Final Report

This document outlines the general guidelines for writing the final report for students completing the first semester of the two course design project/Honors thesis course series. The students should discuss these general guidelines with their advisor and come up with a specific plan for their individual project. You should aim for a 20-page report.

1) Title Page: The title page must include:

a. The title of the project (note that the convention in English is for all words in the

title to be capitalized, except for articles and prepositions).

b. The names and student numbers of the authors of the report (with course number

for each).

c. The name of the project supervisor.

d. The name of any company involved in the project if applicable.

2) Abstract: The abstract is an executive summary that is maximum one page long. It should outline the overall motivation of the work, what where the goals of the project and what was achieved. This section should be as short as possible while conveying the necessary information.

* Objective
  + Dashboard
  + Really explain the objective of the different graphs
  + Live data support
* Purpose
  + Visual representation, interaction, ease of identification
  + What the different graphs aim to do
* What was achieved
  + Graphs implemented

3) Acknowledgment section: If applicable you may acknowledge the contributions of those who contributed to the project but are not on the author list.

We would like to express our special thanks to Professor Aditya Mahajan, our research supervisor for his advice and encouragement during the course of this project.

4) Table of contents.

5) Lists of tables.

6) List of figures.

7) List of abbreviations used in the report.

* CSV Comma Separated Values
* JS JavaScript
* JSON JavaScript Object Notation
* MAB Multi-Armed Bandit
* UCB Upper Confidence Bound
* WO Website Optimization

8) Introduction/Motivation/Objective Chapter: In this section you should introduce the

overall theme of the project, and state clearly what are the goals you are trying to achieve.

This section should also clearly convey why this project is important, what is the potential impact (applications, etc).

With the increasing widespread of e-commerce and e-marketing, the need for optimized websites is also increasing. WOrefers to the process of affecting the visibility of a [website](http://en.wikipedia.org/wiki/Website) in order to provide it with a good rank in [search engines](http://www.webopedia.com/TERM/s/search_engine.html) as well as to make it load faster[1]. Our project arises from the need to find a more efficient way[2] to optimize websites than the currently used statistical hypothesis based classical *A/B testing*\*. A more time, cost and memory efficient algorithm will lead us to designing better web layouts and in turn: websites, leading to increased user engagement and hence higher business prospects. We make use of MAB algorithms to find the most optimal website layout and thereby optimize a website.

\*”A/B testing (sometimes called split testing) is comparing two versions of a web page to see which one performs better” [3].

There are two primary deliverables for this project that have been divided amongst two teams as follows:

1. Framework

* Implements the multi-armed bandit algorithm
* Determines which version of a particular website to display
* Keeps track of the effectiveness of each version, logging the flow of user actions and any other information that may be of interest

1. Dashboard

* Represents the results of executing a set of multi-armed bandit algorithms used for website optimization

Our group is working on deliverable number 2, i.e. building the dashboard.

Below is a description of the goals that we are trying to achieve by building this dashboard. We want to provide the user with an interface with the following abilities

1. Uploading a file to generate bar or line charts
   * The file is generated by the framework (deliverable number 1) and it contains results of a MAB simulation
   * Most basic file contains information at each clock tick (website version or arm selected and the resulting success achieved)
   * Multiple formats supported: CSV, JSON or Tabular
2. Providing live data to generate bar or line charts
   * Automatically updating graphs of real time data
3. Viewing statistics for each arm from the bar or line charts generated by file upload and or live data, such as
   * Success at any particular instant (instantaneous)- Bar chart
   * Overall success over time (average)- Line chart
4. Viewing arm details from a line chart
   * Time when a particular arm was active or inactive
5. Viewing results by time from a bar chart
   * Performance of arms at an hourly basis
6. Applying MAB functions on a particular arm and viewing the simulation result in the form of a graph
   * Eg. of functions- UCB, Epsilon Greedy\*\*

9) Background Chapter(s): In this chapter you should summarize the theory and background that you had to learn, and that you believe are necessary for the reader to understand in order to understand the rest of the report.

**Introduction to MAB**

In order to understand MAB algorithms, it is crucial to first familiarize with some terms commonly used in their reference and then the MAB problem itself. Table describes the terms used in every bandit problem.

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| Agent | User |
| Arm | Decision maker |
| Gain | Measure of success or reward |

Table

A MAB problem can be simply stated as follows[4]:

*An agent chooses (plays) an arm and receives a gain from it. How can the agent maximize his gain? Which arms should the agent play, how many times to play each arm and in what order to receive maximum gain?*

A characteristic typical of a MAB problem is the conflict between making decisions that

* Provide high current gains by *exploiting* high performing arms

OR

* Sacrifice current gains by *exploring* other arms in hopes of finding an even-better arm?

This is also known as the *exploration vs. exploitation* dilemma[5].

To summarize, the objective of the agent is to maximize the sum of rewards earned through a sequence of arms.

A MAB algorithm or strategy attempts to solve this problem by looking for *optimal* or *approximately-optimal* solution. Solution in this case refers to a sequence of arms an agent can play in order to receive the maximum gain. It first exploits the highest performing arms. Next, it explores other arms to see if they perform better than the current highest performing ones.

**WO as a MAB Problem**

In order to model WO as a MAB problem, the essential parameters (agent, arm and gain) must be defined. Table describes the MAB parameters in relation to WO.

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| Agent | User |
| Arm | Website version with unique styling  Eg. Each version has a different .css stylesheet demonstrating different   * Colour scheme * Button sizes * Font sizes * Layouts |
| Gain | Effectiveness of a particular version  It can be defined as a metric of success whose definition varies across domains. Eg. Effectiveness can be equal to the   * Number of purchases of a particular item on an e-commerce website such as Amazon.com * Number of donations received on a fundraising website |

Table

10) Design and Implementation Chapter(s): In this section you must describe any design work that was already completed, and outline your detailed design plans for the next semester.

11) Results and tests: In this section you can describe the results of any initial tests and experimentation that you have done in order to make design decisions. Furthermore you can describe how you plan to test your design in order to make sure it meets the desired specifications. Also outline the design decisions that are required in order to make sure that the final product can be tested.

**Initial Tests and Experimentation**

In order to build the dashboard, a suitable charting library had to be chosen. The research began as an exploration[6] of various JS charting libraries such as Radian, Cubism.js, NVD3.js, Rickshaw. Due to the limited time resources at hand, the options were reduced Radian[7][8] and Rickshaw[9]. Sample charts were created in each of these libraries to carefully examine and distinguish the features each had to offer. Tables describe the parameters on which the libraries were examined.

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| **Reliability** | In development phase  Released in 2013 (very new) |
| **Resource**  **Availability** | Well organized tutorial documentation  External resources for Angular.js directives  Untidy and non-intuitive Github repository |
| **Learning**  **Curve** | Knowledge of HTML  Custom HTML elements can represent functional and data plots  Angular.js knowledge for interactive plots |
| **Features and**  **Extensibility** | Limited basic features (covered by Rickshaw) |
| **Size** | radian.min.js = 96.56 KB  No .min version for .css |

Table Radian

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| **Reliability** | Established framework  Released in 2011 |
| **Resource**  **Availability** | Limited and concise tutorial documentation  Comprehensive '/examples' section in Github repository |
| **Learning**  **Curve** | Knowledge of JavaScript for functional, data and interactive plots |
| **Features and**  **Extensibility** | Feature rich  Vast range of extensions to build on and extend existing functionality |
| **Size** | rickshaw.min.js = 63.592 KB  rickshaw.min.css = 5.975 KB |

Table Rickshaw

Based on the comparison of the tables above, Rickshaw was our final choice of charting library. The reasons are summarized below:

* Increased reliability- more established framework
* Enhanced resource availability- comprehensive Github repository with well explained examples
* Neutral learning curve as JS is the common skill between both group members
* Rich feature set along with a wide range of extensions to customize (suitable for our project)
  + Eg. Time fixture feature for incorporating time series graphs
* .min files available both for JS and CSS
  + rickshaw.min.js is ~30 KB smaller than radian.min.js
  + Smaller file will load faster hence consuming less bandwidth to process

12) Impact on Society and the environment: One to two page section where explore the

environmental and social impact of your project. Your analysis should include your work at McGill as part of this project, however, the main focus should be on the product you are designing, the cost/benefit/risk of manufacturing it, the cost/benefit/risk for consumers using it etc. Particular emphasis should be given to:

a) Use of non‐renewable recourses (energy, etc): Consider all stages of the product

from design, manufacturing, distribution, use by consumers, and disposal/recycling.

b) Environmental benefits: Benefits to the enviroment, comparisons with more

polluting technologies etc.

c) Safety and risk.

d) Benefits to society: Quality of life, economic benefits, etc.

13) Report on Teamwork: In this section outline the individual contribution of each member of the group and assess how well you were able to collaborate as a team. Identify any difficulties and outline a plan to address these difficulties in the remainder of the project (during the second semester).

14) Conclusion: In this section you summarize again what was accomplished in this semester, provide a summary of the next steps and share any insight you have learned.

15) References: A list of references that where referred to throughout the text. Note that it is important to refer to each of them at least once in the main text. Please use IEEE format: <http://www.ieee.org/documents/ieeecitationref.pdf>

[1] Kyle James. (2010, June 15). Why Web Optimization is Important For End User and Business Reasons [Online]. Available: <http://doteduguru.com/id5279-why-website-optimization-is-important.html>

[2] A B Testing The Complete Guide [Online]. Available: https://visualwebsiteoptimizer.com/ab-testing/

[3] Chris Stucchio. (2012, June 3).Why Multi-armed Bandit algorithms are superior to A B testing [Online].Available: <http://www.chrisstucchio.com/blog/2012/bandit_algorithms_vs_ab.html>

[4] Cam Davidson Pilon. (2013, April 06). Multi-Armed Bandits [Online]. Available: <http://camdp.com/blogs/multi-armed-bandits>

[5] [Jean-Yves Audibert](http://imagine.enpc.fr/~audibert), [Rémi Munos](http://researchers.lille.inria.fr/~munos/). ICML Tutorial on bandits [Online]. Available: <https://sites.google.com/site/banditstutorial/>

[6] (2013, Nov 11). Javascript Graphs and Charts libraries Comparison tables [Online]. Available: <http://socialcompare.com/en/comparison/javascript-graphs-and-charts-libraries>

[7] (2013, Nov). openbrainsrc Radian [Online] Available: <https://github.com/openbrainsrc/Radian>

[8] (2011). Radian - Home [Online]. Available: <http://openbrainsrc.github.io/Radian/index.html>

[9] (2013, Nov). Rickshaw A JavaScript toolkit for creating interactive time-series graphs [Online]. Available: http://code.shutterstock.com/rickshaw/

16) Appendices: In many cases it is not necessary to have appendices. You may include

appendices for material that do not naturally fit as part of the main report. For example:

a. Long computer code sections that you feel are necessary to include in the report but

would interrupt the flow if included in the main section.

b. User manuals and other documentation for your designs, computer code, etc.

c. Any important information that is generally long and detailed and would interrupt

the flow of the main sections.