Analytics-Driven Dynamic Game Adaption for Player Retention in *Scrabble*

Brent Harrison
North Carolina State University
Raleigh, North Carolina 27695

David L. Roberts
North Carolina State University
Raleigh, North Carolina 27695

Abstract—This paper shows how game analytics can be used in conjunction with an adaptive system in order to increase player retention at the level of individual game sessions in *Scrabblesque*, a Flash game based on the popular board game *Scrabble*. In this paper, we use game analytic knowledge to create a simplified search space (called the *game analytic space*) of board states. We then target a distribution of game analytic states that are predictive of players playing a complete game session of *Scrabblesque* in order to increase player retention. Our adaptive system then has a computer-controlled AI opponent take moves that will help realize this distribution of game analytic states with the ultimate goal of reducing the quitting rate.

We test this system by performing a user study in which we compare how many people quit playing the adaptive version of *Scrabblesque* early and how many people quit playing a non-adaptive version of *Scrabblesque* early. We also compare how well the adaptive version of *Scrabblesque* was able to influence player behavior as described by game analytics. Our results show that our adaptive system is able to produce a significant reduction in the quitting rate (p = 0.03) when compared to the non-adaptive version. In addition, the adaptive version of *Scrabblesque* is able to better fit a target distribution of game analytic states when compared to the non-adaptive version.

I. INTRODUCTION

In games research as well as in the game industry, there has always been a focus on understanding player retention. *What do players like or dislike about games? What drives a player to stop playing a game? What can we do to keep players playing for longer?* These are some of the motivating questions that drive player retention research. There has already been a great deal of research done on determining motivations for play in games [1, 2] and determining the main contributors to players quitting games; however, less work has been done on incorporating this knowledge into a system in order to increase the amount of time a person plays a game.

In this paper, we explore what can be done to keep people playing a game for longer. In particular, we examine how the use of game analytics can be paired with an adaptive system in order to increase the amount of time people play a game. For the purposes of this work, we have chosen to address player retention on the level of individual game sessions rather than retention over the course of several sessions.

We have chosen to examine this problem using a testbed game of our own creation, *Scrabblesque* (see Figure 1). *Scrabblesque* is a Flash game that was designed to mimic the popular board game, *Scrabble*, with a few key differences.

Fig. 1. A screenshot of *Scrabblesque*.

In *Scrabblesque*, the player competes against a computer-controlled AI instead of another person and the goal of the game is to be the first player to receive 150 points. The reason that we are studying player retention in this game environment is because of the similarities that exist between this environment and many social games currently on the market. Both social games and *Scrabblesque* offer the player a limited number of actions they can take at any given time and a limited number of ways to interact with the game environment. Social games can also benefit greatly from a system that can increase player retention, even at the granularity of individual game sessions, since a majority of game companies' revenue is generated through advertisements and microtransactions.

At a high level, the work presented in this paper builds off of the following concept concerning player retention: *game states exist that are condusive to player retention whereas there are certain game states that will lead to players ending their game session before the game's natural conclusion*. In terms of *Scrabblesque*, we can phrase this concept in the following way: *board states exist that can be used to predict if a player will reach the end of the current game or not*.

Knowing this, the goal of our system becomes clear. Our system should strive to target a *distribution* of the board states that are predictive of players reaching the end of the current game while avoiding board states that are predictive of the player quitting the game before the game's natural conclusion. There are two reasons that we target a distribution of board states to increase retention. The first of these is to increase the replay value of our game since a variety of different experiences will be produced. The second reason is because players' experiences are only partially under our control. While we can make changes to the game environment or mechanics to steer players towards the desirable board states, ultimately they are taking their own actions in the environment. This nondeterminism has been shown to be effectively modeled by target distributions in the past [3].

This presents a problem, however, in that the space of possible board states in *Scrabblesque* is truly massive. As a result, it is intractable to feasibly target a distribution over the true set of *Scrabblesque* board states that are predictive of a player reaching the game's conclusion. To resolve this issue, we choose to reduce the space of possible board states by representing sets of board states in terms of *game analytics*. Game analytics are statistics gathered during gameplay that can be used to describe several aspects of gameplay. In this work, we use game analytics as an abstraction for board states so that the task of targeting a distribution over these states becomes a tractable problem. The main result of this abstraction is that we now must move through this *game analytic space* by making changes to these analytic values. For example, if we have chosen to abstract board states by using the average length of the words on the board, we would move through the space (and towards our goal states) by altering this value.

This abstraction leads to the questions that are at the heart of our argument. *Are game analytics powerful enough to use as the basis for an adaptive system for player retention? Is moving through this game analytic space enough to result in a change in player quitting behavior in Scrabblesque?*

II. RELATED WORK

In this paper, we explore the use of an adaptive system in conjunction with game analytics to affect player retention in *Scrabblesque*. Here, we will review some of the relevant literature in player retention research as well as the relevant literature in adaptive gaming technology.

A. Player Retention

In this paper, we have chosen to address the issue of *retention*, or how to encourage people to play a game for a longer period of time. There has been a great deal of research done on determining what aspects of games cause people to quit playing certain games permanently; however, there has been very little research done about retention at the game session level as we do in this paper.

Tarng, Chen, and Huang [4] studied the play tendencies of 34,524 World of Warcraft players during a two year period to determine if there are any indicators that a player would quit playing. They determined that while predicting a player's short term behavior is feasible, predicting long term behavior is a much harder problem. Tarng, Chen, and Huang [5] performed a follow-up study on a different MMORPG using support vector machines (SVMs) as their predictor. They concluded that it is possible to predict when a player will quit in the near future as long as they do not possess wildly erratic play behavior. This work examines the question of retention over the course of several game sessions, which is different from our goal of increasing player retention at the level of a single game session.

Researchers have also studied the factors that most contribute to player retention in video games. Weber, Mateas, and Jhala used regression to determine what features most contribute to retention in both Madden '11 [6] and Infinite Mario [7]. In Madden '11, the authors use three types of regression techniques (linear regression, M5 regression [8], and additive regression [9]) to determine the features of gameplay that most influence the number of games that a player will play. The features examined were statistics such as offensive play diversity and the number of *game changing plays* (such as interceptions). The authors used a similar technique in infinite Mario to rank features based on their influence over player retention. In our work we build on this idea by using analytics that we have identified to affect player retention in *Scrabblesque* as the basis for an adaptive system in order to increase play time.

Andersen *et al.* [10] showed that the presence of secondary objectives can have an adverse affect on player retention. They showed this by examining the play times of players when playing a game that had secondary objectives (in this case, coins that players could collect) versus players when playing a game that had no secondary objectives. They showed that there was a noticeable difference in the amount of players that played the game for long amounts of time. Again, our work differs from this in that we use information that we have already gathered on player retention in *Scrabblesque* to inform an adaptive system with the goal of reducing the number of players that quit the game session early by making changes to the game at runtime.

B. Adaptive Games

Perhaps the most common strategy to adapt games is to implement a *dynamic difficulty adjustment* (DDA) system. These games will determine the expertise level of the current player and then change the difficulty of the game to best suit that player.

The AI director in Left 4 Dead [11] alters the amount and type of enemies that players will face depending on the perceived amount of *intensity* that players are experiencing. *Intensity* is measured by examining analytics such as how many enemies are on the screen and the amount of damage that has been taken by the players. In this system, there is an alternating cycle of rising tension and relaxation. Once the intensity level of the players has risen above a certain level, the number of enemies begins to decrease. Once this relaxation period has continued for a set amount of time, enemies begin to become more frequent until a certain intensity threshold is met and the cycle begins anew. This system is similar to ours in that it uses analytics in order to estimate the space of possible games that can be produced and then alters those analytics in order to produce different experiences. The differences between our system and the system implemented in

Left 4 Dead is that it does not explicitly target a distribution over games states. The Left 4 Dead AI director uses a set of rules that defines how these analytics should be altered in different situations. Additionally, the AI director targets suspense or tension in the game, rather than player retention.

Another well known example of DDA is the *rubber banding* technique used in games such as Mario Kart [12]. This technique will give noticeable boosts in speed and item quality to players, be they human or computer-controlled, that are doing poorly. So, rather than adjust the quality of the AI based on how well or poorly a player is doing, the mechanics of the game are changed to favor those who are not performing well. In our system, the mechanics of the game environment are static and, therefore, not the source of game adaption. In our system, adaptions are made by intelligently choosing the actions that the computer-controlled AI will make such that they result in a targeted change in analytic values.

Jennings-Teats *et al.* [13] alter the environment of infinite Mario by adding pits and different platform levels in order to dynamically adjust the difficulty. In this system, a statistical model of player skill is used to determine the placement of enemies and the placement of obstacles to ensure that the player is sufficiently challenged. This system uses a strategy similar to ours in that they use analytics to probabilistically adapt the game environment; however, here they are procedurally generating content rather than directly affecting the AI. In this system, we do not have the control to make alterations to the game world itself since we are working with a static game board. All we can do is make subtle adaptions through altering how the computer-controlled AI plays words and gives tiles to the player.

Andrade *et al.* [14] proposed the use of reinforcement learning as a way to teach agents to dynamically adjust their skill in a fighting game. In their system, a reinforcement learning agent takes actions (punches and kicks) in such a way that the fight ends with both combatants having a small difference in life totals, indicating that the agent was a challenging opponent, but not so much as to cause frustration in the player. This work is similar to ours in that it seeks to alter a game analytic (difference in life totals) to affect behavior; however, here they use a simple rule to determine the actions to take rather than targeting a distribution of possible values.

Spronck *et al.* [15] use *dynamic scripting* [16] to adjust difficulty by shifting the way that actions are chosen in their system. Their system has three ways to adjust difficulty: high-fitness penalizing, weight clipping, and top culling. High-fitness penalizing rewards the agent for choosing actions that have a moderate fitness value rather than those that have high fitness values. Weight clipping puts a limit on the probability that certain rules will be selected which increases the chance that the agent will choose suboptimal moves. Top culling makes it impossible to choose rules that perform too well, forcing the agent to take suboptimal actions. Our work differs from this in that we do more than just inhibit how *well* the AI can perform. Our work can be seen as altering the success criteria for our AI in that *good moves* are defined by how close they move to the target game analytic states. In other words, the work done by Spronck *et al.* ascribes to the *artificial stupidity* method of game AI programming [17] which places hard limits on how well the AI can perform, whereas our
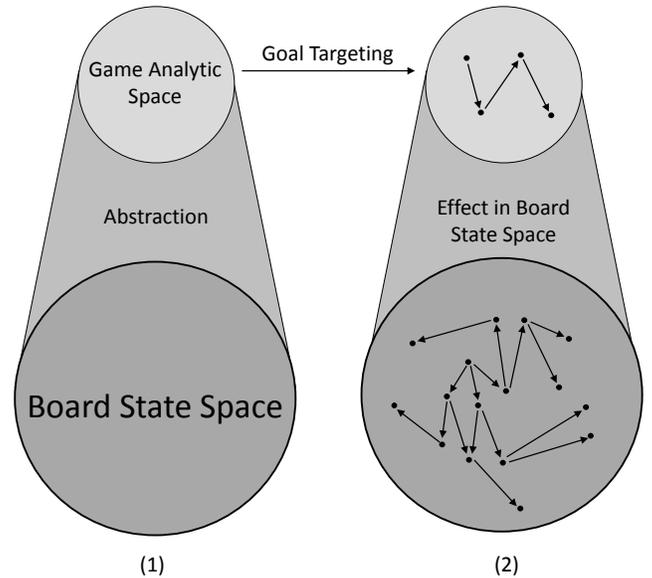

Fig. 2. An overview of the steps taken to increase retention in *Scrabblesque*. The first step is to abstract the board state space into a game analytic space. The second step is to move through this game analytic space in accordance with a target distribution.

system uses *intelligent mistakes* [18], a game AI programming practice which alters how the AI determines success.

While not a technique for creating adaptive games, the adaptive technique that is most like ours is the targeted trajectory Markov decision process (TTDMDP) [3] that is built on the declarative optimization based drama management formalism [19]. TTD-MDPs were created to solve the problem of replayability in an adaptive story world. TTD-MDPs target a distribution of stories, or trajectories through the space of plot events, and attempt to form a policy such that this distribution of stories, as opposed to the best story, will occur. Our work takes the idea of targeting a distribution of goal states and moves it from the story world into a game environment. It has previously been speculated that TTD-MDPs could be used in more general game environments [20]; however, to our knowledge that has yet to occur.

III. METHODOLOGY

In this section, we will review how our system incorporates game adaptivity and game analytics in order to affect player retention in *Scrabblesque*. A high level overview of our methodology can be found in Figure 2. This figure shows that our process takes place in two general phases. During the first phase (denoted by "(1)" in the figure), we make the problem tractable by abstracting the full space of possible board states into a much more manageable space through the use of game analytics. The second phase (denoted by "(2)" in the figure) involves using an adaptive system in order to move through this game analytic space by altering analytic values in accordance with a target distribution of game analytics. Also shown is how a set of moves in this game analytic space might translate back into board states. A small set of moves in game analytic space can correspond to hundreds or even thousands

of possible moves in the board state space, any one of which will suffice. Each of these phases will be discussed in greater detail below.

A. Game State Abstraction

As we have stated earlier, the goal of this paper is to introduce a system for increasing player retention at the session level in *Scrabblesque*. In other words, we want more people to play *Scrabblesque* until the current session's natural end. Also recall that we have stated that this goal can be described in terms of board states. If that is the case, then the goal is to influence the game in such a way so that the current board state is predictive of the player completing the current game while actively avoiding board states that are predictive of the player ending the current game before it normally would. In *Scrabblesque*, this is a problem because the game has a very large space of possible board states. To give some insight as to exactly how big the board state space is, consider only the player's rack of tiles. In *Scrabblesque*, the player is given a rack containing 7 letter tiles that they can use to make words on the board. There are 100 possible tiles that can be on the rack, making the total number of different racks equal to the following:

$$\frac{100!}{93!7!} = 16,007,560,800 \quad (1)$$

By only considering the rack, *Scrabblesque* has over 16 billion unique board states. If we consider the board and the number of states it could possibly be in, then the number of possible unique board states becomes even larger. In order to tractably model player retention in this space, we need to describe the space using a smaller set of analytics that are still predictive of player retention in *Scrabblesque*.

In previous work, we identified three game analytics in *Scrabblesque* that are predictive of player retention (CITATION REMOVED FOR BLIND REVIEW). These three analytics are:

1) **Word Score**: How much the discrete valued word score of the last word played for the player deviates from the expected word score across all observed players
2) **Word Length**: How much the discrete valued length of the last word played for the player deviates from the expected word length across all observed players
3) **Score Difference**: How much the discrete valued difference in player and computer scores deviates from the expected score difference values across all observed players

In this paper, we will not go into detail about how these analytics were computed. For a more detailed discussion, please refer to (CITATION REMOVED FOR BLIND REVIEW). Since player retention is intertwined with game experience, we chose to describe the space of possible board states using analytics that describe the game experience. Each of the above analytics are descriptive of the possible game experiences in *Scrabblesque*. Score difference, for example, describes how much better or worse a player is performing compared to how past players have done. Therefore, observing any fluctuations in this value could be indicative of either the player

performing poorer than expected and becoming frustrated with the experience, or the player performing much better than expected and then becoming bored with the experience. Using the knowledge of predictive analytic states generated in this previous work, we can define our target distribution of game analytic states.

These analytics present a problem, however. These analytics describe game interactions involving both the player and the computer-controlled AI. Word length and word score measure the performance of the player on their turn while score difference measures the overall performance of the player in comparison to the computer-controlled AI. This presents a problem since we cannot directly control player actions. Recall that movement through the space of game analytic states is accomplished by altering the values of these analytics. Since we cannot directly control these analytic values, it becomes impossible to deterministically move through this space. This means that it is impossible to move towards any goal states as defined by our target distribution with certainty.

To address this issue, we must derive a second set of analytics that are reflective of the effects that the AI opponent's moves have on the evolution of game analytics. Since this set of analytics is used to describe only the AI opponent's moves, it describes only the aspects of the game environment that the AI opponent can control. We call this second set of analytics *directly affectable game analytics*. In *Scrabblesque*, the computer-controlled AI can only play a word on the game board and fill the player's rack once they have played a word. With this in mind we use the following set of analytics to describe the state of the game board:

- Number of Candidate Tiles: The number of eligible tiles that enable the player to connect the words they play to existing words on the board

- Number of Consonant/Vowel Candidate Tiles: The number of candidate tiles that are consonants/vowels

- Average Tile Value: The average value (in terms of game score) of candidate tiles

- Proximity to Bonus Squares: The number of bonus squares that the player can reach in a single turn using the tiles in their rack

In addition to these analytics, we also use three others to describe the tiles in the player's rack:

- Number of Consonant/Vowel Tiles: The number of consonant/vowel tiles present on the player's rack

- Average Tile Value: The average value of the tiles in the player's rack

- Number of Repeated Tiles: The number of tiles that are repeated in the player's rack

Since we can manipulate these directly affectable game analytics through the actions that the computer-controlled AI takes, we can now successfully move through the space of game analytics in order to realize the target distribution of game analytic states. Note that these features we just described represent the affordances for players to make their moves in the game. Therefore, these features enable reasoning about how to

adapt the game to affect how a player makes their moves in the environment, which will indirectly have an effect on the distribution of board state features described above.

B. Adaptive System

As we discussed previously, representing the space of board states in *Scrabblesque* in terms of *directly affectable game analytics* enables us to make moves that can bring about game analytic states that are predictive of player retention. This enables us to use these game analytics as the basis of an adaptive system to increase player retention. How exactly do we do this? To show how our adaptive system works, we have chosen to use an informative example.

In this example, we will assume that in order to retain the player we must achieve a score difference value of less than 20 points. In other words, the difference in player score and computer score must not deviate from the expected value of this analytic by more than 20 points. Since score difference depends on both the computer's and the player's turn, we must have the computer move in such a way as to produce a board state that, before the player makes their move, is conducive to bringing about the desired score difference after the player has made their move.

The way we achieve this, in practice, is to query a corpus of board states for examples that exhibit the desired score differential. This corpus is composed of board states gathered from players that have played games of *Scrabblesque* in the past. For each board state that has a score difference of less than 20 points, we retrieve the preceding intermediate board state after the computer's turn. Once we have these intermediate board states, we represent them in terms of the *directly affectable game analytics* that we discussed previously.

In order to determine the move that the computer should make we simply simulate possible actions that the computer can make (placing words and giving the player tiles) and then measure the distance between this set of resultant intermediate board states and the set of intermediate board states extracted from the corpus. It is important to note here that the computer is limited in the types of moves it can make. It can only play words on the board or give tiles to the player if the tiles involved in performing those actions have not already been used. In other words, the computer can only simulate *legal* moves that it can take. The computer then takes the move that minimizes the distance between the intermediate board state that would result from the action being taken and the set of intermediate board states retrieved from the corpus. Here, we calculate Euclidean distance using the set of directly affectable game analytics; however, any arbitrary distance or divergence measure like KL-divergence or Bhattacharyya divergence [21] can be used.

To summarize, the steps taken by our adaptive system are as follows:

1) Select a goal analytic state based on the target distribution of game analytic states that are predictive of player retention
2) Retrieve a set of example board states from a corpus that are consistent with the goal analytic state
3) Retrieve the set of preceding intermediate board states in terms of the set of directly affectable game analytics
4) Simulate possible computer actions
5) Perform the action that minimizes the Euclidean distance between the set of candidate intermediate states and the intermediate states retrieved from the corpus

These steps are represented in the the flowchart in Figure 3.

IV. EVALUATION

To validate the effectiveness of our adaptive system at retaining players in *Scrabblesque*, we first implemented it to target a distribution authored using data gathered from a previous study (CITATION REMOVED FOR BLIND REVIEW). This target distribution was authored by simulating complete games of *Scrabblesque* based on the moves that players had taken in the past. Simulated board states that were not predictive of player retention were discarded and then new moves were calculated. The end result was a set of 100,000 simulated games that each consisted only of board states that were predictive of player retention. We then calculated the distribution of analytic values (word length, word score, and score difference) present in these games and used this as our target distribution. It is worth noting that this is only one way to author a target distribution. There are many other techniques that exist for this task, and we refer the reader to [22] for more information on authoring target distributions.

Using this target distribution for player retention, we performed a user study and examined the logs of player data that were produced. In the following experiments, we compare the results of our adaptive system with the results of a study that we previously conducted with a non-adaptive version of *Scrabblesque* (CITATION REMOVED FOR BLIND REVIEW).

For this evaluation, we are using two criteria for determining success: quitting rate in *Scrabblesque* and the KL-divergence between the target distribution and the distribution of observed player traces. Recall that the ultimate goal of our adaptive system is to reduce the number of players that ended their current game session of *Scrabblesque* before it came to its natural conclusion. With this in mind, it makes sense to use the quitting rate as the primary evaluative metric. The motivation behind using KL-divergence is to evaluate how successful our adaptive system is at influencing changes in analytic values (in other words, to show that any changes in the quitting rate can be linked to the changes in these analytic values). Since we are targeting a probability distribution of analytic values, our adaptive system should be able to fit this distribution of values better than a non-adaptive game.

A. Data Collection

For these experiments, we deployed *Scrabblesque* online and recruited participants via email distribution lists and social networking sites (Facebook, Twitter, *etc.*). We used snowball sampling as we encouraged participants who had taken the study to share the experiment with their friends and family.

Data collection proceeded in two phases. First, in early work (CITATION REMOVED FOR BLIND REVIEW) we
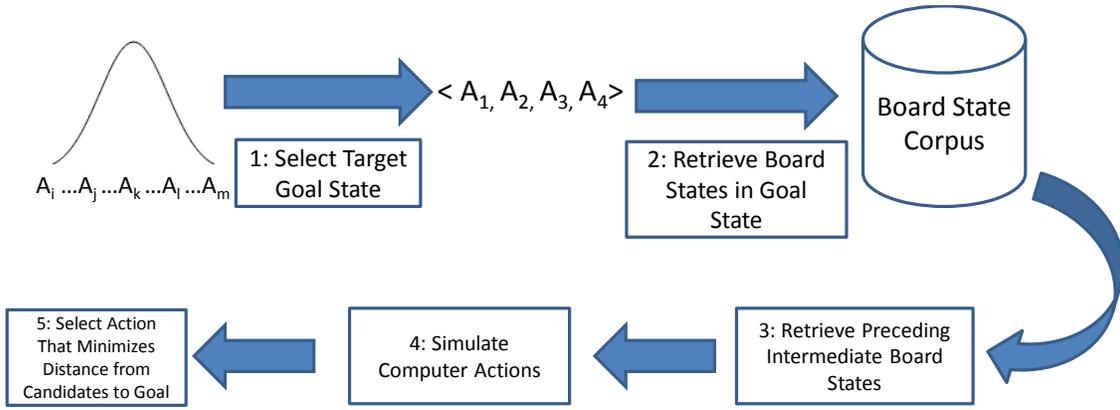
Fig. 3. A flowchart showing the execution of our adaptive system. First, a goal is chosen. Then, instances of this goal are found in the knowledge corpus and used to evaluate a set of candidate actions.

	Finished	Unfinished	Unfinished Game Percentage
Baseline	148	47	24.1%
Adaptive	55	7	11.3%

collected data from 195 games. This data was collected from a version of *Scrabblesque* that was non-adaptive. We decided to use the data gathered from this experiment as a baseline since it was gathered in a non-adaptive version of the game. We will refer to this data as the *baseline data* from now on.

During the second phase of data collection, we asked players to play a modified version of *Scrabblesque* in which our adaptive system had been implemented. All users that played *Scrabblesque* during this phase of data collection played the adaptive version of *Scrabblesque*. By the end of this phase of the data collection process, we had obtained 62 games of *Scrabblesque*. We will refer to this data as the *adaptive data* from now on.

B. Analyzing Quitting Rate

Recall that we authored the target distribution being used in this evaluation using data collected in previous work. In that work we found that there are certain game analytic states in *Scrabblesque* that are indicative of player retention. As a result, we authored this target distribution to increase the occurrences of these game analytic states. Accordingly, the main method for evaluating the effectiveness of our adaptive system is to see how effective it is at lowering the frequency of players quitting the game early.

According to the data we had gathered previously on the non-adaptive version of *Scrabblesque*, 24.1% of all games were ended prematurely. Using our adaptive system, we were able to reduce this percentage to 11.3%. A summary of this result can be seen in Table I. This difference was significant according Fisher's exact test (p=0.03).
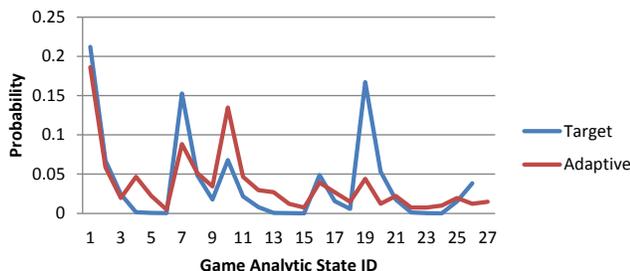
C. Distribution Analysis

The second evaluation done was an analysis of how much the distribution of observed game analytics diverged from the targeted distribution of game analytics for both the non-adaptive version of *Scrabblesque* and the adaptive version of *Scrabblesque*. The reason for this analysis is to give more evidence that the differences in quitting rate were, indeed, the result of our adaptive system and not random chance. If, for example, we see that players who played the adaptive version of *Scrabblesque* more closely match our target distribution, then it is reasonable to say that the adaptions we made produced a shift in behavior that contributed to the reduction in the quitting rate.

Figure 4 shows a visual comparison of the game analytic distributions produced by both versions of *Scrabblesque* compared against the target distribution. In this figure, the *Game Analytic State ID* refers to an arbitrary ID given to each game analytic state represented in terms of word score, word length, and score difference. The important thing to note is that each game analytic state is given the same ID value across figures, so the comparison is fair. As you can see in the figure, there are 4 peaks in the target distribution: game analytic state ID 7, game analytic state ID 10, game analytic state ID 16, and game analytic state ID 19. Visual comparison between Figure 4(a) and Figure 4(b) show that the adaptive distribution more closely fits the peak at game analytic state ID 7, game analytic state ID 10, and game analytic state ID 16.

This visual comparison is not definitive proof that the distribution produced by the version of *Scrabblesque* using our adaptive system better fits the target distribution than the non-adaptive version of *Scrabblesque*. In order to statistically evaluate how well the game analytic distribution produced by our adaptive system fit the target distribution, we calculated the KL-divergence between the target distribution and the distributions produced by both adaptive and non-adaptive versions of *Scrabblesque*. KL-divergence is not a true distance, as it is asymmetric; however, it is a well-understood measure with several important properties. In particular, it is consistent, always non-negative, and zero only when the two distributions are exactly equal. In this case, KL-divergence measures how well each test distribution approximates the target distribution by measuring the entropy in the target distribution that is unexplained by the test distribution. Since the KL divergence is an asymmetric divergence value, we chose to turn this into

Comparison of Target and Adaptive Distributions

(a) Adaptive and Target Comparison

Comparison of Target and Baseline Distributions
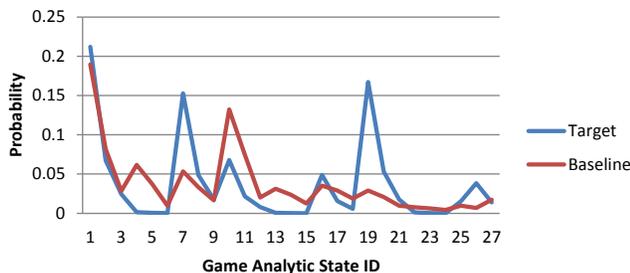
(b) Baseline and Target Comparison

Fig. 4. Visual comparison between the adaptive distribution and the target distribution as well as the baseline distribution and the target distribution.

TABLE II. KL-DIVERGENCE VALUES COMPARING THE BEHAVIOR DISTRIBUTION PRODUCED BY THE NON-ADAPTIVE (BASELINE) VERSION OF *Scrabblesque* AND THE ADAPTIVE VERSION OF *Scrabblesque*. SINCE KL-DIVERGENCE IS ASYMMETRIC, COMPARISONS WHERE DONE IN BOTH DIRECTIONS AND THEN AVERAGED.

	DvT	TvD	Average
Baseline	0.86	0.51	0.69
Adaptive	0.59	0.36	0.48

a distance value by using the following formula:

$$\frac{KL(test, target) + KL(target, test)}{2} \quad (2)$$

In the above equation, $KL(test, target)$ indicates that we calculated the KL-divergence between a test distribution and the target distribution. As you can see, we calculate the KL-divergence in both directions and then take the average to turn it into a distance measure. The results of this analysis can be seen in Table II. As you can see in the Table, we found that in the adaptive condition KL-divergence values were 0.59 and 0.36 for the observed distribution versus the target distribution and the target distribution versus the observed distribution, respectively. This leads to an average KL-divergence value of 0.48, which is lower than the KL-divergence values achieved by the non-adaptive version of *Scrabblesque*. From this, we can conclude that our adaptive system does induce a shift in the game analytic distribution towards the target distribution.

V. DISCUSSION

These results show that the presence of our adaptive system brings about a statistically significant drop in the quitting rate in *Scrabblesque*, indicating an increase in player retention. This result gives further insight into the power that these analytics can have in influencing player retention. The underlying theory behind this system is that there are certain board states that are predictive of player retention and that they can be represented using a set of directly affectable game analytics. We hypothesized that altering the game to bring about game analytic states that are predictive of player retention should result in an increase in player retention. The results support this underlying hypothesis by showing that altering these analytics in *Scrabblesque* resulted in a significant change in the quitting rate. Although not conclusive, these results are strong evidence for the direction of causality: *it appears that features of gameplay influence players to continue playing, and it appears **not** to be the case that players who continue playing cause certain features of gameplay.*

These results are also promising in that there was another layer of indirection involved in the exact implementation of our adaptive system in *Scrabblesque*. Recall that we had to represent goals in terms of analytics characteristic of game play experience, but actions had to be represented in terms of directly affectable game analytics. This added indirection, we feel, gives our results extra strength as we were able to reduce the quitting rate through two levels of indirection (by using affectable game analytics to produce a shift in the target distribution over non-affectable analytics which resulted in a decrease in the quitting rate).

VI. FUTURE WORK

This technique offers many exciting avenues for future research. Perhaps the first question that must be answered is whether or not this technique generalizes to different game environments. This game environment, as has been discussed earlier, is relatively simple in terms of the amount of interaction the player is afforded and the types of actions the computer can take. It would be interesting to see how our adaptive system would scale to a larger environment with more possibilities for user actions.

It would also be interesting to explore the use of this system for purposes other than player retention. In this work, we had access to data that showed that certain game analytic states existed in *Scrabblesque* that were predictive of player retention. If we had access to different data, could this system be used to bring about different changes in behavior?

A possible follow up study to this one could evaluate if the adaptions that this system produces have a noticeable effect on player experience from the point of view of the player. Did they have more fun? Were they more engaged? If this is the case, then such a study would provide further support for the power of this algorithm.

VII. CONCLUSION

In this paper, we have shown how game analytics in conjunction with an adaptive system can increase player retention in *Scrabblesque*. This system works by having the

computer-controlled AI make moves that result in changes to various analytic values that describe board state. These moves are made in accordance with a target distribution over game analytic states that are predictive of player retention in the game.

We have shown that this system in conjunction with game analytics that are used to reduce the space of possible board states can effectively reduce the quitting rate in *Scrabblesque*. In addition, we give evidence that this reduction in the quitting rate can be linked to the change in analytic values by showing that the behaviors produced by our adaptive system better fit the target distribution of game analytic states when compared against a non-adaptive game.

Our technique presents a significant step forward in the area of player retention in that we have successfully implemented an adaptive system and shown that it produces a significant reduction in the quitting rate. Also, this seems to be one of the first works to be done on improving retention at the level of game sessions. We have also shown how powerful the use of game analytics can be since technique relies on only 3 game analytics to produce desired changes in the quitting rate in *Scrabblesque*. This result also provides evidence towards the direction of the causality in the relationship between game analytics and player retention. While our result alone is not enough to be conclusive, it helps to support the idea that guiding players into game states that produce analytic values predictive of desirable behaviors can encourage those behaviors in the future. Moving forward, we hope that this paper will encourage researchers to examine the use of game analytics in other areas of AI in games.

REFERENCES

[1] N. Yee, "Motivations for play in online games," *CyberPsychology & Behavior*, vol. 9, no. 6, pp. 772–775, 2006.

[2] R. Bartle, "Hearts, clubs, diamonds, spades: Players who suit MUDs," *Journal of MUD research*, vol. 1, no. 1, p. 19, 1996.

[3] D. Roberts, M. Nelson, C. Isbell, M. Mateas, and M. Littman, "Targeting specific distributions of trajectories in mdps," in *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, vol. 21, no. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006, p. 1213.

[4] P. Tarng, K. Chen, and P. Huang, "An analysis of wow players' game hours," in *Proceedings of the 7th ACM SIGCOMM Workshop on Network and System Support for Games*. ACM, 2008, pp. 47–52.

[5] P.-Y. Tarng, K.-T. Chen, and P. Huang, "On prophesying online gamer departure," in *Network and Systems Support for Games (NetGames), 2009 8th Annual Workshop on*, nov. 2009, pp. 1 –2.

[6] B. Weber, M. John, M. Mateas, and A. Jhala, "Modeling player retention in madden nfl 11," in *Twenty-Third IAAI Conference*, 2011.

[7] B. Weber, M. Mateas, and A. Jhala, "Using data mining to model player experience," in *FDG Workshop on Evaluating Player Experience in Games*, 2011.

[8] Y. Wang and I. Witten, "Inducing model trees for continuous classes," in *Proc. of Poster Papers, 9th European Conference on Machine Learning, Prague, Czech Republic*, 1997.

[9] J. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[10] E. Andersen, Y. Liu, R. Snider, R. Szeto, S. Cooper, and Z. Popovic, "On the harmfulness of secondary game objectives," in *Proceedings of the 6th International Conference on Foundations of Digital Games*, 2011, pp. 30–37.

[11] M. Booth, "The ai systems of left 4 dead," in *Keynote, Fifth Artificial Intelligence and Interactive Digital Entertainment Conference (AIIDE09)*, 2009.

[12] E. Jimenez, "The pure advantage: Advanced racing game ai," 2009.

[13] M. Jennings-Teats, G. Smith, and N. Wardrip-Fruin, "Polymorph: dynamic difficulty adjustment through level generation," in *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. ACM, 2010, p. 11.

[14] G. Danzi, A. H. P. Santana, A. W. B. Furtado, A. R. Gouveia, A. Leitao, and G. L. Ramalho, "Online adaptation of computer games agents: A reinforcement learning approach," in *II Workshop de Jogos e Entretenimento Digital*, 2003, pp. 105–112.

[15] P. Spronck, I. Sprinkhuizen-Kuyper, and E. Postma, "Difficulty scaling of game ai," in *Proceedings of the 5th International Conference on Intelligent Games and Simulation (GAME-ON 2004)*, 2004, pp. 33–37.

[16] ——, "Online adaptation of game opponent ai in simulation and in practice," in *Proceedings of the 4th International Conference on Intelligent Games and Simulation (GAME-ON 2003)*, 2003, pp. 93–100.

[17] L. Lidén, "Artificial stupidity: The art of intentional mistakes," *AI Game Programming Wisdom*, vol. 2, pp. 41–48, 2003.

[18] M. West, "Intelligent mistakes: How to incorporate stupidity into your ai code," *Game Developer MagazineDigital Edition*, 2008.

[19] M. J. Nelson, M. Mateas, D. L. Roberts, and C. L. Isbell, "Declarative Optimization-Based Drama Management in the Interactive Fiction Anchorhead," *IEEE Computer Graphics and Applications (Special Issue on Interactive Narrative)*, vol. 26, no. 3, pp. 30–39, 2006.

[20] D. L. Roberts, M. O. Riedl, and C. L. Isbell, "Beyond adversarial: The case for game ai as storytelling," in *Proceedings of the 2009 Conference of the Digital Games Research Association*, 2009.

[21] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bulletin of the Calcutta Mathematical Society*, vol. 35, pp. 99–109, 1943.

[22] D. L. Roberts, S. Bhat, K. S. Clair, and C. L. Isbell, "Authorial Idioms for Target Distributions in TTD-MDPs," in *Proc. of the 22nd Conf. on Artificial Intelligence (AAAI07)*, 2007.