IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received 16 July 2024, accepted 23 July 2024, date of publication 29 July 2024, date of current version 6 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3434632

RESEARCH ARTICLE

A Dataset of Sudoku Puzzles With Difficulty Metrics Experienced by Human Players

SHENG-WEI WANG, (Member, IEEE)
Department of Electronic Engineering, National United University, Miaoli 360302, Taiwan
e-mail: swwang@nuu.edu.tw

This work was supported by National United University, Miaoli, Taiwan, under Grant 113-NUUPRJ-02.

ABSTRACT Sudoku is a popular logic-based puzzle game. Difficulty rating of a Sudoku puzzle is a difficult problem and a large number of research works have been proposed to tackle the problem. How to evaluate a difficulty rating algorithm is also a challenging problem. Different existing algorithms evaluate their effectiveness by using different datasets. Furthermore, some datasets are not based on the records played by human players. Therefore, the algorithms cannot be compared by using a standard dataset. In this paper, we construct a dataset of Sudoku puzzles with difficulty metrics experienced by human players. We first developed an Android application named Cloud Sudoku to collect the playing records. The development of Cloud Sudoku application has been briefly described in our previous work. This paper describes more details about database structures and features to collect the data. Two difficulty metrics are then proposed for each Sudoku puzzle based on the duration spent and the ratio which is successfully solved. A dataset of Sudoku puzzles with their difficulty metrics is designed and constructed. The difficulty metrics are shown to be close to the records played by human players. Therefore, the proposed dataset can be used to evaluate the difficulty rating algorithms. The dataset is now available on Github platform.

INDEX TERMS Dataset, difficulty metrics, difficulty rating, human players, Sudoku.

I. INTRODUCTION

Sudoku is a popular logic-based puzzle game [1]. A Sudoku puzzle is a 9×9 grid with some given hints as shown in Fig. 1(a). The player shall fill all empty cells with integer numbers ranging from 1 to 9 without any repetition in each row, column, or 3×3 sub-grid. In a Sudoku puzzle, only a single solution exists according to the given hints. The unique solution of Sudoku puzzle presented in Fig. 1(a) is shown in Fig. 1(b).

The sciences behind Sudoku puzzles have attracted a large number of researchers' interests [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24]. Some research works focused on fundamental properties of Sudoku [2], [3], [4], [5], [6]. Others can be generally classified into three categories, namely, the generating algorithms [7], [8], [9], [10], the solving algorithms [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], and the difficulty rating algorithms [21], [22], [23], [24].

In this paper, we considered the difficulty rating algorithms. Accurate difficulty rating algorithms can be employed for further applications; for example, to classify the difficulty level of a new puzzle, to predict the player's playing time, and to recommend next game for players in game systems or applications. Existing difficulty rating algorithms use different techniques to rate the difficulty level of a Sudoku puzzle. Some of them use computational models [21], [22] and others rate the difficulty from human perspectives [23], [24]. However, how to evaluate these algorithms is a challenging problem no matter how the algorithms rate difficulty level of a Sudoku puzzle,. To evaluate their accuracies, these difficulty rating algorithms face the following problems:

- Different difficulty rating algorithms employ different datasets to show their accuracies. It is difficult to compare with other algorithms since there is no standard dataset for these difficulty rating algorithms.

The associate editor coordinating the review of this manuscript and approving it for publication was Shadi Alawneh.

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

(a) The Sudoku puzzle

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

(b) The unique solution

FIGURE 1. **A Sudoku puzzle and its unique solution.**

- Some of the datasets used by the algorithms are constructed by computational models rather than those played by human players. Note that a Sudoku can be said **difficult** only if the human players feel that the Sudoku is difficult. Therefore, the dataset shall be constructed by the records played by human players.
- Although some of the datasets employed by the algorithms are constructed based on the human playing records, the dataset is not dedicated for difficulty rating algorithms. When a human player attempts to solve a Sudoku puzzle, for example, he may be interrupted and resumes to play the game after a while. The time spent in solving the Sudoku puzzle is not accurate. Such records shall be marked as **invalid** records since these records are not suitable to be included into the dataset.
- Finally, the difficulties calculated by most of the algorithms are real numbers while the difficulty levels of a Sudoku in the dataset are labeld as discrete difficulty levels such as "Easy", "Medium", "Hard", and etc. How to map the real numbers to the difficulty levels becomes another problem.

In order to accurately evaluate the difficulty rating algorithm, a dataset closed to real world is desirable. In this paper, we construct a dataset of Sudoku puzzles with difficulty metrics experienced by human players. To collect the records, we developed an Android mobile application name Cloud Sudoku to collect the records which is designed to collect the playing records of human players [25], [26]. A preliminary version which briefly introduced Cloud Sudoku application development has been accepted in [26]. In this paper, we described more details about the database structures and other features which are designed for constructing the dataset for difficulty rating algorithms.

Two difficulty metrics are proposed based on the records collected from Cloud Sudoku application. Main idea of proposed difficulty metrics is that if the playing time is longer than average playing time of a player, he may think that the Sudoku puzzle is difficult. Besides, since some human players may quit a game if he feels the puzzle is difficulty and cannot solve the puzzle, we also take the completion rate of a Sudoku puzzle into consideration.

The effectiveness of dataset is validated by calculating the correlation coefficients [27] between the difficulty metrics and the real playing records of each puzzle. Until 2024-04-15, we have collected 29,020 records in which 26,448 records are valid. There are 344 Sudoku puzzles in the proposed dataset. From 63 valid players, the average correlation coefficient is 0.7663.

Main contributions of this paper are as follow.

- An Android mobile application named Cloud Sudoku is developed to collect the records played by human players.
- A dataset of Sudoku puzzles with two difficulty metrics is proposed for future difficulty rating algorithms.
- The dataset is shown to perform close to real-world records.

The rest of this paper is organized as follows. Related research works of Sudoku are studied in the next section. The Android mobile application is introduced in Section III-A. The dataset with two difficulty metrics is proposed in Section III-B. Section IV describes the current status of dataset and evaluates its effectiveness. Finally, some concluding remarks are given.

II. RELATED WORKS

A large number of research works have been proposed to study the sciences of Sudoku [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24]. In [2], [3], [4], [5], and [6], some fundamental properties of Sudoku puzzles and solutions are investigated. A number of Sudoku generating algorithms are proposed in [7], [8], [9], and [10]. Solving algorithms for a Sudoku problem are introduced in [11], [12], [13], [14], [15], [16], [17], [18], [19], and [20]. Difficulty rating algorithms are also a hot research topic in Sudoku research [21], [22], [23], [24]. Finally, we study how to construct a dataset for evaluating a proposed algorithm from previous works [28], [29], [30], [31], [32], [33], [34].

A. RELATED WORKS ON SUDOKU

We first investigate some fundamental properties of Sudoku. Generating a valid Sudoku puzzle and solving techniques are then also studied. Finally, difficulty rating algorithms are introduced.

1) FUNDAMENTALS OF SUDOKU

Some fundamental research works have been published to study the mathematics behind the Sudoku [2], [3]. It shows that there are 6.671×10^{21} different Sudoku puzzles. In [4], the authors show that the minimum number of hints in a Sudoku puzzle with unique solution is 17. The Sudoku problem can be reduced to an SAT problem and can be proved its NP-completeness [5]. An efficient algorithmic 3-SAT formulation is proposed in [6].

2) SUDOKU GENERATING ALGORITHMS

Generation of a Sudoku puzzle with unique solution is one of the most popular research topics. The generating algorithms can be classified into two categories, namely, with or without given difficulty level as an input.

In [7], the human solving techniques are given a difficulty weight and the Sudoku puzzle generation are based on which techniques are employed to solve the puzzle. In [8], the authors proposed a generating algorithm in which the difficulty levels are considered as the number of rounds to remove a cell. In the above algorithms generating a Sudoku puzzle, the real numbers representing difficulties mapping from discrete difficulty levels are given as an input.

The other category of Sudoku generating algorithms are those without given difficulty levels in advance. In [9], Sudoku puzzles are generated by genetic algorithms. A generating algorithm which guarantees unique solution itself without relying other test function is proposed in [10].

3) SUDOKU SOLVING ALGORITHMS

Solving a Sudoku puzzle has been proved as an NP-complete problem. There are a large number of research works focusing on solving Sudoku puzzles. In [11], the problem is formulated as a constraint programming algorithm and solve the problem by programming techniques. Evolutionary algorithms are widely used to solve the Sudoku puzzle in [12], [13], [14], [15], and [16]. Furthermore, deep learning techniques [17], sinkhorn [18], search technique [19] and node based coincidence algorithm [20] are also employed to solve the Sudoku puzzle.

4) SUDOKU DIFFICULTY RATING ALGORITHMS

Difficulty rating of a Sudoku puzzle is a difficult problem. Computational models are employed to rate the difficulty of a Sudoku puzzle in [21] and [22]. In [21], a computational model of solving activities is proposed. In [22], the Sudoku solving problem is formulated as a graph colorability(GCO) problem. A solver to the GCO problem is used and the difficulty to solve the GCO problem is rated. The difficulty to solve the GCO problem is then mapped to the difficulty to solve the Sudoku problem. The authors in [23] and [24] rate the difficulty in a human perspective. Each human solving technique is given a weight value. Difficulty level of a Sudoku puzzle is the sum of weights of the techniques which are employed in solving the Sudoku puzzle.

B. DATASET CONSTRUCTION

In evaluating a proposed algorithm, a dataset close to real world is required. There are a large number of datasets proposed for evaluating different algorithms [28], [29], [30], [31], [32], [33], [34]. In these datasets, data are first collected by different tools such as sensors or mobile apps. The abnormal data are then filtered and finally the datasets are constructed. In this paper, we follow the procedure to build our dataset for Sudoku difficulty rating algorithms.

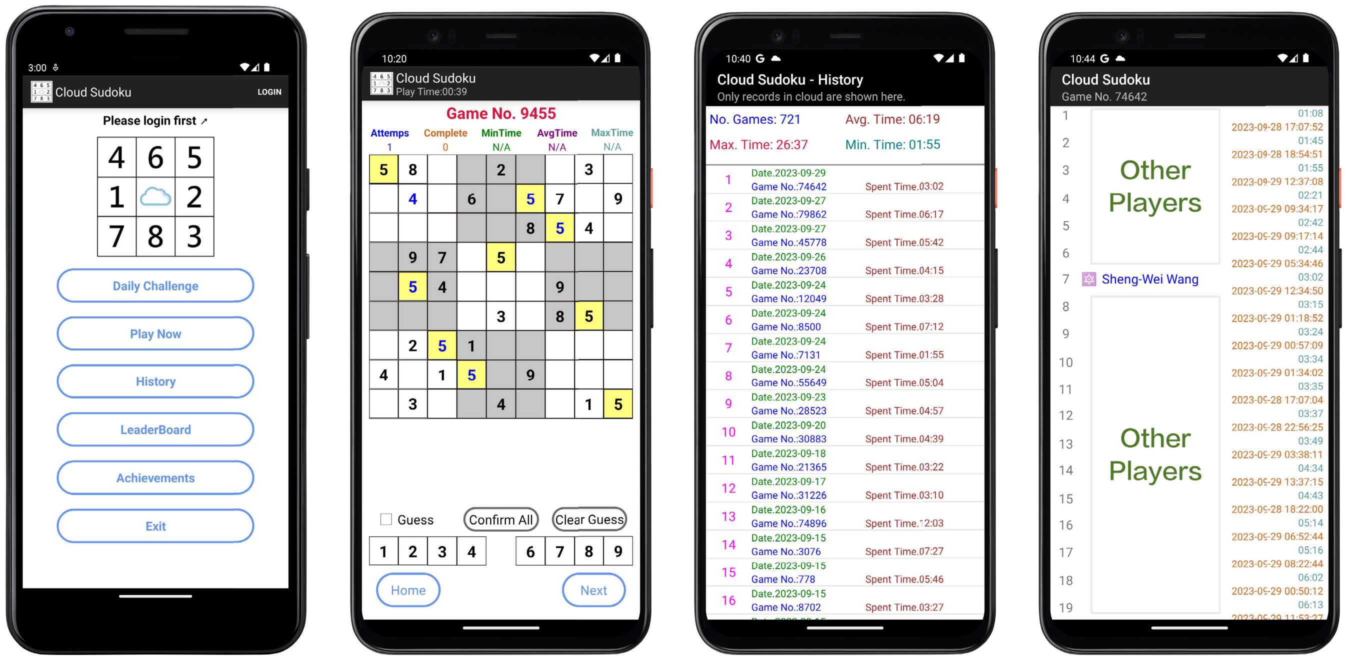
Currently, different existing algorithms employ different datasets to study the accuracies of the proposed algorithms. Therefore, evaluating the algorithms becomes a challenging problem. To solve the problems, we hope to construct a dataset of Sudoku puzzles for evaluating the difficulty rating algorithms. The dataset shall be constructed by human players. Playing time and the complete rate are the metrics to indicate the difficulty level of a Sudoku puzzle. In the dataset, difficulty metrics which are represented as real numbers are associated with a Sudoku puzzle.

III. MATERIALS AND METHODS

In constructing the dataset, we first collect the records played by human players by developing an Android mobile application. Then we proposed two difficulty metrics based on the playing time and completion rate in solving a Sudoku puzzle.

A. CLOUD SUDOKU MOBILE APPLICATION

To construct the dataset based on the playing records played by human players, an Android mobile application named Cloud Sudoku [25] is developed. A preliminary version of Cloud Sudoku application development has been accepted



(a) Main Menu (b) Playing (c) History (d) Ranking

FIGURE 2. **User interfaces of Android Cloud Sudoku mobile application.**

in [26]. In this paper, we describe more details about designs of the mobile application.

A Sudoku puzzle with a unique solution is generated by a positive integer, which is defined as the game number of this puzzle. When a player starts to play a game, a random game is rendered to the player. The player then begins to solve the puzzle and a timer starts. Once the puzzle is solved, the timer stops and the playing record is uploaded to the cloud database.

1) USER INTERFACES

In the Cloud Sudoku application, we have four primary user interfaces illustrated in Figure 2. Main menu of Cloud Sudoku is shown in Figure 2(a). We have six buttons in the main menu and one of which is to exit the app. The buttons "Daily Challenge" and "Play Now" are link to the playing interface shown in Figure 2(b). The difference is that the "Daily Challenge" provides a specific puzzle for all players to compete in the day they played. The "History" and "LeaderBoard" buttons show the playing history and ranking of a player respectively.

In Figure 2(b), the playing interface is shown for player. To fill a number in a cell, the user first selects a number and then clicks on the empty cell where they wish to place it. If the selected number already exists in the same row, column, or 3×3 sub-grid, it cannot be placed in the cell. The number of attempts and completion records are shown on the screen. Also, the minimum, maximum, and average playing time for this Sudoku puzzle are provided for the player. When all empty cells are filled without any conflicts, the puzzle is successfully solved and the record is sent to the cloud database.

The player's gaming records, including corresponding timestamps and time spent on solving puzzles, are displayed in the History interface, as shown in Figure 2(c). Finally, to create a competitive environment and encourage players to engage with more puzzles, we provide a ranking page showcasing all players, as depicted in Figure 2(d). For privacy reasons, the names of other players are covered in this paper.

2) SUDOKU GENERATING ALGORITHM

Generating a Sudoku puzzle with a unique solution is important in the mobile application. Throughout the generating algorithm, we use a random number generator with initial seed, which is defined as the game number of this Sudoku puzzle. Different from other Sudoku generating algorithms, we do not employ the difficulty level as an input to generate the Sudoku puzzle. Instead, we use exhaustive search to remove as many cells as possible such that the uniqueness of solution is maintained. The Sudoku generating algorithm used in Cloud Sudoku application is modified by the generator proposed in previous research works [2], [10]. Details of the Sudoku generating algorithm used in the Cloud Sudoku application are as follows.

Step 1 Given Sudoku game number *sudokuId* as seed for random number generator, generate a valid solution in which there is no repetition of numbers in each column, row, and 3×3 sub-grid.

Step 2 Mark all unremoved cells as *removable*.

Step 3 Randomly remove one removable cell and check if the solution is unique.

- If **YES**, go to **Step 2** to remove next cell.
- Otherwise, the removed cell is restored and marked as *unremovable*. Go to **Step 3** to remove another one.

Step 4 If there is no more removable cell resulting in unique solution, the Sudoku generating algorithm **STOP** and **OUTPUT** the Sudoku problem string.

3) DATABASE STRUCTURES

The data are stored in either device or cloud database systems. The local database system is implemented by SQLite database software and the cloud database system is implemented by MySQL database system. The local database is used for temporarily storing the records in case that the network connection is unreachable. Once the network connection is established, all records in local database will be uploaded to the cloud database and the local database is cleared.

Three tables are created in the database, namely, *player*, *game*, and *record*. The table *player* describes the basic information of a player with primary key *player_id*. The table *sudoku* represents the Sudoku puzzles as follows.

(sudokuId, sudokuPuzzle, sudokuSolution, sudokuAttempts)

The filed *sudokuId* is the primary key of the table. Fields *sudokuPuzzle* and *sudokuSolution* are the strings of Sudoku puzzle and its corresponding solution. The value of *sudokuAttempts* is an integer denoting the number of attempts to solve the Sudoku puzzle. When a Sudoku puzzle is provided in Cloud Sudoku, the value of *sudokuAttempts* of this game is incremented by 1.

A *record* in the database is represented by a tuple as follows.

(recordId, recordSudoku, recordPlayer, recordTime, recordCompleteTime, isValid)

The field *recordId* is the primary key. The fields *recordSudoku* and *recordPlayer* are the foreign keys associated to the specific Sudoku game and player respectively. Playing time in seconds are stored in *recordTime*. The timestamps that the puzzle is solved is stored in *recordCompleteTime*. Finally, a flag to indicate whether the record is valid for calculating the difficulty metrics is stored in *isValid* field. Default value of *isValid* field is set to *true*. If the record meets some conditions that make the record not appropriate to be considered in the dataset, the value of to *isValid* field is set to *false*. The conditions for invalid records are described in the following subsection.

4) FEATURES OF CLOUD SUDOKU MOBILE APPLICATION

In order to construct a dataset for evaluating difficulty rating algorithms, the Cloud Sudoku application have the following features.

- Since the playing time is the key component in deciding the difficulty level a Sudoku puzzle, accurately measuring the time spent in solving a Sudoku puzzle is very important. A timer is used to count the interval of consecutive actions in solving a Sudoku puzzle. If there exits an interval longer than 1 minute, the record is marked as invalid.
- First 20 records which are played by a new player will be marked as invalid since the player has not been familiar with the user interfaces in the mobile applications.
- In order to construct a dataset, the number of playing records on each Sudoku puzzle shall be large enough. Therefore, when a player starts to play a game, the application will provide the player the game which has not been included in the dataset with higher probability.

With the above features, we have sufficient number of valid playing records to construct the dataset for difficulty rating algorithms. Current status of the database is given in Section IV.

B. DIFFICULTY METRICS

A Sudoku puzzle is difficult if human players think it is difficult. If a Sudoku puzzle is difficult, player may spend more time on solving the puzzle or quit without solving it successfully. Based on the idea, we proposed two the difficulty metrics to identify the difficulty level of a Sudoku puzzle.

First of all, we introduce two parameters in the proposed difficulty metrics. Note that only the valid records are considered in calculating the difficulty metrics. A player is considered as a **valid player** if he has completed sufficient number of valid records such that the playing status by him has become stable. Therefore, Th_{vp} is defined as the threshold of number of completed valid records to be a valid player.

Second, a Sudoku puzzle is considered as a **qualified puzzle** if there exists sufficient number of valid records for this puzzle. The proposed dataset contains only the qualified puzzles. Let Th_{qp} be the threshold of the number of valid records to be a qualified puzzle. By using the two parameters Th_{vp} and Th_{qp}, we proposed two difficulty metrics to indicate the difficulty level of each Sudoku puzzle.

1) D_{TO}: CONSIDERING PLAYING TIME ONLY

Different players have different capabilities in solving a Sudoku puzzle. If a player spent more time than his average playing time on a specific puzzle, he may think that the puzzle is more difficult. Otherwise, the Sudoku is easy for him. Based on this idea, we formulate the difficulty metric D_{to} as follows.

Let t_{ij} be the playing time spent by a valid player i on qualified puzzle j with valid record. Let t_i be the average playing time by a valid player i over all valid records played by him. A *difficulty indicator* d_{ij} of valid player i on qualified puzzle j is calculated as follows.

$$d_{ij} = \frac{t_{ij}}{t_j} \quad (1)$$

If d_{ij} is larger than 1, it means the Sudoku puzzle j is probably difficult for player i. On the other hand, if d_{ij} is very small, Sudoku puzzle j is easy for player i.

Denote P_j as the set of valid players who complete qualified puzzle j validly. The difficulty metric $D_{to}(j)$ which considers only the playing time of qualified puzzle j can be calculated as follows where $|P_j|$ is the number of valid players in P_j.

$$D_{to}(j) = \frac{\sum_{i \in P_j} d_{ij}}{|P_j|} \quad (2)$$

Large value of $D_{to}(j)$ means Sudoku puzzle j is generally difficult. If the value of $D_{to}(j)$ is small, the Sudoku puzzle j is easy for most players.

2) D_{TR}: CONSIDERING PLAYING TIME AND COMPLETION RATE

In addition to the playing time of a Sudoku puzzle, some players may quit the game if he cannot complete the puzzle. Therefore, the rate of completion shall be considered jointly. Let C_j be the completion rate of playing records over all attempts to complete the puzzle. Generally, higher completion rate implies that the puzzle is easier. Therefore, we formulate the difficulty metric $D_{tr}(j)$ which considers both playing time and completion rate of Sudoku puzzle j as follows.

$$D_{tr}(j) = D_{to}(j) \times \frac{1}{C_j} \quad (3)$$

Same as difficulty metric D_{to}, large value of D_{tr} means that the Sudoku puzzle is difficult. The Sudoku puzzle is easy if the value of D_{tr} is small.

IV. CURRENT STATUS AND EVALUATIONS OF THE PROPOSED DATASET

In this section, we first describe the current status of the Cloud Sudoku application system. Next, we evaluate the effectiveness of the proposed difficulty metrics. Since other Sudoku datasets used in the Sudoku difficulty rating algorithms [21], [22], [23], [24] are not public accessible, it is difficult for us to compare the effectiveness for other datasets in this paper.

A. CURRENT STATUS OF CLOUD SUDOKU SYSTEM

The Cloud Sudoku application has been released to Google Play Store since 2023-05-01 [25]. Until 2024-04-15, the number of players who have successfully solved at least one game is 147. The number of Sudoku games which are successfully solved by at least one player is 5,045. Finally, the number of valid records is 24,668. By selecting appropriate values of $Th_{vp} = 50$ and $Th_{qp} = 20$, the numbers of valid records and valid players are 24,668 and 63 respectively. In the proposed dataset, there are 344 qualified Sudoku puzzles with difficulty metrics.

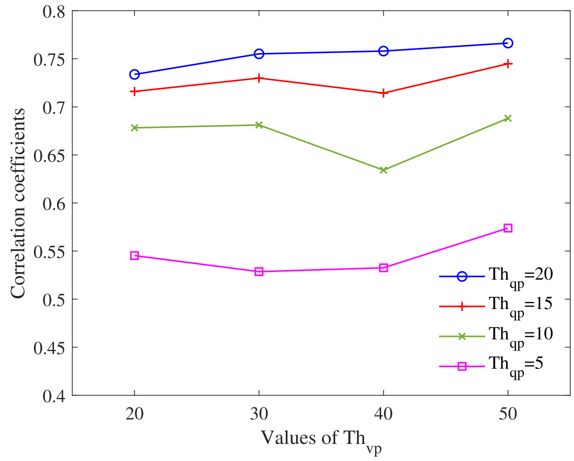


FIGURE 3. Correlation coefficients between difficulty metric D_{to} and valid records played by valid players.

Next, we evaluate effectiveness of the proposed difficulty metrics by calculating the correlation coefficients [27] between the difficulty metrics and the real playing records. For each player, we extract the qualified puzzles that he has completed validly. Three vectors are generated. Two vectors V_{to} and V_{tr} represent the two difficulty metrics D_{to} and D_{tr} of each valid Sudoku puzzle respectively. The third vector V_{pt} is composed of the difficulty indicators calculated from equation (1) by each player. The two correlation coefficients are calculated between (V_{to}, V_{pt}) and (V_{tr}, V_{pt}) for a player.

The correlation coefficients vary frequently. When a valid record is added into the database, the difficulty metrics are updated and the correlation coefficients shall be re-calculated. When the numbers of valid player, valid records, and qualified puzzles are large enough, the difficulty metrics and correlation coefficients become more stable. The results shown in this paper is based on the playing records captured on 2024-04-15.

B. EFFECTIVENESS OF THE PROPOSED DIFFICULTY METRICS

Note that we have two parameters Th_{vp} and Th_{qp} denoting the threshold values for valid players and qualified puzzles. We first study the effects of the two parameters. Figs. 3 and 4 show the correlation coefficients between two difficulty metrics D_{to}, D_{tr} and the real playing records respectively. The values of Th_{vp} and Th_{qp} range from 20 to 50 and from 5 to 20 respectively. From the figures, we can make the following observations:

- All the correlation coefficients in the two figures are larger than 0.5. The maximum and minimum values of the correlation coefficients between D_{to} and real playing records are 0.7663 and 0.5286 respectively. The maximum and minimum values of the correlation coefficients between D_{tr} and real playing records are 0.6733 and 0.5058 respectively. The results indicate that

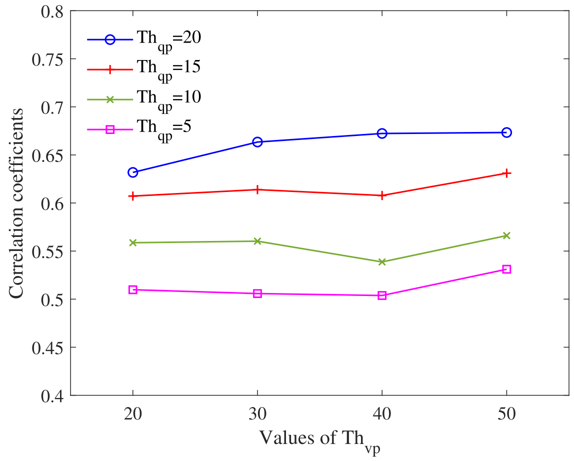


FIGURE 4. Correlation coefficients between difficulty metric D_{tr} and valid records played by valid players.

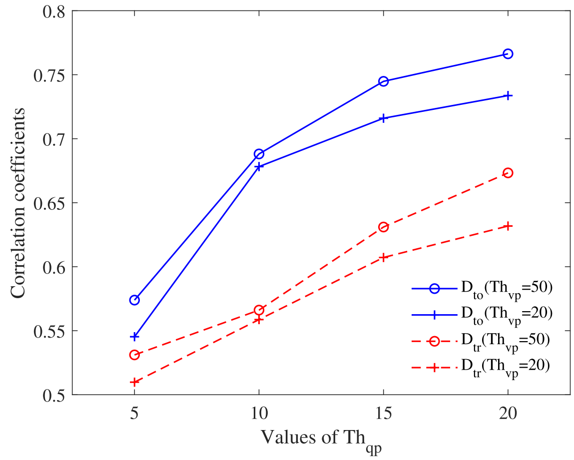


FIGURE 5. Comparison of two difficulty metrics D_{to} and D_{tr} with Th_{vp} = 20 and 50.

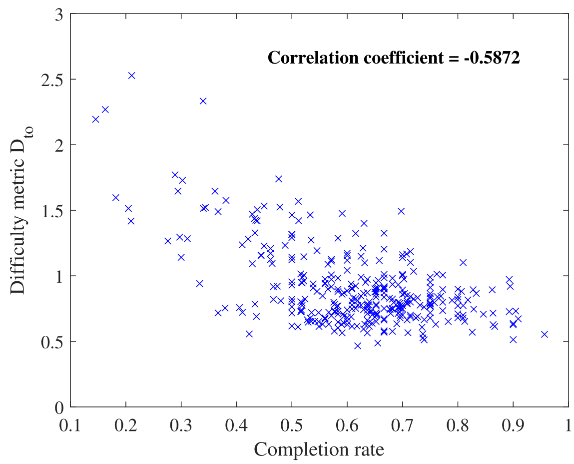


FIGURE 6. Relationships between difficulty metric D_{to} and completion rate.

the proposed difficulty metrics are moderately positively correlated to the real playing records. The dataset is shown to be closely correlated to the real playing records. Therefore, the dataset can be employed for evaluating the difficulty rating algorithms.

- The correlation coefficients do not necessarily increase when the value of Th_{vp} increases. Note that the value of Th_{vp} is designed to obtain a more stable value of average playing time. The results show that $Th_{vp} = 20$ is enough and $Th_{vp} = 50$ is better for calculating the difficulty metrics. That is, a player is familiar with the user interfaces when he completed 20 puzzles validly. If we need more stable dataset, $Th_{vp} = 50$ is better.
- The correlation coefficients increase when increasing values of Th_{qp}. Large value of Th_{qp} implies that a puzzle requires a large number of valid records to be qualified. Larger number of valid records on a Sudoku puzzle makes the data on the Sudoku puzzle more stable. Therefore, the difficulty metrics are closer to the difficulties experienced by human players.

We next compare the difficulty metrics D_{to} and D_{tr}. Fig. 5 shows the correlation coefficients between difficulty metrics and real playing records when Th_{vp} is 20 and 50. From the figure, we found that the correlation coefficients between D_{to} and the real playing records are larger than those between D_{tr} and the real playing records. However, we cannot declare that the difficulty metric D_{to} is closer to the real world than D_{tr}. This is because that the real playing records are those successfully completed by the human players. The completion rates are not considered in the real playing records while the difficulty metric D_{tr} jointly considered the completion rate. In addition, when the values of Th_{qp} increase, the difficulty metrics increase too. Therefore, we set $Th_{vp} = 50$ and $Th_{qp} = 20$ in the proposed dataset from the above observations.

Finally, we are interested in the relationship between the playing time and completion rate. It is believed that when a Sudoku puzzle is difficult, the playing time increases and the completion rate decreases. In the proposed dataset, we show that the relationship between the two metrics follows the argument. Note that the difficulty metric D_{to} considers only the playing time. For a Sudoku puzzle j, a data point $(C_j, D_{to}(j))$ is generated where C_j is the completion rate. Fig. 6 shows the 344 generated data points on an XY-plane where X- and Y-axes represent the values of the completion rate and difficulty metric D_{to} respectively. From the figure, we observe that the difficulty metric D_{to} decreases when the completion rate increases. The correlation coefficient is -0.5872 which are moderately negatively correlated to each other. The result shows that the proposed difficulty metric D_{to} follows the common sense that the completion rate is smaller if the puzzle is more difficult.

V. CONCLUSION

In this paper, we present a dataset of Sudoku puzzles with two difficulty metrics which are experienced by human players. To collect the data, an Android mobile application named Cloud Sudoku is developed. A preliminary version of developing the mobile application has been accepted

TABLE 1. Example Sudoku puzzles with difficulty metrics in the proposed dataset.

No.	Sudoku Id	Sudoku Puzzle	D_{to}	D_{tr}
1	124	..7........5.4..7..695...31...4.58.2.5..2..4.6.23.1...29...358..3..1.2........3..	0.6517	0.9992
2	200	..61..97........3....734..5....49..2.15...86.9..68....6..293....9........51..74..	1.1547	2.5980
3	388	82..7.....95.86...6...4..3...6..8...13.....46...3..1...7..3...4...75.31.....9..85	0.7601	1.0979
4	401	..29...7..1...7....9.821...1.....5.7.47.6.21.2.9.....4...418.3....3...8..3...24..	0.8460	1.3535
5	430	2.163.48....9...3.3.....6......7..4651..8..7374..9......4.....7.3...5....79.628.4	0.6132	1.1169
6	442	17.8.2.3.2.3....89......7....42.8......1.9......4.68....2......79....4.6.6.7.4.52	0.9827	1.6847
7	505	....1.3..3....714...2.34.....9....144.3...6.261....7.....69.2...275....6..6.2....	0.8412	1.1777
8	570	8.....9...6.....84..47....69..46..7..389.754..7..38..97....28..24.....5...6.....2	0.9462	1.4509
9	595	.73.....848..6.5....9.7........896..3..2.6..5..734........2.3....1.3..272.....41.	1.1606	1.6580
10	624	...6.3..5.8.....3719..5....2.5....6..61...84..3....5.1....2..9897.....5.6..5.7...	0.6787	0.9049
11	888	...4.23.1........6..7.9.2.47..64.8....85.91....4.71..24.6.5.7..9........8.52.7...	0.7812	1.3540
12	966	..6.....71..7.4.....5...13.5.28..4...9..7..2...4..67.9.41...6.....2.5..19.....3..	0.8237	1.5726
13	1135	..35.7.64.9....7.......6.93....8..518.2...3.775..6....62.3.......5....3.34.2.51..	0.7870	1.1110
14	1246	.3.5.8.....9....3..58.762...9....4.86..9.4..38.4....2...572.16..2....3.....6.5.8.	0.7713	1.3223
15	1342	....1327......26..37.....8..13....6.9..3.8..2.8....73..3.....15..16......2719....	0.6636	0.9369
16	1404	93........654..9.84.7..9......87..5...8...6...7..14......7..4.67.3..681........27	0.8640	1.1520
17	1411	.56.9.17.1...7...6...1....2..9.128....8...2....168.5..3....1...2...4...9.65.2.43.	0.6029	1.0336
18	1423	46.....599.3..5..1....8..768..2.........9.........6..524..7....6..3..1.735.....82	0.9097	1.3646
19	1472	.1..4...7...9....2.78..61.4.8...19....1...6....93...7.1.56..28.8....4...7...5..1.	0.8985	1.3477
20	1474	.9.2..5173..1.8.......5......461...5.13.4.27.7...324......9.......5.1..4561..4.2.	0.6501	1.1855

in [26]. Details of the database systems and features of the application is introduced in this paper. In Cloud Sudoku application, a Sudoku puzzle generating algorithm without given difficulty level is proposed. A flag to indicate the validity of a playing records is introduced to identify whether a record is included in calculating the difficulty metrics or not.

After collecting the records, two difficulty metrics are introduced. The difficulty metric D_{to} considered the playing time to solve a puzzle only. The other one D_{tr} considered the playing time and completion rate both. The difficulty metrics are evaluated by the real playing records. The evaluation results show that the correlation coefficients between the difficulty metrics and the real playing records are both larger than 0.5 which is considered moderately positively correlated. When the number of valid records of a Sudoku puzzle is large enough, the correlation coefficient can be larger than 0.7 which means strongly positively correlated. The results show that the proposed difficulty metrics are close to the real difficulties experienced by human players. In the future, the dataset can be used to evaluate the difficulty rating algorithms. To access the up-to-date dataset, please see Appendix A.

Currently, the dataset is constructed from the data collected in Android mobile application. Therefore, the dataset is limited to the users who use Android phones. In the future, we will develop the iOS and web-based Sudoku systems to collect the data. The playing time in different platforms will be different and a normalization shall be applied. These improvements will be left for future works.

APPENDIX A
SOME EXAMPLE PUZZLES IN THE DATASET

In the proposed dataset, each Sudoku puzzle is associated with a game number, game string, and two difficulty metrics D_{to} and D_{tr}. Table 1 shows 15 example Sudoku puzzles in the dataset when the parameters $Th_{vp} = 50$ and $Th_{qp} = 20$. In the table, the Sudoku puzzle is represented as a string with 81 characters where every 9 characters consist of a row. If a character shows an integer from 1 to 9, this is the hint number on the cell. The dot character "." means a hidden cell which shall be determined by the player. Finally, two difficulty metrics D_{to} and D_{tr} are presented and the completion rate of a Sudoku puzzle can be calculated by D_{to}/D_{tr}. The up-to-date dataset can be obtained from https://github.com/synnwang/sudoku_dataset_difficulty under CC0-1.0 license.

REFERENCES

[1] J.-P. Delahaye, "The science behind sudoku," *Sci. Amer.*, vol. 294, no. 6, pp. 80–87, Jun. 2006.

[2] B. Felgenhauer and F. Jarvis, "Mathematics of sudoku I," *Math. Spectr.*, vol. 39, no. 1, pp. 15–22, 2006.

[3] E. Russell and F. Jarvis, "Mathematics of sudoku II," *Math. Spectr.*, vol. 39, no. 2, pp. 54–58, 2006.

[4] H.-H. Lin and I.-C. Wu, "Solving the minimum sudoku problem," in *Proc. Int. Conf. Technol. Appl. Artif. Intell.*, Nov. 2010, pp. 456–461.

[5] I. Lynce and J. Ouaknine, "Sudoku as a SAT problem," in *Proc. AI&M*, 2006, pp. 1–9.

[6] D. Rai, N. S. Chaudhari, and M. Ingle, "An efficient algorithmic 3-SAT formulation for sudoku puzzle using graph coloring," in *Proc. Int. Conf. Adv. Comput. Telecommun. (ICACAT)*, Dec. 2018, pp. 1–6.

[7] M. Hunt, C. Pong, and G. Tucker, "Difficulty-driven sudoku puzzle generation," *UMAP J.*, vol. 29, no. 3, pp. 343–362, 2007.

[8] G. Li, X. Qi, N. Wang, R. Huang, and J. Zhao, "Genarate and solve sudoku puzzle," in *Proc. 10th Int. Conf. Inf. Technol. Med. Educ. (ITME)*, Aug. 2019, pp. 671–673.

[9] T. Mantere and J. Koljonen, "Solving, rating and generating sudoku puzzles with GA," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 1382–1389.

[10] B. Sun, X. Sun, Y. Wu, Y. Yin, and G. Yang, "A new algorithm for generating unique-solution sudoku," in *Proc. 4th Int. Conf. Natural Comput.*, vol. 7, 2008, pp. 215–217.

[11] H. Simonis, "Sudoku as a constraint problem," in *Proc. CP Workshop Modelling Reformulating Constraint Satisfaction Problems*, vol. 12, 2005, pp. 13–27.

[12] H. Chel, D. Mylavarapu, and D. Sharma, "A novel multistage genetic algorithm approach for solving sudoku puzzle," in *Proc. Int. Conf. Electr., Electron., Optim. Techn. (ICEEOT)*, Mar. 2016, pp. 808–813.
[13] X. Qi, G. Li, N. Wang, X. Wang, and L. Wen, "Method study on solving sudoku problem," in *Proc. 3rd Int. Conf. Data Sci. Bus. Anal. (ICDSBA)*, Oct. 2019, pp. 269–271.
[14] N. A. Hasanah, L. Atikah, D. Herumurti, and A. A. Yunanto, "A comparative study: Ant colony optimization algorithm and backtracking algorithm for sudoku game," in *Proc. Int. Seminar Appl. Technol. Inf. Commun. (iSemantic)*, Sep. 2020, pp. 548–553.
[15] H. Lloyd and M. Amos, "Solving sudoku with ant colony optimization," *IEEE Trans. Games*, vol. 12, no. 3, pp. 302–311, Sep. 2020.
[16] G. K. Baydogmus, "Probability selection for solving sudoku with ant colony optimization algorithm," in *Proc. 1st Int. Conf. Inf. Syst. Inf. Technol. (ICISIT)*, Jul. 2022, pp. 161–165.
[17] K. S. Vamsi, S. Gangadharabhotla, and V. S. H. Sai, "A deep learning approach to solve sudoku puzzle," in *Proc. 5th Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, May 2021, pp. 1175–1179.
[18] T. K. Moon, J. H. Gunther, and J. J. Kupin, "Sinkhorn solves sudoku," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1741–1746, Apr. 2009.
[19] C. Wang, B. Sun, K.-J. Du, J.-Y. Li, Z.-H. Zhan, S.-W. Jeon, H. Wang, and J. Zhang, "A novel evolutionary algorithm with column and sub-block local search for sudoku puzzles," *IEEE Trans. Games*, vol. 16, no. 1, pp. 162–172, Mar. 2024.
[20] K. Waiyapara, W. Wattanapornprom, and P. Chongstitvatana, "Solving sudoku puzzles with node based coincidence algorithm," in *Proc. 10th Int. Joint Conf. Comput. Sci. Softw. Eng. (JCSSE)*, May 2013, pp. 11–16.
[21] R. Pelánek, "Difficulty rating of sudoku puzzles by a computational model," in *Proc. 24th Int. Florida Artif. Intell. Res. Soc. Conf.*, 2011, pp. 671–673.
[22] S. Ono, R. Miyamoto, S. Nakayama, and K. Mizuno, "Difficulty estimation of number place puzzle and its problem generation support," in *Proc. ICCAS-SICE*, Aug. 2009, pp. 4542–4547.
[23] C. Xu and W. Xu, "The model and algorithm to estimate the difficulty levels of sudoku puzzles," *J. Math. Res.*, vol. 1, no. 2, pp. 43–46, Aug. 2009.
[24] H. Wang, Y. W. Wang, and C. T. Sun, "Rating logic puzzle difficulty automatically in a human perspective," in *Proc. DiGRA Nordic Conf. Local Global–Games Culture Soc.*, 2012, pp. 1–7.
[25] S.-W. Wang. *Cloud Sudoku*. Accessed: Jul. 15, 2024. [Online]. Available: https://play.google.com/store/apps/details?id=com.yicheng.sudoku
[26] Y.-C. Lin, W.-Y. Kuo, and S.-W. Wang, "Cloud sudoku: A sudoku application to collect game records by human players," in *Proc. Int. Conf. Consum. Electron. Taiwan (ICCE-Taiwan)*, Taichung, Taiwan, Jul. 2024, pp. 1–2.
[27] P. Schober, C. Boer, and L. A. Schwarte, "Correlation coefficients: Appropriate use and interpretation," *Anesthesia Analgesia*, vol. 126, no. 5, pp. 1763–1768, 2018.
[28] L. Guo, L. Wang, C. Lin, J. Liu, B. Lu, J. Fang, Z. Liu, Z. Shan, J. Yang, and S. Guo, "Wiar: A public dataset for WiFi-based activity recognition," *IEEE Access*, vol. 7, pp. 154935–154945, 2019.
[29] G. Tong, Y. Li, D. Chen, Q. Sun, W. Cao, and G. Xiang, "CSPC-dataset: New LiDAR point cloud dataset and benchmark for large-scale scene semantic segmentation," *IEEE Access*, vol. 8, pp. 87695–87718, 2020.
[30] S. Wei, X. Zeng, Q. Qu, M. Wang, H. Su, and J. Shi, "HRSID: A high-resolution SAR images dataset for ship detection and instance segmentation," *IEEE Access*, vol. 8, pp. 120234–120254, 2020.
[31] N. Li, X. Shen, L. Sun, Z. Xiao, T. Ding, T. Li, and X. Li, "Chinese face dataset for face recognition in an uncontrolled classroom environment," *IEEE Access*, vol. 11, pp. 86963–86976, 2023.
[32] S. Wang, T. Li, X. Shen, and H. Zhao, "China university online public opinion risk dataset," *IEEE Access*, vol. 12, pp. 55225–55233, 2024.
[33] I. Skula and M. Kvet, "A framework for preparing a balanced and comprehensive phishing dataset," *IEEE Access*, vol. 12, pp. 53610–53622, 2024.
[34] L. Maman, E. Ceccaldi, N. Lehmann-Willenbrock, L. Likforman-Sulem, M. Chetouani, G. Volpe, and G. Varni, "GAME-ON: A multimodal dataset for cohesion and group analysis," *IEEE Access*, vol. 8, pp. 124185–124203, 2020.

SHENG-WEI WANG (Member, IEEE) received the B.S. degree in information management from National Taiwan University, Taipei, Taiwan, in 2001, and the Ph.D. degree in communications engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2008. After one-year of military service, he has been a Faculty Member with the Department of Applied Informatics, Fo Guang University, since 2009. He was a Faculty Member with the Department of Computer Science and Information Engineering, Tamkang University, from 2021 to 2022. Since August 2022, he has been a Faculty Member with the Department of Electronic Engineering, National United University, where he is currently an Associate Professor. His research interests include blockchain, routing and wavelength assignment (RWA) algorithms in all-optical WDM networks, peer-to-peer applications, queueing theory, and performance evaluation.