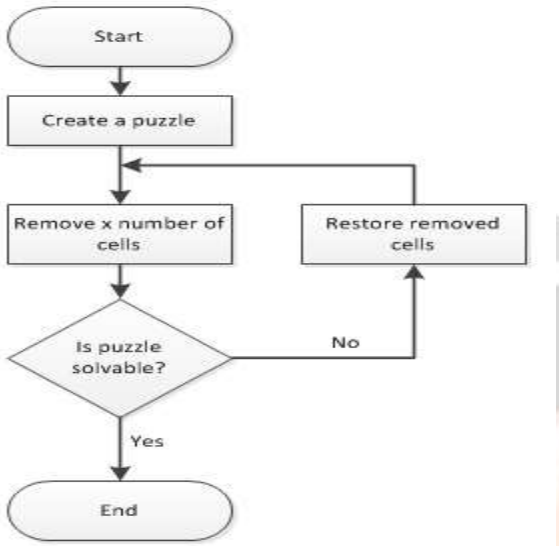
C. COMPARISION

In this section we present the result of the testing and also examine the differences between the pencil-and-paper algorithm and the brute force algorithm. The proposed algorithm has proved that is able to solve Sudoku puzzle with any levels of difficulty. We have assumed to have four levels of difficulty during testing. These levels are; easy, medium, hard and evil (challenging). This algorithm is able to solve the easy and medium puzzle without using backtracking method (less than 20 ms). In order to solve the puzzles with more difficult level such as hard or evil the algorithm uses the backtracking method as well. During testing we have noticed that the given algorithm performs better than the brute force algorithm in the term of the runtime (the time the algorithm takes to be executed). The diagram-1 below shows the differences between these two algorithms. Note that the puzzles, which are used in the testing, are taken from a valid webpage. The webpage generates Sudoku puzzles with different ratings.

1.2 SYSTEM ARCHITECTURE



1.2 Data flow diagram



TESTING STRATEGY

A. Unit Testing

Unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict, written contract that the piece of code must satisfy. Units in the proposed system are display result form, command button etc.

B. Integration testing

It takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

WEBAPP GUI

GUI means Graphical User Interface which is used to interact with the users. The primary goal of GUI is interaction. The design of GUI should be good and simple so it becomes easy for the users to interact with the system. If the architecture of GUI is convenient then it becomes easy for the users to give correct information so that the correct data is stored on the server. Indirectly this correct data helps in correct analysis and prediction. For Web application development, React Js provides a drag and drop feature so that the designing phase of GUI becomes easy and fast.

CONCLUSION

Through generating this Sudoku solver and generator I feel I have improved my programming ability. This was perhaps the largest program in terms of time invested and lines of code written that I have created.This project presents an overview of how designers structure their exploration of RL concepts in an interactive visualization and the challenges they face when exploring these concepts. We evaluate how designers approached design elements common in an interactive visualization. This study is limited by its focus on Q-Learning, a particular type of ML, and in a sudoku game design system. It is also limited in the number of participants and their domain expertise. In this study, we did not analyze the differences in behavior and needs between designers with different domain expertise (e.g., animation versus game design). However, our findings highlight the general behavior and needs of digital designers exploring an interactive visualization for ML concepts.

REFERENCES

[1] Golomb, S. and Baumert, L., Backtrack programming. J. ACM, 12 (1965) 516-52 Available: .

https://dl.acm.org/doi/10.1145/321296.321300

[2] https://www.gamespot.com/reviews/brain-age-review/1900-6147704/

[3] https://curlie.org/Games/Puzzles/Brain_Teasers/Sudoku

[4] Lewis, R., Metaheuristics can solve sudoku puzzles, Journal of Heuristics, Vol. 13,Issue 4 (2007) 387401
https://www.researchgate.net/publication/220403361_Metaheuristics_can_solve_Sudoku_puzzles