Programme: Computing

Module: CPU5000

Module Title: Level 2 Project

Assignment Title: Report and Reflective Literature Survey

Report Title: Determining Social Media Randomness Through Virtual Robots

Student Identification Number: 1500936

Word Count: 2583

Tutor Name: Amanda Dewhurst, Louise Ashby

Date of Submission: 16/03/17

Contents

Introduction

The following report outlines the aims and objects of my project and research topic, along with a descriptive summary of the professional issues addressed by it. It reviews the four main, relevant aspects and discusses any implications they may have and why they are necessary to complete the project. It contains any questions raised and explains how my study intends to answer them, along with any technical and professional skills that may be required in order achieve my objective.

Research Aims

My chosen research topic is the study of randomness and how random strings of text can be parsed to autonomously control a collection of virtual robots in the game Robocode. I intend to produce lots of data and visualise it in graphical format so that I can analyse the correlation of data and determine the randomicity of the robot's actions.

The game is written in the programming language 'Java' and is packaged with its own editor to create your own robots. Although you can customise the robot in your own way, the fundamental controls are always required. I intend to replace these controls values with randomly generated ones to randomise the robots' function. The source of my random values will be from Twitter, a social media feed. The 'tweets' will be converted from a string of text into numerical values that will dictate the robot characteristics such as movement speed, direction, radar scanning, fire rate and power etc. I will pitch various Twitter accounts against one another and record the results. I will then analyse the data for patterns, associations and trends to see if the generated values were truly random or pseudo-random. Questions I wish to answer through this project are;

- "Can a social media feed autonomously control virtual robots and pitch them against one-another in battle?"
- "Are social media feed posts truly random, or are there any patterns that suggest a pseudo-random nature?"

I intend to answer these questions through my research and practical application.

Professional Aims

My project is centred around the programming game Robocode and the social media website, Twitter. Given the projects technical nature, there are several skills I need to develop and advance in order to produce my intended result.

Firstly, knowledge of either the Java Programming Language or .NET Platform is paramount as these are the 2 methods in which you can create and modify the virtual robots in Robocode. I will be pursuing the Java route as this is the language I am most familiar with through practice in my other University modules.

Another, similar skill is data visualisation. I intend to produce graphs and charts that represent the various statistics that are produced at the end of a match. These graphs will then be used to analyse the data, and spot patterns and trends that will ultimately aid the conclusion of the project. I will be using a JavaScript Library called D3.js (Data Driven

Documents) which I have also previously used in another of my University modules. I will need to investigate how Java and D3.js can collaborate to automatically produce these graphs after each fight by storing the data in a database before passing it to the JavaScript. This will require knowledge of several languages, both client and server side. Such languages include; HTML5; CSS3; JavaScript; PHP; MySQL; JSON and Java8.

Furthermore, mathematical knowledge involving graphs and probability will be required in order to write the correct code to plot the produced data set and study the randomness of the Twitter tweets. Therefore, proper research into data correlation, linear regression, statistical probability and more will be required.

Literature Review

Introduction
The following literature survey concisely outlines the four main sub-research areas of my project, discusses each source and comments on any implications it has on the project. It should be noted that due to the strong technical nature and lack of historic documentation of my chosen research areas, there is not a great amount of academic material such as books and journals to base my research on. Therefore, I have chosen to cite and reference other credible sources such as websites in my project research.

Research Area 1 – Twitch Plays Pokémon
Twitch is the world's leading social video platform for streaming gaming oriented content around the world. It was founded in June 2011 and currently amasses nearly 10 million visitors every day. The site is popular among both veteran and aspiring YouTubers and attracts the attention of big e-sports players and tournaments. (Twitch, 2017)

Twitch Plays Pokémon (TPP) is a social experiment created by an anonymous Australian programmer and involves streaming a game of Pokémon on Twitch in the Game Boy emulator 'VisualBoyAdvance'. The anonymous programmer developed an Internet Relay Chat (IRC) bot, written in Python, which captures messages from the Twitch chat and parses those strings directly into commands for the game. The stream soon became popular and reached approximately 60,000 concurrent playing users. It took approximately 16 continuous days to complete the first game 'Pokémon Red'. Since then, numerous other Pokémon games from the franchise have been hosted and streamed on Twitch for the users to collaborate on. (Wikipedia, n.d.) (Cunningham, 2014)

Twitch Plays Pokémon had a big influence on the underlying premise of my project and provided me with the inspiration to integrate the social media platform 'Twitter' with the Java Programming game 'Robocode'. One major difference, however, is the manner in which the 'random' items and moves were generated. TTP used a Pseudorandom Number Generator (PRNG) in order to predetermine the Pokémon, items and moves in some of the games. I will be using the randomicity of tweets from certain Twitter accounts to generate the instructions for the robots.

Research Area 2 – Robocode & Java

Robocode is a programming game that can be used to teach or learn Java or .NET. It can also be a gateway to explore AI and machine learning techniques along with problem-based learning. The game consists of an arena in which 2 or more virtual robots are pitched against one another in battle, as seen in Appendix 2. The three basic components of any robot are its radar, movement and targeting. The Robocode API documentation contains all the classes and methods that can be utilised by the programmer. For instance, the radar can be used to scan the environment for other robots and act upon finding one. The movement can be set to be random, move in a set pattern or move based on other in-game events. Also, the turret can be moved independently in order to target other robots irrespective of the robots heading direction. There are many other events such as 'On-Hit-By-Bullet' that can be configured to give your robot the edge over its opponent. (Anon., 2003)

When creating a new game in Robocode, there are lots of configurable options. Such as; the robots who will be fighting; the number of rounds; battlefield size; gun cooling rate etc. After the set number of rounds have been completed, a table is produced listing all the robots who contended along with statistical information about the game. Such information includes; Rank; Robot Name; Total Score; Survival; Survival Bonus; Bullet Damage; Ram Damage x 2; Ram Bonus; No. of 1st s; No. of 2nd s and No. of 3rd s. Please see Appendix 3 for reference. Further information regarding the game results is discussed in the final section on D3.js. (Nelson, 2000)

Robocode plays a major role in my project as it is the entity that provides all the data that will ultimately produce the answers to the questions raised. However, I will not be using Robocode in the conventional manner. Instead of creating robots manually via the use of the Java Code Editor, I will be producing robots 'randomly' by taking commands from an external source. This source is 'Twitter'. Twitter is a huge social media site in which users can follow other users whose content they wish to see on their feed. 'Tweets' can be used to start or a join a conversion with anyone in the world. (Twitter, 2017)

How is Twitter going to control a virtual robot? Each virtual robot on the battlefield will be created from one Twitter Account. The program will connect to the Twitter API using a 3rd party Java Library called 'Twitter4J'. This library will allow me to download tweets from a given account and store them into a database as text strings. These strings can then be broken down and parsed into numerical values that will be distributed among the robots' control code.

Research Area 3 – Randomness vs Pseudo Randomness

When we think of randomness, most people think of a coin flip or a dice roll, but what does 'Random' *actually* mean? We can define randomness as something that is un-predictable and contains no recognisable patterns. (Vsauce, 2014)

As previously mentioned, Twitch Plays Pokémon made use of a Pseudo-Random Number Generator (PRNG). As you might expect, the generated numbers are not truly random as they use pre-defined tables of values that are picked 'randomly' by a mathematical formula. This means that given enough time, patterns will start to emerge in the generated values. (Haahr, n.d.)

Appendix 1 shows the difference between two bitmaps. The left one is generated by Random.org's True-Random Number Generator (TRNG), while the right is generated by the PRNG rand() function from the PHP API. You can clearly see the pattern in the right bitmap image that has emerged due to the pseudo-random nature of the algorithm inside the rand() function. Conversely, the left image appears to be somewhat uniformly distributed as no clear pattern can be immediately distinguished. This is because Random.org have created a true random number service that generates randomness via atmospheric noise. (random.org, 2017)

Given the clear differences between PRNGs and TRNGs, I would like to analyse the random nature of the text strings that are produced by Twitter feeds to determine where the lie on scale between pseudo-random and truly random. One possible bias towards the randomness would be the frequency in which specific letters appear in the English language. This in-uniformity is caused by a great number of relatively small common English words that are more likely to appear in the tweets on the Twitter feed. It is also possible that certain Twitter accounts see certain words appearing more often than others, meaning that the constituent letters in these words would introduce another similar bias into the system. It would also be interesting to see the difference in languages from accounts whose users are not English. (Students, 2017)

Research Area 4 – Data Visualisation with Data Driven Documents (D3.JS)

D3.js is a JavaScript library created by Mike Bostock. It is a powerful tool that can produce a myriad of data graphical representations by bringing data to life using several web programming languages such as HTML, SVG and CSS. It seamlessly integrates with modern browsers without the need for a proprietary framework and allows easy manipulation of the DOM (Document Object Model).

Setting up a D3 development environment simply requires you to either download the D3.js library from d3js.org or to link a copy of the library hosted on a CDN (Content Developer Network) Server. I am familiar with the library as I have previously used it in my Web Programming module at University.

I will be using D3.js to automatically produce a series of graphs representing the match data after a battle in Robocode. The data will be stored in a database, converted to a readable format such as .JSON, .CSV or .TSV and consumed by the D3 code in order to produce the graphs on a webpage. I will then be able to analyse the graphs to spot any correlation and decide if the data is sparse or sporadic enough to qualify as being pseudo random or if it is uniformly distributed to be truly random.

Conclusion

To summarise, I feel that the research I have conducted has put the project into perspective and has given me the knowledge that it is a manageable, yet somewhat respectable endeavour. It has also helped me identify which areas will need the most time and research, and those that will need more technical ability. Moreover, it has also made it clear which skills of my own need improving in order to overcome certain obstacles.

Overall, it is clear that a lot of research will need to be conducted to gain the knowledge and understanding required to complete the study and produce my intended results. However, the aforementioned research into my project aims has not only consolidated my interest in the topic, but reinforced the weight that this project carries.

For the next step, I intend to research into the 'Twitter4J' Java Library and the Twitter API and also familiarise myself with the relevant Java Classes & Packages from the Java API that are responsible for connecting a Java Program to a MySQL Database. I will look into integrating the Java Program's Database with the D3 JavaScript Library for the final data visualisation step. Finally, I will revise my A-Level Mathematics Notes for information on Statistical Linear Regression, Advanced Probability and any other relevant topics that will help with the graphical analysis.

References

Anon., 2003. *RoboWiki.* [Online]
Available at: http://robowiki.net/
[Accessed 12 03 2017].

Cunningham, A., 2014. *The bizarre, mind-numbing, mesmerizing beauty of "Twitch Plays Pokémon".* [Online]
Available at: https://arstechnica.com/gaming/2014/02/the-bizarre-mind-numbing-mesmerizing-beauty-of-twitch-plays-pokemon/
[Accessed 12 03 2017].

Haahr, D. M., n.d. *Introduction to Randomness and Random Numbers.* [Online]
Available at: https://www.random.org/randomness/
[Accessed 12 03 2017].

KhanAcademy, n.d. *Pseudorandom number generators.* [Online]
Available at: https://www.khanacademy.org/computing/computer-science/cryptography/crypt/v/random-vs-pseudorandom-number-generators
[Accessed 12 03 2017].

Kobayashi, K., Uchida, Y. & Watanabe, K., 2003. *A study of battle strategy for the Robocode,* s.l.: s.n.

Nelson, M. A., 2000. *Robocode,* s.l.: s.n.

O'Mara, M., 2014. *Twitch Plays Pokémon a wild experiment in crowd sourced gameplay.* [Online]
Available at: http://business.financialpost.com/fp-tech-desk/post-arcade/twitch-plays-pokemon-a-wild-experiment-in-crowd-sourced-gameplay?__lsa=e52a-6474
[Accessed 12 03 2017].

Ramirez, D., n.d. *Twitch Plays Pokemon: A Case Study in Big G Games.* [Online]
Available at: http://library.med.utah.edu/e-channel/wp-content/uploads/2016/04/digra2014_submission_127.pdf
[Accessed 05 03 2017].

random.org, 2017. *Statistical Analysis.* [Online]
Available at: https://www.random.org/analysis/
[Accessed 12 03 2017].

Robocode, 2017. *Robocode 1.9.2.0 API.* [Online]
Available at: http://robocode.sourceforge.net/docs/robocode/
[Accessed 13 03 2017].

Simone Heidbrink, T. K., 2016. Religion in Digial Games Respawned. *Heidelberg Journal of Religions on the Internet,* 10(1), pp. 74-95.

Students, E. f., 2017. *Frequency of Letters.* [Online]
Available at: http://www.english-for-students.com/Frequency-of-Letters.html
[Accessed 12 03 2017].

Twitch, 2017. *About.* [Online]
Available at: https://www.twitch.tv/p/about
[Accessed 08 03 2017].

Twitter, 2017. *Getting started with Twitter.* [Online]
Available at: https://support.twitter.com/articles/215585
[Accessed 13 03 2017].

Twitter4J, 2017. *Twitter4J.* [Online]
Available at: http://twitter4j.org/en/
[Accessed 14 03 2017].

Vsauce, 2014. *What is Random?.* [Online]
Available at: https://www.youtube.com/watch?v=9rIy0xY99a0
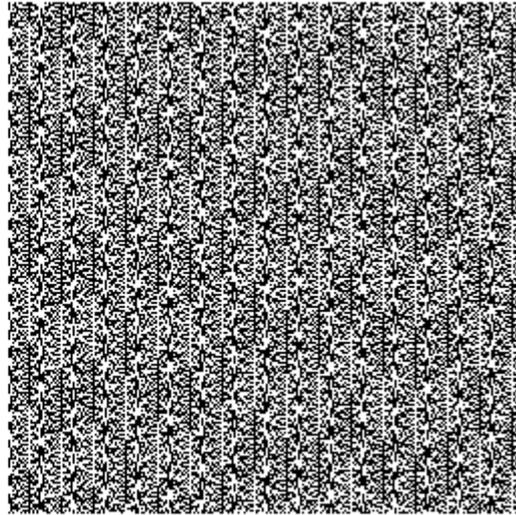[Accessed 12 03 2017].

Wikipedia, n.d. *Twitch Plays Pokémon.* [Online]
Available at: https://en.wikipedia.org/wiki/Twitch_Plays_Pok%C3%A9mon
[Accessed 05 03 2017].

Appendices

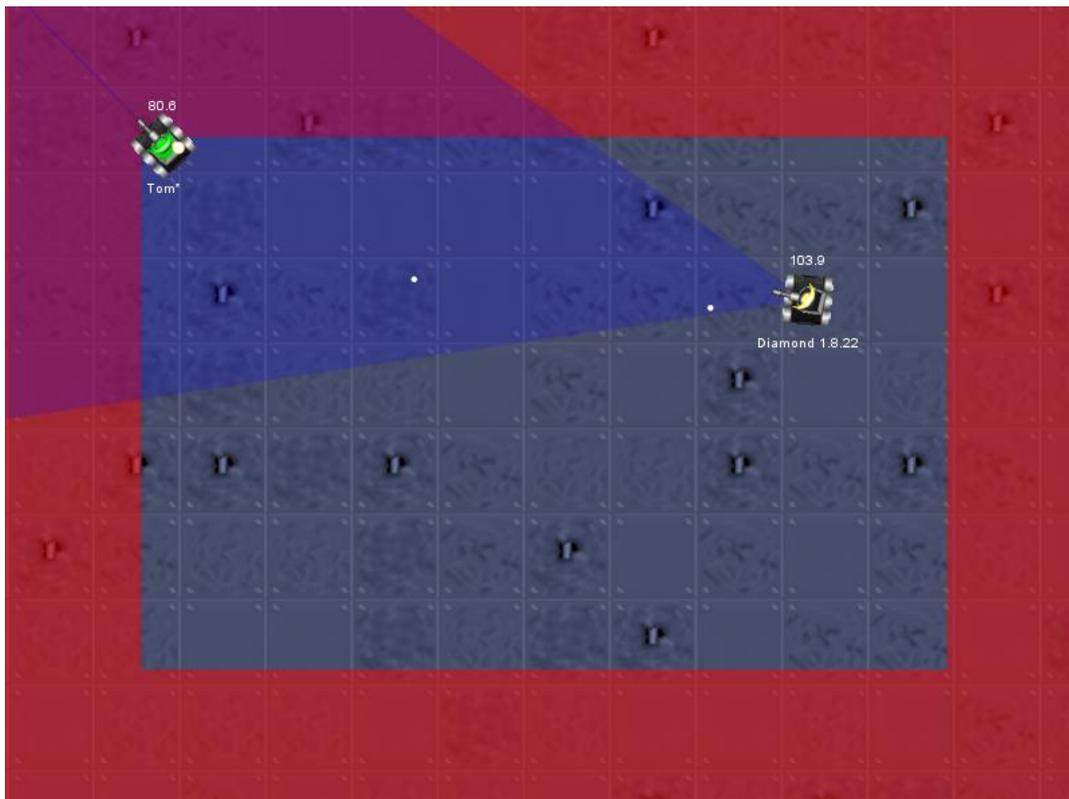## Appendix 1 – True Random Bitmap vs Pseudo-Random Bitmap Image
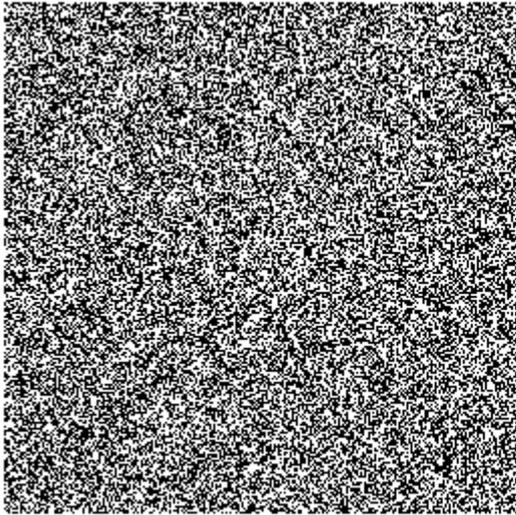
RANDOM.ORG   PHP rand() on Microsoft Windows

Appendix 2 – Screenshot of Robocode Battle

Appendix 3 – Robocode Match Results

Rank	Robot Name	Total Score	Survival	Surv Bonus	Bullet Dmg	Bullet Bonus	Ram Dmg * 2	Ram Bonus	1sts	2nds	3rds
1st	voidious.Diamond 1.8...	1741 (100%)	500	100	933	189	20	0	10	0	0
2nd	tp.Tom*	5 (0%)	0	0	4	0	1	0	0	10	0

Results for 10 rounds

Save   OK