A step in the right direction: Botdetection in MMORPGs using movement analysis

Marlieke van Kesteren [a]   Jurriaan Langevoort [a]   Franc Grootjen [ab]

[a] *Department of Artificial Intelligence, Radboud University Nijmegen*
[b] *Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour*

Abstract

Massively multiplayer online role playing games (MMORPG) is a genre in computer games which is gaining a lot of popularity all over the world. Cheating in these games can often be done by the use of a bot program. Such a program controls a character and allows the player to gain an unfair advantage over legitimate players, because repetitive and often boring tasks can be performed without any effort. Several methods for detecting bots have been proposed in the past. The method described in this research focuses on differences in the frequency of occurrence of different angles in the movement of bots and humans. A histogram is made based on the distribution of angles and a nearest centroid classifier is used. The result of this method is a classification rate of 100% on the used dataset.

1 Introduction

With the growing network bandwidth, increasing cpu and graphic processing power, massively multiplayer online role playing games (MMORPGs) rapidly increase in popularity. Consumers spend an increasing amount of money on subscription to these games [10]. The most well-known example of a MMORPG at the moment is World of Warcraft whose creators reported they had reached more than 11.5 million subscribers worldwide at the end of 2008 [2].

In MMORPGs a person plays a fictional role in a virtual world and thereby interacts with many other players. Each player runs the game on a computer and connects to a server which allows communication between players and sharing of events in the virtual world. A player creates a character which is used as an avatar and chooses a role for their character to play within the game. Players usually develop their character by completing quests and/or killing monsters, alone or in cooperation with other players. The character receives a reward (money or an item) in return and gains experience. Arguably, the most important part of the game is collecting loot (items that drop from monsters when slain, or as rewards for quests completed). These items will determine the strength of your character (together with the experience gained by slaying monsters and completing quests). To attain the best gear you need to play the game (a lot if possible). In order to ease this 'burden', bots are made.

A bot, short for robot, is an automated program with artificial intelligence that players use for different purposes[5]. In case of an MMORPG a bot program controls a player's character by walking around the virtual world and performing tasks. By running a bot program, a player avoids carrying out repetitive tasks themselves while their character continues developing and collects items and money. Many game developers prohibit the use of bots in their game via the End-User License Agreement of the games. The running of bots will have severe negative effects on the game.

- Firstly, the bots are a source of great annoyance to other players. Bots will run around and kill indiscriminately, this means that there will be less monsters to kill for the people who actually want to play and enjoy the game.

- Secondly, bots collect items of value, when these items make their way into the game's economy, they will influence it greatly. Once valuable items that would have given the player great wealth are now distributed cheaply because the bots *farm* these items constantly (farming is a term used for the behavior of slaying monsters for the sole purpose of getting specific items of value).
- Thirdly, there are people who use these bots to collect items and sell them on for real-life currency. This is a big problem for games where the items equal the strength of the character. When people buy items their characters will become stronger (more quickly) than the characters of people who do not buy their items. This in turn means that, if the players want to keep up with the buyers, they will have to buy items which completes the vicious circle.

2 Background

One way of dealing with these problems is to detect bots and ban the people using them from the game. A main requirement for any detection method is to have no *false positives*, that is, no legitimate players are classified as a bot. If you wrongly classify humans as bots, you will accuse (and most likely punish) people wrongfully, and thereby harm your customer relations. A low number of false negatives is a desirable property as well, because this means many of the bot players are detected. One of the simpler methods of bot detection is the implementation of a CAPTCHA (Completely Automated Public Turing-test to tell Computers and Humans Apart; as seen in figure 1).

Figure 1: CAPTCHA example [12]

The CAPTCHA's (as introduced by von Ahn, Blum, Hopper and Langford [1]) provide a good way of detecting automated programs, since they are specifically designed to be easily solvable by humans but very difficult for computers. However the method of presenting this test does not lend itself to a fast-paced game such as (most of the) MMORPGs. The user has to stop playing the game to fill in a CAPTCHA in order to continue playing. (The CAPTCHA's and their applicability are addressed in more detail in Golle et al. [7], here it suffices to say that the users of games do not wish to be interrupted to fill in a CAPTCHA).

Another way of detecting bots is to analyze the network traffic. This is researched by Chen et al. for the MMORPG Ragnarök Online:

> *"To address this problem, we analyze the traffic generated by human players versus game bots and propose solutions to automatically identify game bots." [3]*

The conservative approach of the proposed integrated schemes (that gives zero false positives) produces a 90% correct decision rate, given an input size of 10,000 packets. The progressive approach yields a false negative rate of less than 1% and achieves a 95% correct decision rate, given an input size of 2,000 packets. The research looks promising with a 90% classification rate (without false negatives) however there was only a dataset of 4 human players and a small variation in network configuration. The results that were found may not hold for people who have different play styles than the people who were tested. Also Chen et al. argue that *"...the burstiness trend scheme is immune to random-delay attacks" [3]*, however the delays that could be built into the system need not be random. There could well be a systematicity or regularity to the human behavior and how it differs from bot behavior. And if there is such systematic relation between the packets coming in and the amount of time it takes for a human player to respond, the bot-maker could implement similar timings into their bots decision-making process.

Another genre of games in which there has been done research into bot detection are FPS (first person shooters). Chen et al. [4] have explored the possibility to classify bots and humans by using the avatar trajectory. The four main features used in this research are given here in abbreviated form (for a more extensive explanation see [4])

1. ON/OFF activity, the amount of time a player stays in one place (OFF) and the amount of activity (ON).
2. Pace, the speed at which a player walks during the game. Several speeds are possible, such as running, walking, etc.
3. Path,
   - Lingering, *"For an avatar at (x, y) at time t, if its distance from (x, y) was always less than d during the period $(t, t+p)$, we say that the avatar was lingering during $(t, t+p)$, given the parameters (d, p)."'* [4]
   - Smoothness, *"...determines whether an avatar moves in straight or zig-zag patterns."*[4]
   - Detour, measures the effectiveness of the movement. *"If an avatar is at (x_1, y_1) at time t_1 and at (x_2, y_2) at time t_2, we compute the detour by dividing the length of the movement by the effective offset of an avatar during the period (t_1, t_2)."*[4]
4. Turn, the change from one direction to the next direction. Angles 30, 60 and 90 degrees are used, as is the sum of all the degrees that are greater than 30 degrees.

By using these features for classification Chen et al. managed to get a classification rate of 98% or higher. We do not know whether the features will be difficult to counter for the bot-makers, as this is almost as important as the classification itself (if bot-makers are able to counter everything fairly easy, it will not be worth the effort to implement these findings into a working system). ON/OFF activity and Pace can easily be monitored by the bot itself, and thus can be countered with minimal effort (e.g. if a bot is too slow, let him run more). The turn feature itself is not very complex, only looking at specific angles, this could also be countered by letting the bot monitor itself and act accordingly. The path feature is harder to counter. It looks at the result of behavior, namely the movement patterns. Humans walk in a different way from bots, to reproduce similar walking behavior a more complex A.I. is needed to steer the bots.

Another research that has been done into bot recognition is by Cornelissen et al. [6]. In this research the focus lies on detecting bots in an MMORPG (Ragnarök Online). The 7 features used in this research are:

1. Number of packets per second
2. Average time between 'Move' packets
3. Average distance between 'Move' packets
4. Average time between 'Take Item' packets
5. Average time between 'Chance Direction' and 'Take Item' packets
6. Average time between a kill and the first following 'Take Item' packet
7. Number of attacks per second

By using these features Cornelissen et al. have managed to get a classification rate of 92% [6].

However we believe that many of the features used in this research are relatively easy to counter for bot-makers. Every feature that uses time can be influenced by random (or chosen) time intervals between certain actions. The distance between 'Move' packets can also be influenced, albeit harder (since it will require more complex changes to the programme versus only the addition of a timer).

In our research we will try and find a more robust (i.e. harder to counter for bot-makers) feature for classification. We know that most bot programs have a very simple A.I. (there is no need for a complicated system if a simple system will suffice). We believe that we can differentiate between bots from humans based on the complexity of their behavior. We chose to use a feature that is directly related to the behavior, namely the movement of the players. In figure 2 and 3 you can see the movement of a human and bot respectively in the virtual world. These two figures illustrate the difference in movement. Humans walk in different patterns than bots. Our hypothesis is that we can differentiate bots and humans based on the difference in their movement patterns. In the data analysis and results section, we will explain more about the features used in the classification. Similar movement based research has been done by Chen et al. [4, 5], however our research will take a different approach and will utilise a different type of game (MMORPGs versus FPS).

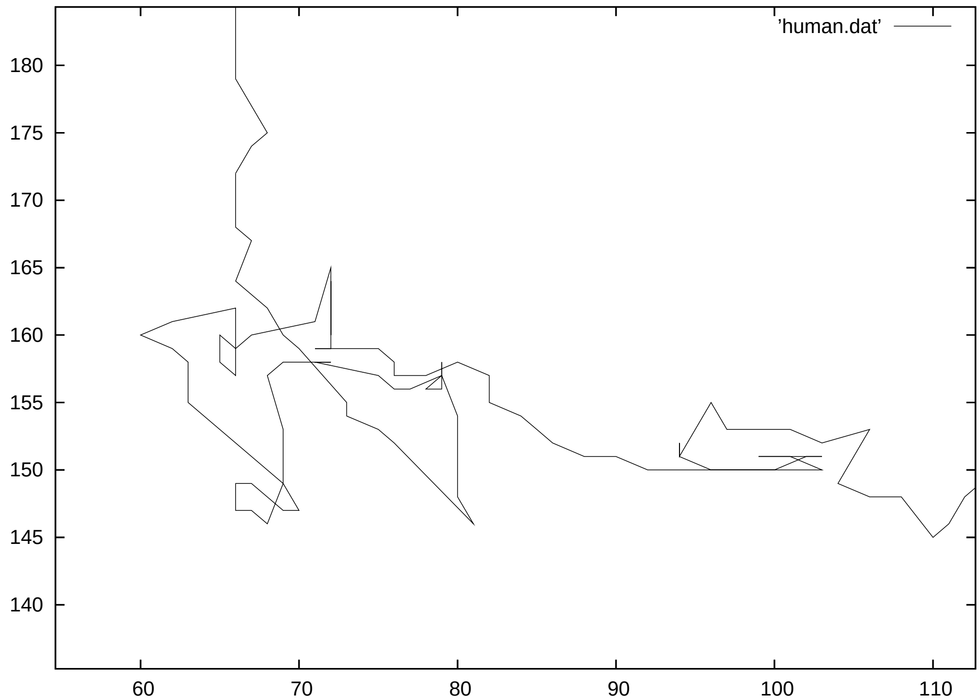

Figure 2: Path of a human through the virtual world

3 Method

3.1 Participants

The participants of this research were 32 human players (of different jobs/levels), of which 7 failed to meet the requirements of the research, and 25 bot players using Openkore (version 2.0.6) [11]. Players were arbitrarily chosen, from a group of regular players of the game, and asked if they wanted to participate in the research.

3.2 Materials

Participants used a personal computer, no special setup was used to collect these data. A simple personal computer was used to run the Openkore program of which none of the default settings were changed. The game Ragnarök Online was used in the experiment.

3.3 Procedure

The participants were asked to play for 10 minutes. They were instructed to play the game as they would normally do. The players were informed that their session was recorded. Of the 32 participants 7 did not play for the required 10 minutes and were therefore discarded from the data (the required 10 minutes of playtime is an arbitrarily chosen time span to ensure that the collected data is representative of the player). Twenty-five Openkore bots were configured (default settings) and observed in different locations fighting different monsters. The bots all represented a different character (jobs) whose location from the start differed. The 10 minute long sessions of the 25 bot players were recorded. For both the human and bot players the data recorded consists of packets and timestamps, which are explained in the measurements section in more detail.

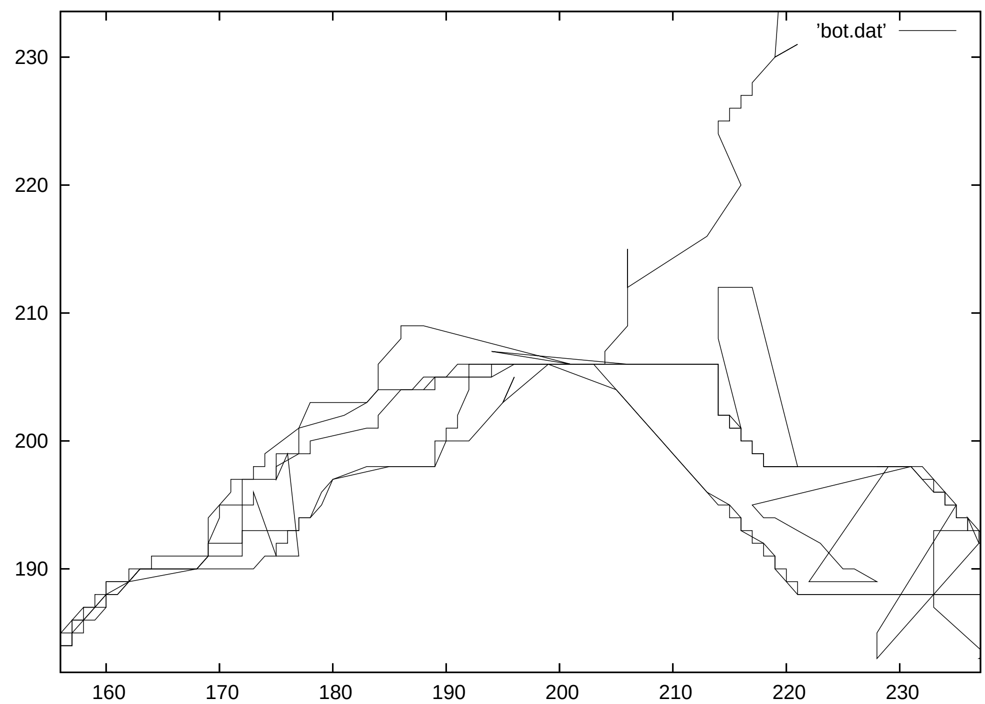

Figure 3: Path of a bot through the virtual world

3.4 Measurements

The actions of the players were recorded by writing packets, that were sent from the client to the server, to a file. These packets contain all the information about the actions of the player. Before a packet was written to the file a time stamp was saved indicating the time at which the packet was saved. Table 1 illustrates the packets that were selected for the experiment. The data we have used in this research is the data that was collected by Cornelissen et al.[6]

Packet	**Information**
MakeConnection	Starts a new session
MoveToXY	Moves the player to location (X,Y)

Table 1: Selected Packets

4 Results

Decoding of the MoveToXY-packet for every human and bot player resulted in 50 lists with a varying amount of (x,y)-coordinates. These lists represent the path that a human or bot had walked which are partially illustrated in figure 2 & 3 for one human and bot player. For all coordinates, except the first and last in the list, the angle of the movement was computed using the cosine rule. Angles were rounded up to represent an integer value between 0 and 180 degrees. Here 180 degrees should be interpreted as walking straight ahead and 0 degrees as a complete turn. The frequency with which angles occurred were added for all human and bot players respectively and plotted in a histogram (see figure 4). The differences in the distributions, which are shown clearly in the histograms, will be used to discriminate humans from bots.

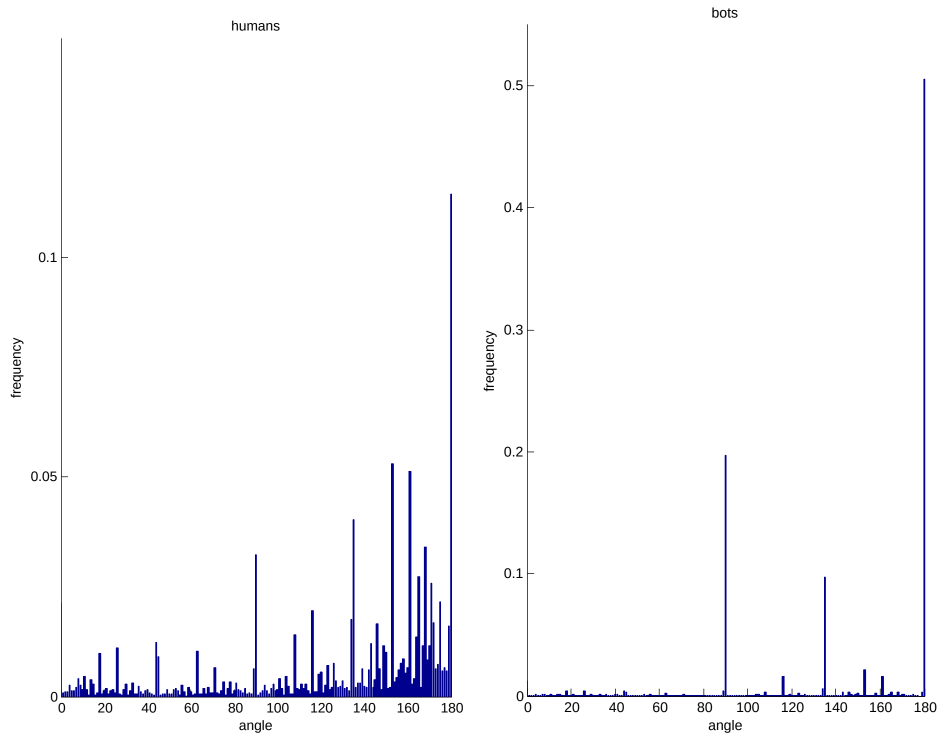

Figure 4: Average angle distribution of humans and bots

In order to distinguish humans and bots we will calculate the prototypical (average) human h (and bot b). We define h (and b) using the normalised accumulated data for each angle α:

$$h_\alpha = \frac{\text{raw\_freq}(\alpha)}{\sum_{\beta=0}^{180} \text{raw\_freq}(\beta)} \tag{1}$$

where raw_freq is the total number of occurrences of a certain angle in either humans or bots. The result of normalization is two distributions, each with surface of 1. The normalized values represent the average probability of an angle to occur in human and bot players. For instance the average probability of a bot walking 180 degrees is larger than 0.5, whereas for humans this chance is just above 0.1[1].

Assume we have a unknown player x with normalized values x_α. The nearest centroid classification method [9] is used in the following way. The distance $\Delta(x, h)$ between x and the prototypical human h is calculated as follows:

$$\Delta(x, h) = \sum_{\alpha=0}^{180} (x_\alpha - h_\alpha)^2 \tag{2}$$

Likewise, the distance $\Delta(x, b)$ between x and the prototypical bot b is:

$$\Delta(x, b) = \sum_{\alpha=0}^{180} (x_\alpha - b_\alpha)^2 \tag{3}$$

To classify the unknown player, we simply compare distances:

if $\Delta(x, h) \leq \Delta(x, b)$
  then data x is classified as a human player
  else data x is classified as a bot player
fi

To further improve the discrimination, angles that are not significantly different when comparing bot to human data (t-test with $p < .05$ for each angle) were removed and only relevant angles remained. The

[1] Note that this example holds for the *average* bot, identification by just inspecting the value for 180 degrees alone was not possible

removal is justified because such an angle will not help to distinguish bots from human players in the classification.

Because of the small dataset, leave-one-out cross-validation was used. That is, all 25 bots and 25 humans were classified once. Whenever a human path was tested, its data was not used to when calculating the prototypical human h (and vice versa for bots). A 100% correct classification rate was obtained using this method.

5 Discussion

Our findings suggest that the differences in the distributions of angles to be a valuable feature to distinguish between human players and OpenKore bots in Ragnarök Online. The one hundred percent classification rate means there were no false positives (i.e. no humans recognised as bots). If a player, who pays to enjoy the game, is wrongfully accused of being a bot, this will have a great impact on the reputation of the game. These results also means that the method could be implementable into a real life system for detecting bots in MMORPGs, since we can classify bots with a great deal of certainty. However we feel that there needs to be more extensive testing done with a larger data-set before these claims can be made.

The single strongest point of our research is the 100% classification. This is an improvement on Cornelissen et al. who managed a classification rate of 92% with the same dataset. Furthermore, the method presented here uses an explainable algorithm, opposed to the black box neural network used by Cornelissen et al. It would also be interesting to see how our methods will fare against the features proposed by Chen et al. (for FPS games). Another good point is the robustness, we believe that our methods will make it very difficult for the bot-makers, because we look at all the possible angles rather than only at specific angles.

A weak point of our research is that we have only a limited amount of data on which we could test our hypothesis. Another weak point, that is a direct cause of the first one, is that we could not find any reliable dependency data. We believe that if we take into account the fact that humans walk different angles, depending on the last angles that were walked, we could produce a more robust set of features. Bot-makers can relatively easily counter our proposed features by walking some angles that it would not normally walk (i.e. it can measure the angles it walks itself and walk a few arbitrarily chosen angles to make the histogram look like that of a human, this would take some time, but will prevent our classifier from recognising it is dealing with a bot). Although this would work, it would also deteriorate the effectiveness of the bot. The bot will no longer be able to choose where it is going based on the quickest way to kill monsters, but it is limited to specific angles it needs to walk in order to be not detected. Another way would be to improve the bots Artificial Intelligence in order to give it a more human-like nature, however this is not easily implemented. The reason that the bots walk certain angles more than humans is likely because of the length of movement. Bots generally click very close to the player, whereas human players tend to click further away. The virtual world, in this game, is built like a chess board, with specific coordinates you can click on (rather than a more continuous distribution of the possible coordinates you can walk to). This means that bots (who click close to themselves) are more likely to make their avatar walk in angles of 90, 135 or 180 degrees. We believe that, despite this property, our features will still be difficult to counter. Since our classifier looks at each angle, so the distribution of the angle has to be very similar to those of the humans.

As already mentioned in the conclusion section, our research has provided us with a very good way of detecting bots in Ragnarök Online. We believe that our results are applicable and usable by companies who offer MMORPGs. Firstly because the methods used are relatively easy, the calculations can be done in real time while the people are playing the game. Secondly we can also say, with a great deal of certainty, if someone is a bot, or human. This will allow the companies to act on the results found (i.e. there is no chance that a human will be wrongfully accused of being a bot).

For future research more games and their different bots should be assessed. Although OpenKore claims to have a market share of 95% on Ragnarok Online bots, there are more bot programs which might use different protocols for walking around. Other Ragnarök Online bots need to be evaluated in order to conclude this method is usable for detecting any bot in the game.

Acknowledgments

We would like to thank M. Sadakata and A. Brandmeyer for their support during the research and the writing of this paper.

References

[1] L. von Ahn, M. Blum, N.J. Hopper, and J. Langford. CAPTCHA: Using hard AI problems for security. In *Proceedings of Eurocrypt*, pages 294–311, 2003.

[2] Blizzard Entertainment. World of warcraft subscriber base reaches 11.5 million worldwide, 2008. `http://www.blizzard.com/us/press/081121.html`: Last accessed 06/19/2009.

[3] K.-T. Chen, J.-W. Jiang, P. Huang, H.-H. Chu, C.-L. Lei, and W.-C. Chen. Identifying MMORPG bots: A traffic analysis approach. In *Proceedings of ACM SIGCHI ACE'06*, 2006.

[4] K.-T. Chen, A. Liao, H.-K. K. Pao, and H.-H. Chu. Game bot detection based on avatar trajectory. In *Proceedings of the 7th International Conference on Entertainment Computing*, 2008.

[5] K.-T. Chen, H.-K. K. Pao, and H.-C. Chang. Game bot identification based on manifold learning. In *Proceedings of ACM NetGames 2008*, 2008.

[6] A. Cornelissen and F. Grootjen. A modern turing test: Bot detection in MMORPGs. In A. Nijholt, M. Pantic, M. Poel, and H. Hondorp, editors, *Proceedings of the 20th Belgian-Dutch Conference on Artificial Intelligence (BNAIC2008)*, pages 49–55, 2008.

[7] P. Golle and N. Ducheneaut. Preventing bots from playing online games. *Computers in Entertainment (CIE)*, 3(3):3–3, 2005.

[8] Gravity. Ragnarök online. `http://www.ragnarokonline.com/`: Last accessed 06/19/2009.

[9] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[10] Media analyst Screen Digest. There's life beyond world of warcraft, 2009, Last accessed 06/19/2009. `http://www.screendigest.com/press/releases/pdf/PR-LifeBeyondWorldOfWarcraft-240309.pdf`.

[11] OpenKore. The OpenKore Project (Ragnarök Online Bot). `http://www.openkore.com/`: Last accessed 06/19/2009.

[12] Wikipedia-users. Wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/CAPTCHA/`: Last accessed 06/19/2009.