Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I,
IMECS 2013, March 13 - 15, 2013, Hong Kong

Extracting Desirable Moves from Match Logs of *SRPG* Using n-Gram Statistics

Takuya Narita, Haruhiko Sato, Satoshi Oyama, Masahito Kurihara

***Abstract*—In recent years, more and more people are playing video games. However, tutorials that support most games do not help novice players develop the knowledge leading to winning strategies. In this study, as a first step, we define such knowledge as a sequence of Desirable Moves, and we extract these moves from match logs of games. A match log is a recorded transition history of coded game states. We convert the coded states for a match to a character string, and use n-gram statistics to extract fixed-form patterns from the strings. We suggest plural encoding methods based on heuristics, and we test the methods having AIs compete against each other .**

***Index Terms*—game AI, Desirable Moves, n-gram statistics, extraction of fixed-form patterns.**

I. INTRODUCTION

IN recent years, more people than ever are playing video games. Much of the expanding markets for social and casual games are being driven by increased availability of mobile phones. But, even though online tutorials are available for many games, mastering a video game can be a considerable challenge. This is because tutorials only show how to play; they only illustrate the rules of the game. Therefore, novice players must endure considerable trial-and-error before they develop understandings that lead to winning strategies. But novices start with little knowledge about a game; therefore, they usually have little insight as to how to efficiently gain the experience they need to succeed. Often, a novice becomes frustrated and abandons a game prematurely.

Some novices try to avoid the learning process by finding, from friends or from the Internet, a few simple winning strategies. But this type of play fails to develop a player ' s skill; therefore, the player cannot develop a deep understanding of the game. Rather than becoming frustrated, these types of players become bored and abandon the game for other pursuits. To help novice players develop appropriately, game developers should provide tutorials that show novices winning strategies and illustrate why they work. The challenge is how to obtain such knowledge.

This study is a first step towards developing better tutorials for video games players. We define the sequence of moves in a winning strategy as a set of 'Desirable Moves'. As a test platform, we use Simulation Role Playing Games (SRPG) that is a principal class of consumer games and extract the Desirable Moves from match logs of SRPG. Apparently, players whose experience and skill are superior to a standard player determine a move, intentionally or unintentionally, from Desirable Moves based on their experience. Therefore, we try to extract Desirable Moves from match logs involving such players.

Manuscript received January 8, 2011; revised January 31, 2013.

T. Narita is with Graduate School of Information Science and Technology, Hokkaido University, Hokkaido, JAPAN e-mail:t-narita@complex.ist.hokudai.ac.jp

H. Sato is with Graduate School of Information Science and Technology, Hokkaido University, Hokkaido, JAPAN e-mail:haru@complex.eng.hokudai.ac.jp

S. Oyama is with Graduate School of Information Science and Technology, Hokkaido University, Hokkaido, JAPAN e-mail:oyama@ist.hokudai.ac.jp

M. Kurihara is with Graduate School of Information Science and Technology, Hokkaido University, Hokkaido, JAPAN e-mail:kurihara@ist.hokudai.ac.jp

II. HOW TO EXTRACT DESIRABLE MOVES

Desirable moves are consecutive moves that lead to a win, either over the long term or short term. There is a strong possibility that moves selected by a player whose experience and skills are superior to a standard player in order to win a match are factors leading to a win irrespective of whether the winner selected those moves intentionally. Therefore, moves appearing repeatedly in match logs of winners are identified and extracted as Desirable Moves.

A typical procedure for extracting data from the match logs of a game is the acquisition of move sequence patterns from game record database using n-gram statistics by Nakamura [1]. He considered records of a game of Go as encoded character strings of Go moves; then, he extracted move sequence patterns from a Go database. The move sequences were extracted as fixed-form patterns based on n-gram statistics, which is used in natural language processing. In this study, we encode the game state on every move from a match log, combine each coded state into a character string for the match, and extract the Desirable Moves using n-gram statistics.

A. n-Gram Statistics

An n-gram is a contiguous sequence of n characters (or words) in a text; therefore, n-gram statistics reflect how often strings (or words) of n characters (or words) appear[2]. In this study, we treats one match log as a character string, but as this string is not a natural language, morphological analysis is not available. Therefore, we merely determine the frequency with which n-character strings appear. However, there is a problem that, when sampling fragmental character strings, it is difficult to know that a finished expression has been extracted. In general, when sampling n-grams for many n, the frequency of a character string x appearing in the text $f(x)$ and the frequency of another character string y, which is a substring of x, appearing in the text$f(y)$ satisfy the relationship $f(x) < f(y)$. Therefore, it is difficult to judge fixed form characteristics of the string from only a comparison of simple appearance frequencies. Some techniques, such as entropy thresholds [3] and substring frequency profiles[1] is suggested for the solution to this problem. This study uses the normalized frequency method[4] , have been suggested for solving this problem. This study uses the normalized

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

frequency method [3] and compares the approach using simple frequencies in the acquisition of Desirable Moves from match logs.

B. Normalized Frequency Method

When two character strings x and y satisfy the relationship $f(x) = f(y)$, and if their lengths x, y satisfy $x < |y|$, then it appears that y is more important than x. This is because, as the length of a character string increases, the types of characters in the string can increase, and the appearance frequencies of long character strings are relatively small. To compensate for this, a normalized method can be used in which the appearance frequency of a character string is multiplied by a coefficient $\alpha(n)$ that depends on the length (n) of the string. Let $\beta(n)$ be the number of different n-grams that appear in the target text. Then, the normalized coefficient $\alpha(n)$ and normalized frequency $f_{normalize}(x)$, respectively, are calculated by

$$\alpha(n) = \sum_{i=1}^{n} \beta(i) \quad (1)$$

$$f_{normalize}(x) = (f(x) - 1) \cdot \alpha(|x|) \quad (2)$$

Fixed-form patterns are extracted at large orders of this normalized frequency. If the normalized frequency of a character string has high rank and its normalized frequency is larger than that of its substring, then the substring is erased. In this way, fragmentary character strings are removed as much as possible.

III. METHOD OF ENCODING STATE

Extracting Desirable Moves is affected by, how each state is encoded If we try to use all available information for each state, then the amount of information is large, the patterns in encoded states increase, and it becomes difficult to extract Desirable Moves using the appearance frequency. Therefore, it is necessary to encode the characteristics of each state using a moderately sized granularity. In addition, since our AI program will use the Desirable Moves to actually select a game move, we must encode each state using local information. Therefore, in this study, we propose and compare two encoding plans. Both two plans depend on a heuristic about playing SRPG and a heuristic about SRPG that we created to use for an experiment.

A. Encoding Plan 1

Each state is encoded as a nine-dimensional vector with the following nine elements.

- Index indicating which player has to play (turn index)
- Ratio of the recent hit point (HP) of turn unit to its maximum HP
- Manhattan distance to the nearest enemy unit
- Manhattan distance to the nearest ally unit
- Action taken
- Ratio of the recsent HP of the target unit of action to its maximum HP
- Manhattan distance to the target
- Amount of change in the distance to the nearest enemy unit
- Amount of change in the distance to the nearest ally unit

We explain each element. When a match has finished, we set the turn index of the winner to $+1$, and that of the loser to -1. When the data are used to identify the next Desirable Moves for a player, that player ' s turn index is set to $+1$ because we presume that the player intends to win the game. The HP is one of the most important parameters in SRPG; therefore, we encode it. But, in this plan, we encode the ratio of the recent HP to the maximum HP. In doing so, we round the ratio to one decimal point, and ratios between 0 and 1.0 are replaced by 0.1. To encode positions between units, we could use relative coordinates or distances. But, in reality, in an SRPG, there are many situations in which relative distances between neighboring objects are more important than relative coordinates. Therefore, in this study, we adopt distance. For the action parameter, we set a unique identification number to the action that each unit can take and include it at the time of encoding. The HP of the target unit of action and the distance to the target are encoded just as described above. But, when an object is not a unit, we set HP to 0. Finally, we consider the importance of the amount of change in distance. When we consider only the distance as a measure of the movement of a unit, we cannot distinguish whether the unit approaches an enemy (or ally) or retreats. Therefore, we monitor changes in distances.

B. Encoding Plan 2

Encoding plan 2 involves nine elements, similar to encoding plan 1. But, plan 2 differs from plan 1 only in how the HPs are treated. In plan 1, the HPs are encoded as fractions of their maxima, whereas in plan 2, we encode the current HP with a value of the close, at intervals of ten. The following are the nine elements involved in encoding plan 2.

- Index indicating which player has to play (turn index)
- **The recent HP of turn unit**
- Manhattan distance to the nearest enemy unit
- Manhattan distance to the nearest ally unit
- Action taken
- **The recent HP of the target unit of action**
- Manhattan distance to the target
- Amount of change in the distance to the nearest enemy unit
- Amount of change in the distance to the nearest ally unit

IV. EXPERIMENT

In our experiments, we considered three variables: whether we use encoding plan 1 or plan 2, whether each n-gram includes the moves of both players or just those of the winner, and whether n-gram statistics uses simple or normalized frequency. These choices produce eight combinations, that is, eight ways to extract Desirable Moves. We test these ways by having AIs compete against each other.

A. Game For Experiment

To test the ideas developed above for Desirable Moves, we use an SRPG created for this experiment. The game screen is shown in Figure 1. Each unit has a HP, and if the HP of a

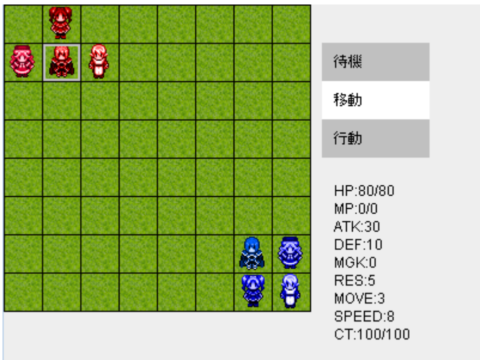

Fig. 1. Custom SRPG games Screen

unit becomes 0, then that unit is dead. Each player operates one of his units in turn. The order of operating units can vary; a unit is selected on the basis of its speed. The winner is the player to first defeat all enemy units.

B. Extraction of Desirable Moves

Match logs were created by competing against AIs that reproduce the typical actions of a player who was an expert of the game. A total of 100 of these match logs were used to provide Desirable Moves. In this experiment, moves whose frequencies or normalized frequencies were low appear few times in the match logs. Therefore, we extracted Desirable Moves from the $1,000$ moves ranked highest using frequency or normalized frequency.

C. Experimental Results and Evaluation

The AI selects the next move to be that having the highest rank in the list of Desirable Moves; this was done by checking all the next states that can be reached for each available action that can be taken, consistent with the history of transferring states to present. When the moves that can be taken do not exist in the set of Desirable Moves, the AI chooses a random move.

To distinguish among Desirable Moves, we use the following notation to identify choices among three variables. A Desirable Move may be based on frequency (F) or normalized frequency (NF); its n-grams may contain moves for both players (D) or just the winner (S); its encoding patterns may be either from encoding plan 1 (1) or encoding plan 2 (2). For example, a situation based on frequency containing moves for both players in each n-gram and using encoding pattern 1 is represented by FD1. This gives eight possible combinations of the three variables. Each combination was played in 500 matches.

Results for the winning ratios of player 1 (P1) to player 2 (P2) are shown in Table I for frequency vs. frequency, Table II for normalized frequency vs. normalized frequency, and Table III for frequency vs. normalized frequency.

From all three tables, we see that encoding plan 1 wins games more often than encoding plan 2. The difference in the two encoding plans was in how to handle the HPs. In this experimental environment, the maximum HP of each unit differs, and even the maximum is less than 100. Therefore, encoding plan 1 that handled the HPs as ratios gave smaller particle sizes. This suggests that the way in which the HP

TABLE I
WINNING RATIOS (P1/P2) WHEN BOTH AI PLAYERS USED DESIRABLE MOVES BASED ON SIMPLE FREQUENCIES.

P1\P2	FD1	FD2	FS1	FS2
FD1	–	0.668	0.696	0.684
FD2	0.332	–	0.576	0.64
FS1	0.304	0.424	–	0.58
FS2	0.316	0.36	0.42	–

TABLE II
WINNING RATIOS (P1/P2) WHEN BOTH AI PLAYERS USED DESIRABLE MOVES BASED ON NORMALIZED FREQUENCIES.

P1\P2	NFD1	NFD2	NFS1	NFS2
NFD1	–	0.516	0.516	0.524
NFD2	0.484	–	0.496	0.48
NFS1	0.484	0.504	–	0.558
NFS2	0.476	0.52	0.442	–

is encoded is an important element in identifying Desirable Moves. The results also suggest that the performance of the AI by Desirable Moves improves by making a particle size smaller. But, we expect that making the particle size too small decreases versatility and performance; therefore, it is necessary to find the right balance.

For play between AIs using normalized frequencies (Table II), the winning rate is almost 0.5, and the win rate seems to approach 0.5 as the numbers of matches are increased. When normalized frequencies are used, there seems to be little influence of the particle size of the HP or whether moves from one or both players are included in the encodings. However, when Desirable Moves are extracted by simple frequency, the method including moves from both players has the better winning rate. In addition, Table III shows that extracting Desirable Moves using frequency is better than extracting themed by using normalized frequency. These results suggest that knowledge about enemy moves and one's own strategy is better than knowing only one's own strategy. However, this observation becomes less relevant when a game becomes very long. Then, choices for effective action become less clear.

Finally, Table IV shows the results when the AI that provided the original match logs competed against each of the eight AIs discussed above. The original AI completely defeated all eight of the proposed AIs, except the combination NFS2 won a few times. In the two-way comparisons of Tables II and III, the NFS2 combination lost nearly every time. Note that in the NFS2 combination, Desirable Moves were extracted using normalized frequencies, only one player's moves, and encoding plan 2.

TABLE III
WINNING RATIOS (P1/P2) WHEN ONE AI PLAYER USED DESIRABLE MOVES BASED ON SIMPLE FREQUENCIES AND THE OTHER USED DESIRABLE MOVES BASED ON NORMALIZED FREQUENCIES.

P1\P2	NFD1	NFD2	NFS1	NFS2
FD1	0.646	0.652	0.648	0.68
FD2	0.592	0.582	0.612	0.624
FS1	0.544	0.546	0.564	0.57
FS2	0.516	0.562	0.594	0.58

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE IV
FRACTION OF WINS WHEN THE AI USED TO CREATE THE ORIGINAL MATCH LOGS COMPETED AGAINST EACH OF THE OTHER EIGHT AIS. THE ORIGINAL AI WON ALMOST EVERY TIME.

FD1	0	NFD1	0
FD2	0	NFD2	0
FS1	0	NFS1	0
FS2	0	NFS2	0.004

V. CONCLUSION

Using match logs from an SRPG competition, we converted the logs to character strings that contain transitional histories of encoded game states; then, we extracted Desirable Moves using n-gram statistics. In addition, we studied two types of encoding methods and a total of eight combinations of methods for extracting Desirable Moves. These were compared by having AIs based on each combination compete with each other. Of course, more encoding methods are possible; therefore, more encodings and combinations must be studied. In this study, the number of match logs used for the extraction of Desirable Moves was constant; therefore, we should also study how performance changes when the number of match logs is increased and decreased. Match logs themselves competed with other AIs, but a different result may be obtained when using the log from a human player.

We used only Desirable Moves in these experiments, but extracting Undesirable Moves can also be done by the same method. Then, by preventing an AI from choosing an undesirable move the performance resulting from choosing desirable moves may be improved. Furthermore, any moves that appear in the high ranks of both Desirable and Undesirable Moves may have little influence on victory or defeat; therefore, removing those from the list of Desirable Moves may also improve performance.

Surprising results were obtained when the AIs using Desirable Moves competed against the AI that served as the basis for the match logs. Under present conditions, it is difficult to judge whether factors that are important by these two types of play are different or the results simply occurred by chance. This must be addressed in our future work.

The extraction of Desirable Moves studied here is a first step towards producing more meaningful tutorials that will help novice players master video games quickly and in a satisfying way. It remains a problem to identify the best ways to provide extracted Desirable Moves to players.

REFERENCES

[1] T. Nakamura, "Acquisition of move sequence patterns from game record database using n-gram statistics," *Game Programming Workshop 97 (GPW97)*, pp. 96–105, 1997.

[2] M. Nagao and S. Mori, "A method of n-gram statistics for large text data of japanese, and the automatic extraction of words and phrases." *IPSJ SIG Technical Report. IPSJ-SIGNL Report*, vol. 93, no. 61, pp. 1–8, 1993-07-09. [Online]. Available: http://ci.nii.ac.jp/naid/110002934647/

[3] S. Shimohata, T. Sugio, and J. Nagata, "Extraction of frozen patterns from corpora through entropy thresholds," *IPSJ SIG Technical Report. IPSJ-SIGNL Report*, vol. 95, no. 110, pp. 71–78, 1995-11-17. [Online]. Available: http://ci.nii.ac.jp/naid/110002935078/

[4] H. Nakawatase, "A keyword extraction method by using statistical text information," *IEICE Technical Report. DE95(81)*, vol. 95, no. 81, pp. 9–16, 1995-05-26. [Online]. Available: http://ci.nii.ac.jp/naid/110003188732/

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)