National College of Ireland
B.Sc (hons) in Computing (Data Analytics)
2017/2018

Glenn Connell
x14441832

Analysis to determine the user satisfaction regarding content releases within video games with a F2P model.

Technical Report

Glenn Connell x14441832

Declaration Cover Sheet for Project Submission

SECTION 1

Name: Glenn Connell
Student ID: x14441832
Supervisor: Simon Caton

SECTION 2 Confirmation of Authorship

The acceptance of your work is subject to your signature on the following declaration:

I confirm that I have read the College statement on plagiarism (summarized overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature: Glenn Connell

Date: 13/5/18

Glenn Connell x14441832

Table of Contents

Glenn Connell x14441832

Abstract

The problem of the massive influx in text data from the vast variety of web-based sources has been addressed by the recent rapid advances that the field of text mining has experienced. This is largely due to sharp increase in the availability of technology capable of routine interaction with web applications and the creating of data, both structured and unstructured. This project aims to harness some of this data to provide actionable insights into the polarity of user sentiment regarding content releases within the game League of Legends.

Glenn Connell x14441832

1 Introduction

1.1 Background

The computer gaming market is a market in which market growth has regularly occurred, with recent booming periods caused by the mobile gaming market and the increasing popularity of the Free to Play (F2P) model. The F2P model in particular has proven extremely successful as Riot Games flagship game League of Legends has become one of the world's most popular computer games with a peak of over 30 million unique users per month. It goes without saying that the potential profit of such a userbase is immense. The F2P model thrives on an enthusiastic active userbase and as such seeks to retain the largest portion of its userbase as possible. This goal, low customer churn can be achieved utilizing many varied strategies. For the purpose of this project the strategy of frequent content updates will be the focus. Specifically, the project will focus on observing and attempting to predict user sentiment in relation to the content releases.

My Inspiration:

The inspiration of this project is born from the recent uptick in the utilization of the F2P model by game developers, seemingly justified by the explosive success of several recently released titles with a F2P model. An example of which is Fortnite. Fortnite is a relatively new title to the market and yet it has not only surpassed all other direct competitors in profitability, up to 16% of all core computer game players have played the game and it held a massive 12.8% of viewership hours for the month of February on the streaming and video platforms of Twitch & YouTube Gaming respectively. This success has paved the way for game developers to more effectively secure funding for new titles with F2P models resulting in the boom in popularity in titles following the model we see today.[1]

[1]https://resources.newzoo.com/hubfs/Reports/Newzoo_The_Rise_of_the_Battle_Royale_Genre.pdf [Accessed 10/5/18].

1.2 Aims

Aim 1: Come to an understanding of the business side of the project scope and the business goals.

Aim 2: To begin developing the project the first goal is to gather a dataset suited to the proposed question, it must also stand up to certain criteria such as size and time relevance.

Aim 3: The acquired dataset must be cleaned into a more specific and comprehensive dataset e.g. remove unnecessary data from the dataset.

Aim 4: The third aim of the project will be to apply a variety of models in order to obtain one which provides satisfactory results.

Aim 5: Once a suitable model has been chosen the next aim is to evaluate the results of the model in order to ensure all business goals and questions have been reached/achieved.

Aim 6: The final technical task will be visualise the results of this project in such a way to concisely and correctly convey the information in a comprehendible manner for the customer.

1.3 Technologies

RStudio

RStudio will be used to construct this project. RStudio is an open-source integrated development environment for R Language programming, and will be used in conjunction with excel where the datasets will be stored locally and be used to retrieve necessary information. [2]

R Language

The R programming language will be used to build this project, as it has a wide variety of diverse packages available that provides a massive library of functionality.[3]

Python

The other programming language that will be used in this project is Python, a general purpose, versatile language that enable the user to create functionality as well as perform analysis.[4]

Anaconda

The Anaconda distribution for Python is an immensely useful collection of environments that enables flexible and reproducible data science and machine learning.[5]

Spyder IDE

Spyder IDE is a powerful interactive development environment for the Python language, distributed within Anaconda. It provides advanced editing, testing, debugging and introspection features.[6]

SPSS

[2] https://www.rstudio.com/ [Accessed 30 Nov. 2017].
[3] Available at: https://www.r-project.org/ [Accessed 30 Nov. 2017].
[4] https://www.python.org/ [Accessed 30 Nov. 2017].
[5] https://www.anaconda.com/what-is-anaconda/ [Accessed 6 May 2018].
[6] https://pythonhosted.org/spyder/ [Accessed 6 May 2018].

SPSS is a popular statistical package from IBM which can perform highly complex data manipulation and analysis with simple instructions.

The project will make use of SPSS to reinforce the accuracy of the results of the statistics gathered over the course of the project.[7]

Excel

Excel is a spreadsheet tool with built in statistical and graphing commands that allow a user to manipulate information and data loaded into it.

The project will make use of excel to store the data locally in a readable format, as the data will be stored in the .csv file format.[8]

1.4 Structure

Throughout the development of this project, the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology will be followed strictly, which will ensure the project reaches the required standard. While the methodologies Knowledge Discovery in Databases (KDD) and Sample, Explore, Modify, Model and Assess (SEMMA) were considered initially for the purposes of this project, it was determined that CRISP-DM would be more suited to the project as it strives to reach business related goals. This decision while not made lightly, was a clear choice as neither KDD or SEMMA address the business aspect of the process, each focuses more on the modelling aspect of the process. Detailed below is a brief guide to the CRISP DM methodology.

[7]https://www.ibm.com/analytics/data-science/predictive-analytics/spss-statistical-software [Accessed 30 Nov. 2017].
[8] https://en.wikipedia.org/wiki/Microsoft_Excel [Accessed 30 Nov. 2017].

Figure 1 CRISP-DM

- **Business Understanding:** Focuses on comprehending the project objectives and requirements from a business perspective, and then changing this knowledge into a data mining problem.
- **Data Understanding:** Begins with an initial data collection and proceeds with several activities in order to get familiar with the data, to identify quality problems and to discover insights into the data. There is a close link between Business Understanding and Data Understanding as they both require some understanding of the data.
- **Data Preparation:** The data preparation phase covers all initiatives taken to construct the final cleaned dataset from the initial raw data. Tasks for this process include data cleaning, construction of new attributes, and transformation of data for modeling tools etc.
- **Modeling:** In this phase, various modeling techniques are selected and applied in order to find a satisfactory one, some techniques require specific data formats to be used for optimum results. There is a close link between Data Preparation and Modeling as many data problems are discovered while modeling or one gets ideas for constructing new data.

- **Evaluation:** After succeeding in creating a successful model the model must be thoroughly evaluated and you must determine that all important business objectives have been touched on.
- **Deployment:** This is the final stage and involves using your evaluated model results to construct something comprehensible for your client.

Citeseerx.ist.psu.edu. (2017).

1.5 Definitions, Acronyms and Abbreviations

F2P: Free to play.

API: Application Programming Interface.

IDE: Integrated Development Environment.

GUI: Graphical User Interface.

Cleaning: This refers to the processing and refining of data in such a way to enhance its readability and to ensure a greater level of accuracy with any associated statistics.

CSV: Comma Separated Values, a file format readable by the Microsoft Excel tool.

CRISP-DM: Cross Industry Standard Process for Data Mining, a data mining process model that is widely used.

Programming Application: The programming IDE chosen to complete the task for example Rstudio or Spyder IDE.

JSON: JavaScript Object Notation, refers to the lightweight data-interchange format that is easily readable by both humans and machines. Its use of common conventions similar to many of the most popular programming languages, makes it an ideal format for APIs.

DTM: Document Term Matrix, a matrix that describes the frequency of terms within the data.

Tf-idf: Term frequency-inverse document frequency, a weighting factor defined in order to determine how important a word is to a document.

KDD: Knowledge Discovery in Databases, a data mining process model similar in nature to CRIP-DM.

SEMMA: Sample, Explore, Modify and Asses, another widely used data mining process model.

AUC: Area Under the ROC Curve, an evaluation metric used in machine learning.

ROC: Receiver Operating Characteristic curve, a graphical plot used in order to illustrate the ability of a binary classifier as its discrimination threshold is altered.

1.6 Literature Review

As we witness the rise of the technological era the issue of the ever-increasing volume of text data becomes more apparent, to combat this the field of data mining can utilize the very same technology advances that have created the problem to combat it. With these advances in technological capability we see a great surge in versatile and sophisticated techniques.

Sentiment Analysis

Sentiment analysis and opinion mining is the field of study that analyses people's opinions, sentiments, evaluations, attitudes, and emotions from written language. It is one of the most active research areas in natural language processing and is also widely studied in data mining, Web mining, and text mining. In fact, this research has spread outside of computer science to the management sciences and social sciences due to its importance to business and society as a whole. (Bing Liu, 2012) This is put in perspective when one considers the magnitude of the daily output a data from social media alone. A massive collection of opinionated data ripe for utilization within a variety of domains.

The classification of text into predefined categories has received a large increase in notoriety in recent years. In the research community the dominant approach to this problem is based on machine learning techniques. (Sebastiani F, 2002) This approach comes with a multitude of advantages including a decline in labour costs and a great deal of adaptability. It is clear that this approach is not only applicable to this project, its seems to be a necessity in order to complete the project within the time parameter.

SMOTE

A combination of our method of over-sampling the minority class and under-sampling the majority class can achieve better classifier performance (in ROC space) than only under-sampling the majority class. (Chawla, N., Bowyer, K., Hall, L. and Kegelmeyer, 2002). As we seek to create and train classification models over the course of this project, this technique will be utilized as needed in order to balance our data, which is necessary in order to create a suitably effective classifier.

Cross-Validation

Cross-validation is a widespread strategy for model selection and risk estimation because of its simplicity an apparent universality. (Arlot, S., Celisse, A. 2010) The method is an answer to the issue of training a model and testing the model with the same data produces a biased result. The model performs overoptimistically, to avoid this over the scope of the project a method of cross-validation will be utilized.

Ensembling

Ensemble learning strategies, especially boosting and bagging decision trees, have demonstrated impressive capacities to improve the prediction accuracy of

base learning algorithms. (G.I. Webb., Z. Zheng, 2004). The capability to further increase the accuracy of our classification model will enhance its appeal, this appeal will increase exponentially alongside the size of the dataset.

2 System

2.1 Requirements

2.1.1 Non-Functional requirements

Performance/Response time requirement

High performance and a low response time can be seen as a luxury in regard to this particular project as the current scope of the project relies on some models that require rather lengthy work times enables, additionally all future visualisations for the client could possibly be pre-rendered in order to cut down on lengthy wait times.

Availability requirement

As the dataset is stored locally, it will be available to the system over the course of the projects scope.

Recover requirement

As the basis of this project is formed on data and its manipulation and study, recoverability is an important factor. To avoid catastrophic loss of data all obtained data will be backed up in cloud storage such as GitHub, Microsoft OneDrive and Dropbox.

Robustness requirement

Robustness is not applicable to this project. (See Recover requirement above)

Security requirement

The interactions to obtain the data for this project from the Riot Games API will be conducted over HTTPS, which is suitably secure for the nature of this project. The files will be stored locally on a password protected computer with all common security considerations taken.

Reliability requirement

The Riot Games API is a well maintained and well documented API. Any new developments such new or depreciated methods will be accounted for over the course of the project.

Maintainability requirement

This system is considered a single project and will only be maintained up until the delivery date.

Portability requirement

Due to the nature of this project it would not be suited to any form of port.

Extendibility requirement

This project has good prospects for extension, and while no plans are in place potential extensions to the project can be found below. (See System Evolution).

Reusability requirement

The earlier stages of this project could be reused and repurposed for other studies and applications, but the later stages would be unique to the scope of this project.

Resource utilization requirement

Hardware such a PC/laptop will be needed for this project, in conjunction with this programming environments, visualisation tools and backup storage will be utilized.

2.1.2 Data requirements

For the purpose of this project input data retrieved from the respective APIs should be in the form of JSON as it enables swift use of the data with minimal data preparation.

2.2 Design and Architecture

The below diagram details the Architecture of the project from a high-level view. The components of the diagram are the dataset, which is stored in the local directory, the programming application which will perform the manipulation of the dataset and finally the visualisation application which will provide the final visualisations of the results of the project in order to suitably display them to the intended customer.

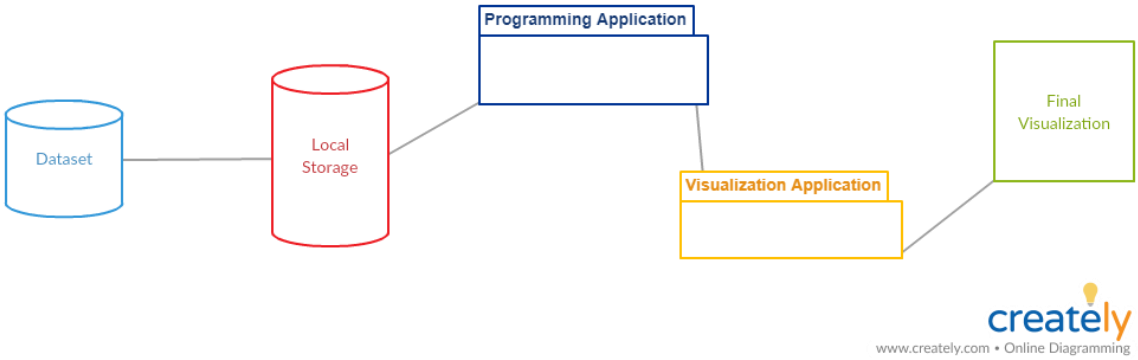

Figure 2 System Architecture

2.3 Implementation

For the purposes of this section the completed work will be expressed in code snippets as appropriate in conjunction with text. This project utilized both python and R code and will be labelled as such.

2.3.1 Twitter Analysis

In order to acquire and perform data preparation and modelling on Twitter data, the R language and RStudio was used. The method used in this project was adapted from the method observed within the documentation for the Text2Vec package.[9]

[9] http://text2vec.org/vectorization.html#tf-idf [Accessed 10/5/18]

Setup of R Environment

The first stage of the development is to inform the R environment of the libraries that will be needed over the duration of the project. This is performed by first installing the required package and then loading it into the script using the “library” command, as shown in figure 3. While the majority of the packages are critical to the success of the script, both the “doParallel” and “purrrlyr” packages have a singular use in the script. The “doParallel” package is rather convenient in that it allows for the reduction of the time it takes to fit our model by just over thirty minutes, almost halving the time. The package “purrrlyr” is utilized for its Dmap function a useful function that allows for the application of pre-processing functions to singular columns of the dataset. The remanding packages will be discussed upon utilization within the code.

Loading and Pre-processing of Training Data

The next step was to obtain and pre-process a training set of tweets, for the purpose of this project we utilized a resource that provides a mostly pre-processed set of 1,600,000 tweets.[10] The CSV containing the training set was loaded in, split into a training and test set and then prepared for modelling. The preparation performed upon the data was handled mostly utilizing the “Text2Vec” package as it provides many useful functions that simplify the typical tasks for any text mining venture such as vectorization and creating a Document Term Matrix(DTM). The package also provides functionality for term frequency-inverse document frequency (tf-idf) modelling, which allows for us to calculate the importance of various terms that appear within our document. This prevents more popular terms from overshadowing the less frequent terms.

[10] http://help.sentiment140.com/for-students/ [Accessed 10/5/18].

Model Training

The model chosen for the purpose of this project is cv.glmnet, a variant of glmnet that runs glmnet nfolds +1 times, firstly to get the lambda (λ) sequence and then continually to compute the fit with each of the folds omitted.[11] This type of model was chosen as glmnet is a remarkably efficient in fitting a variety of models and the cv.glmnet function allows for several parameters to be tuned in order reach an optimum performance for the model. The first of these tuning parameters "family", is the form of regression we wish to perform which in our case is Binomial logistic regression as we are seeking to perform classification on our tweets.

The second of these tuning parameters is the alpha value (α) which determines the form of shrinkage method to be used during the course of our run. There are three possible methods of shrinkage available to the cv.glmnet function, The Lasso (α = 1), Ridge regression (α = 0) and Elastic Net which encompasses α = 0-1 exclusive. Lasso utilizes a penalty that imposes a level of sparsity upon the coefficients which enables the results of this method to be more interpretable. Ridge utilizes a penalty that imposes a limit upon the size of the coefficient vector. The Elastic Net method is a compromise between the two methods and aims for a sparse solution with all highly correlated features to be averaged. (Hastie et al)

The third is the type.measure parameter which determines the form of loss to use for the cross validation process. For our purposes as we are utilizing two-class logistic regression we have access to the Area Under the Curve (AUC) type.measure, which gives the area under the Receiver Operating Characteristic curve (ROC curve). We will elaborate on this in the Evaluation section.

---

[11] https://www.rdocumentation.org/packages/glmnet/versions/2.0-16/topics/cv.glmnet [Accessed 10/5/18]

The fourth is the number of folds on which to perform cross validation, nfolds. This also determines the number of times glmnet is run in total.

Parallel is a tuning parameter available to cv.glmnet that enables the calculations to be completed in parallel with one another on the cores of the machine, if the machine is capable. This is enabled due to the foreach capabilities of the "doParallel" package, which vastly decreases the time that the model requires to run.

Maxit is the parameter that sets the maximum number of iterations for all λ values. Thresh is the convergence threshold for coordinate descent. Each inner coordinate-descent loop continues until the maximum change in the objective after any coefficient update is less than thresh times the null deviance.[12]

Twitter API Interaction

Once we have trained the model the next step is to obtain the tweets we wish to classify. To achieve this a Twitter account was needed to create a Twitter application. This allowed access to an Access Token, API Secret and a Access Token Secret. Once we have established a connection to the Twitter API with these credentials we can then search for relevant tweets to our project. Once the tweets have been obtained we once again perform the same pre-processing we performed upon the first set of tweets.

Result

Once the pre-processing task has been completed the model can then be run upon the obtained tweets in order to classify them, upon completion we can then plot the results of the model using the popular "ggplot2" package.

---

[12] https://www.rdocumentation.org/packages/glmnet/versions/2.0-16/topics/glmnet [Accessed 5/11/18]

2.3.2 Reddit Analysis

In order to acquire and perform analysis upon data from Reddit the language Python and the Spyder IDE were utilized and the method was adapted from this.[13] Two separate datasets where targeted for this analysis, headlines and comments. For headlines we targeted the new headlines of posts submitted to the League of Legends subreddit, the sample size for this project is rather small (> 1000) due to API constraints combined with time constraints. For comments an announcement thread of a new release within the game was targeted, in order to provide an apt example of the target of this research project. The analysis for each of the datasets follows a similar pipeline with a select few differences in each. The code was completed to the standards of the PEP-8 Python coding conventions.

Python Environment setup

The first stage in the pipeline is the importing of the needed packages for the project. The selection of packages loaded are those typical to any text mining project such as “numpy” and “pandas”. The “seaborn” package was utilized in order to enhance the visualizations produced over the course of the project. The science kit learn (sklearn) package is needed in order to perform and evaluate out selected models. The Natural Language Tool Kit (NLTK) package is utilized in order to perform our sentiment analysis of the text.

Reddit API Interaction

Before any analysis or pre-processing can be completed the data must first be retrieved via the Python Reddit API Wrapper (PRAW). In order to obtain the credentials in order to communicate with PRAW we need to first create a Reddit account and then create a Reddit application in order to access our Client Id and our Client Secret. Once these have been obtained they are combined with a unique

[13] https://www.learndatasci.com/tutorials/sentiment-analysis-reddit-headlines-pythons-nltk/#: [Accessed 10/5/18]

User Agent and an optional password in order to successfully perform communications.

For the headline acquisition we simply iterate over the headlines within the subreddit, with a limit set at 1000 headlines. For the comment acquisition requires some extra methods as the comments are stored within a comment forest object which contains the various levels of comments, it also contains multiple moreComments objects these moreComments objects will interfere with any attempt to iterate over the comments within the thread. To avoid this problem, we can utilize the replace_more method, which removes the moreComments objects within the data enabling safe iteration.

Data Pre-Processing

Once we have obtained the desired data we can then utilize the NLTK's Vader Sentiment Analyzer which enables us to categorize our data according to our sentiment, positive, neutral and negative. Along with these scores is the compound score which is the rating of each comment from Extremely Positive (a score of 1) to Extremely Negative (a score of -1). We can now set a label for our three levels of sentiment, positive above 0.1 compound score and negative for -0.1 compound score. The values in between will be considered our neutral cases.

In order to perform tokenization, the NLTK package is once again utilized and a word tokenizer is used that ignores punctuation in order to reduce the feature size in order to increase efficiency. The NLTK package also handles the issue of stop words, words irrelevant to the aim of our analysis.

Data Exploration

Once we have assembled our dataset we can then perform some light data exploration in order to observe information such as typical examples of positive and negative headlines and the totals of the individual categories.

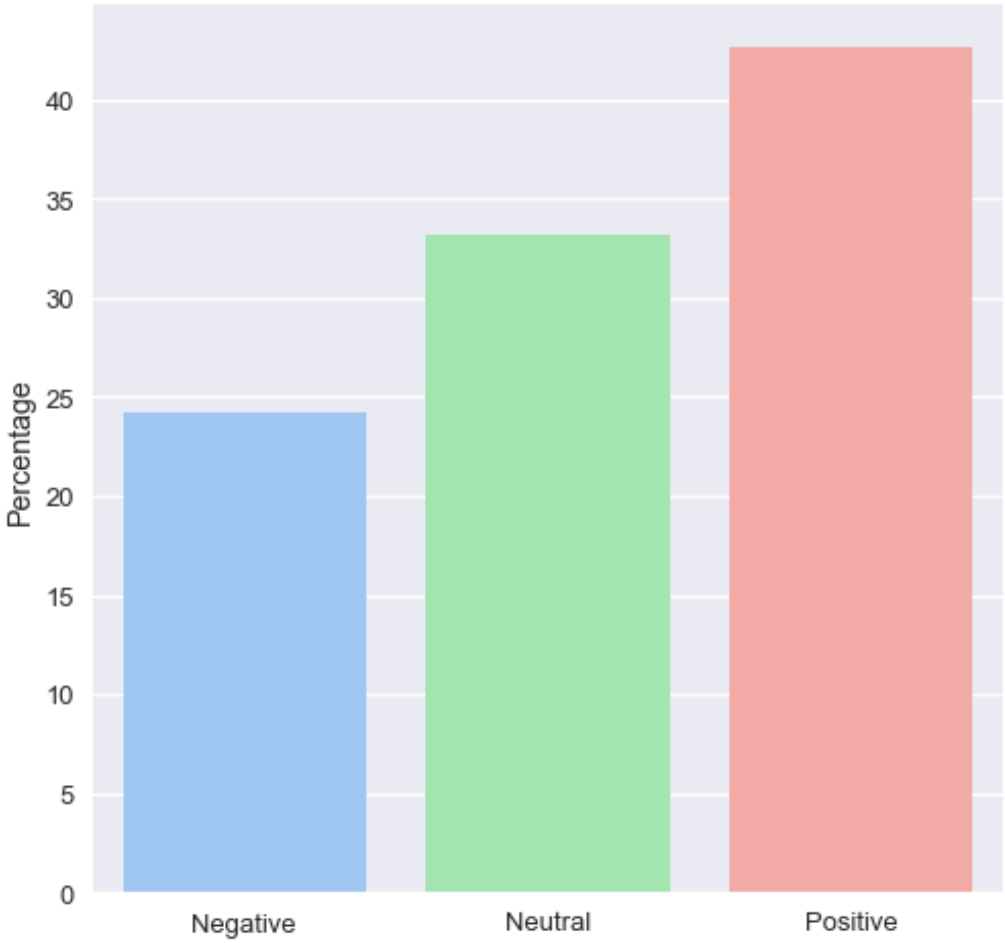

Figure 3 Sentiment Distribution on Reddit Comments

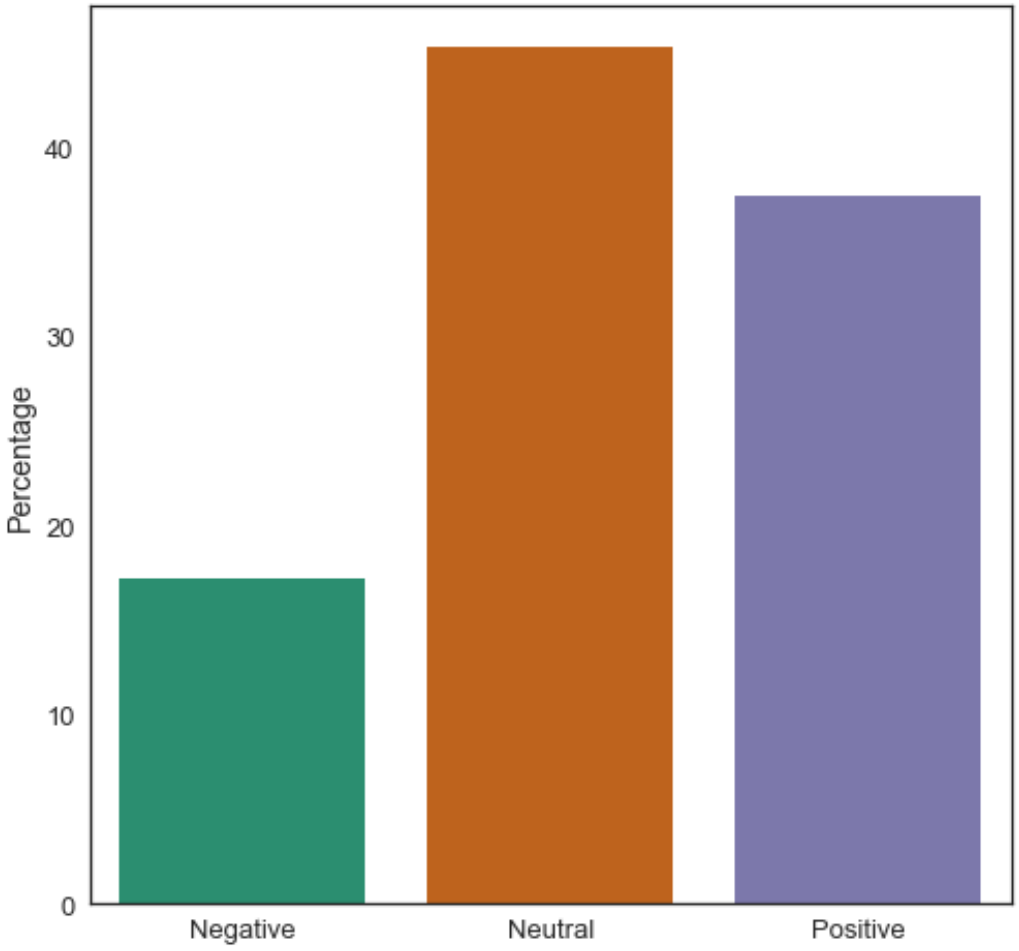

Figure 4 Sentiment Distribution on Reddit Headlines

Data Preparation for Modelling

The first step in preparing our data to be used in modelling is to split our data into our individual sets, training and test. This step is critical in order to avoid bias due to our model being affected by data leakage. Our test set will consist of one fifth of our data.

The second step in preparing our current data for modelling is the process of feature transformation, which entails the transformation of our data, currently text, into numeric form. To do this the “sklearn” package was utilized. CountVectorizer from “sklearn” enables both tokenization and vectorization simultaneously. This

method utilizes a sparse array in order to maintain a semblance of efficiency as rather than store every value of indices for every word it simply retains the non-zero values. To further retain efficiency, we may utilize the max_features argument of CountVectorizer if we wish to limit the number of features within the dataset. This argument will limit the number of features within the data to the numeric value supplied, which is equivalent to the same value of the most frequent words within the data.

Handling the Imbalance Within the Dataset

When observing the total positive and total negative features within our data we discover that the data is skewed towards positive, so much so that if we solely predict a positive result we would achieve an almost 65% rate of accuracy. As our model is binary this is not indicative of success and as such other methods will be utilized. As we are seeking to perform classification upon our imbalanced data it would be appropriate to first balance it, as this imbalance makes it more likely that the chance of misclassifying the minority class is at risk of being a much higher probability of misclassifying the majority class.

In order to combat this issue, the Synthetic Minority Over-sampling Technique (SMOTE) method can be utilized. The SMOTE method is an over-sampling technique that performs a combination of the method of over-sampling the minority class and under-sampling the majority class in order to balance the dataset, which increases the performance of the classifier due to the higher level of sensitivity the classifier to the minority class. (Chawla, N., Bowyer, K., Hall, L. and Kegelmeyer, 2002). To perform the SMOTE method upon our data the “imblearn” package is required.

Model Training

The data is now prepared for modelling. For the purposes of this project each of the classification models available to the “sklearn” package will be utilized. The models that were used over the course of the project are Multinomial Naïve Bayes, Bernoulli Naïve Bayes, Logistic Regression, Stochastic Gradient Descent

(SGD), Linear Support Vector Classification (LinearSVC), Random Forest and Multi-Layer Perceptron (MLP) Classifiers. The results of each model will be elaborated upon in the evaluation section.

In order to further enhance our modelling of the data the cross-validation method was utilized. Cross-validation is a method that enables a better understanding of the generalization qualities of a given classifier. The method of cross-validation utilized during this project is the Monte Carlo technique, which performs a random split upon the data into a set of training data and test data. This data is then fed to the model and then the model is run. This process is then repeated each time generating a random split independent to each of the other splits. This enable the results to provide less variance in your estimate at the cost of a risk of bias. This capability is provided by the “sklearn” package.

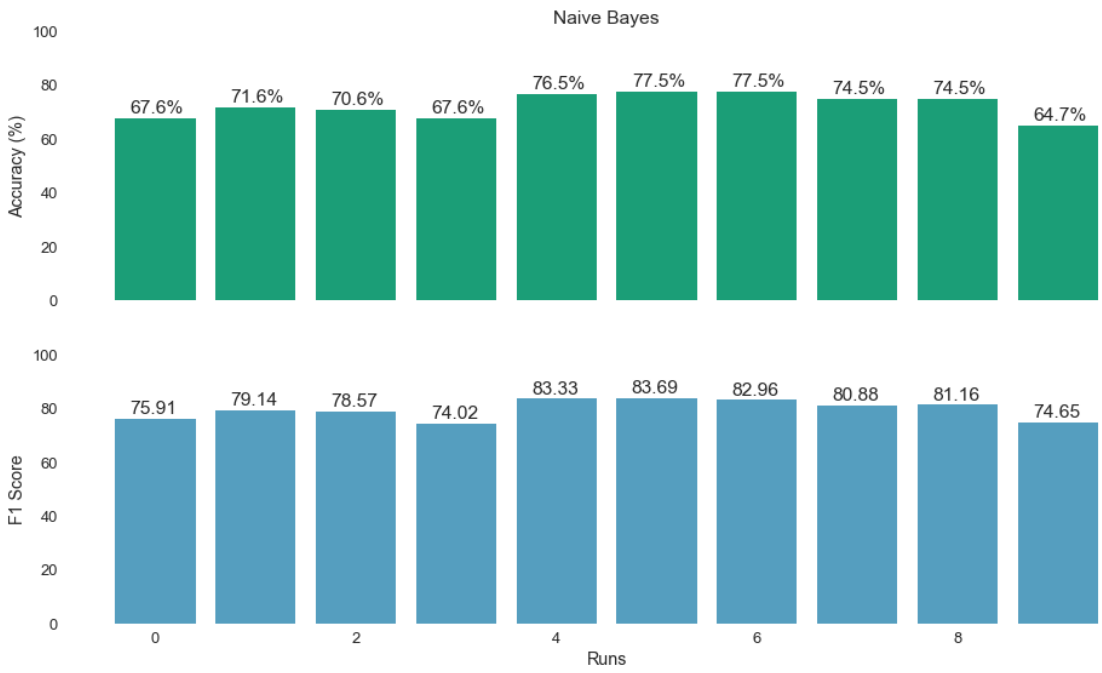

Figure 5 Power of Cross-validation Upon Headlines Data

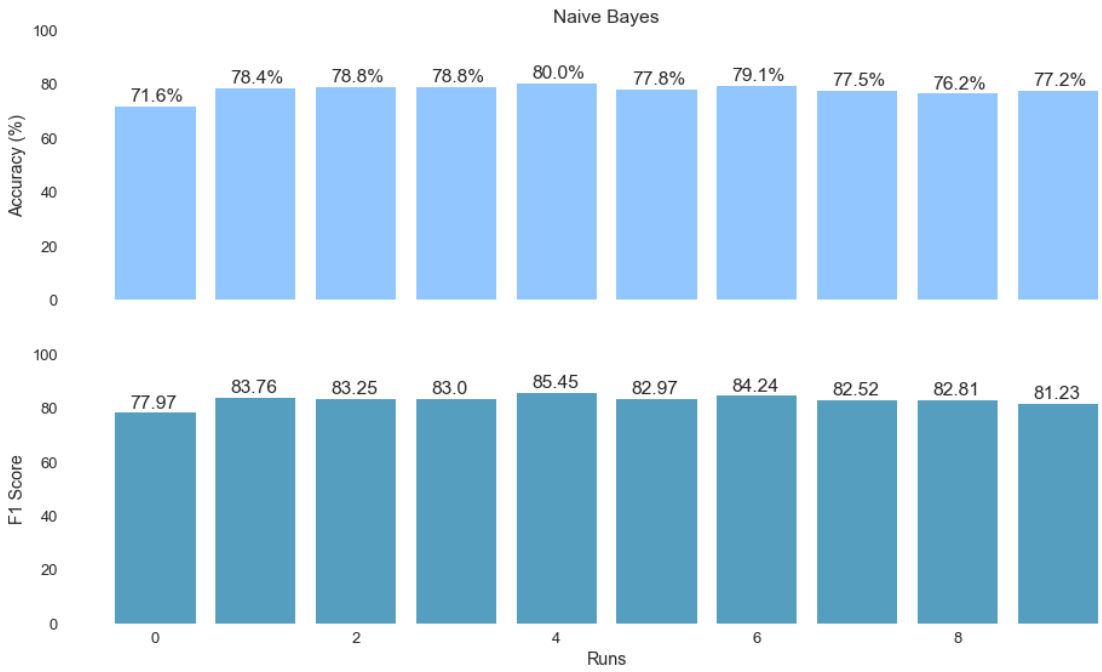

Figure 6 Power of Cross-validation Upon Comments Data

An alternate cross-validation technique that can be utilized is the k-fold technique. This technique utilizes a split of the data into k folds, upon each pass of the model each fold is used individually as test set and the remainder of the data is used as the train set. This continues until each of the fold has performed as the test set. This method sacrifices variance in order to avoid the risk of bias.

Ensembling

Ensembling is a technique in utilized in order to improve evaluation metrics. It typically consists of a finite selection of differing models, in our case classification models. It follows the principal that bigger is better, it utilizes each of the classifiers supplied to it simultaneously in order to improve its prediction efficiency. In our case we have used a simpler form of Ensembling known as Majority voting suited to our binary classification requirements.

2.4 Evaluation

In regard to the approach undertaken over the scope of this project, I believe this approach is rather unique and succeeds in providing novel insights that can be utilized in order to enhance a business strategy or perhaps even formulate a new one. Within this section each of the models utilized over the course of the project will be evaluated and the results will be aggregated in order to reach a conclusion.

Twitter Analysis

The glmnet model we created in order to perform the twitter analysis performed quite well with an AUC score of 0.8792. Below we can observe the AUC score for each value of λ.

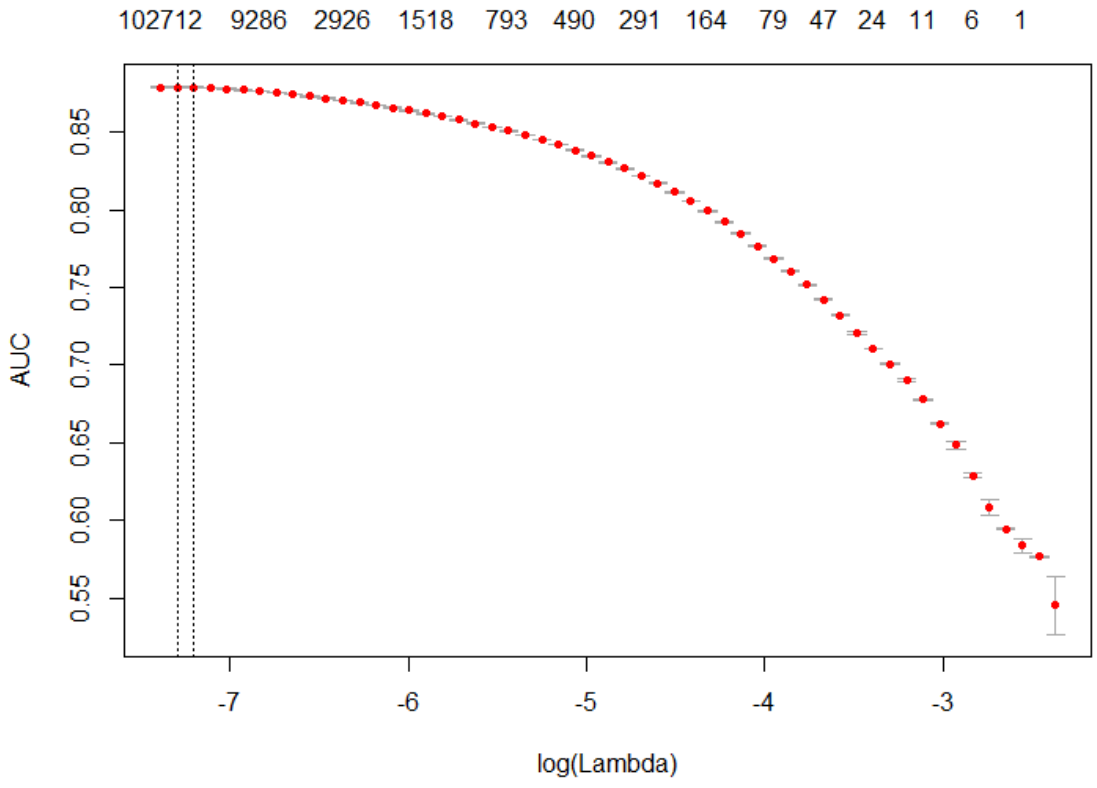

Figure 7 AUC for each value of λ

Upon observation of our final results we can observe clear spikes of activity within. On the 11/5/18 we can observe a large increase in the volume of tweets, with an abnormal amount of both extremely negative tweets and extremely positive tweets. This date coincides with the release of a trailer for a new content release within the game. This piece of content is aimed towards one of the least popular roles within League of Legends, which explains the large influx of negative tweets.

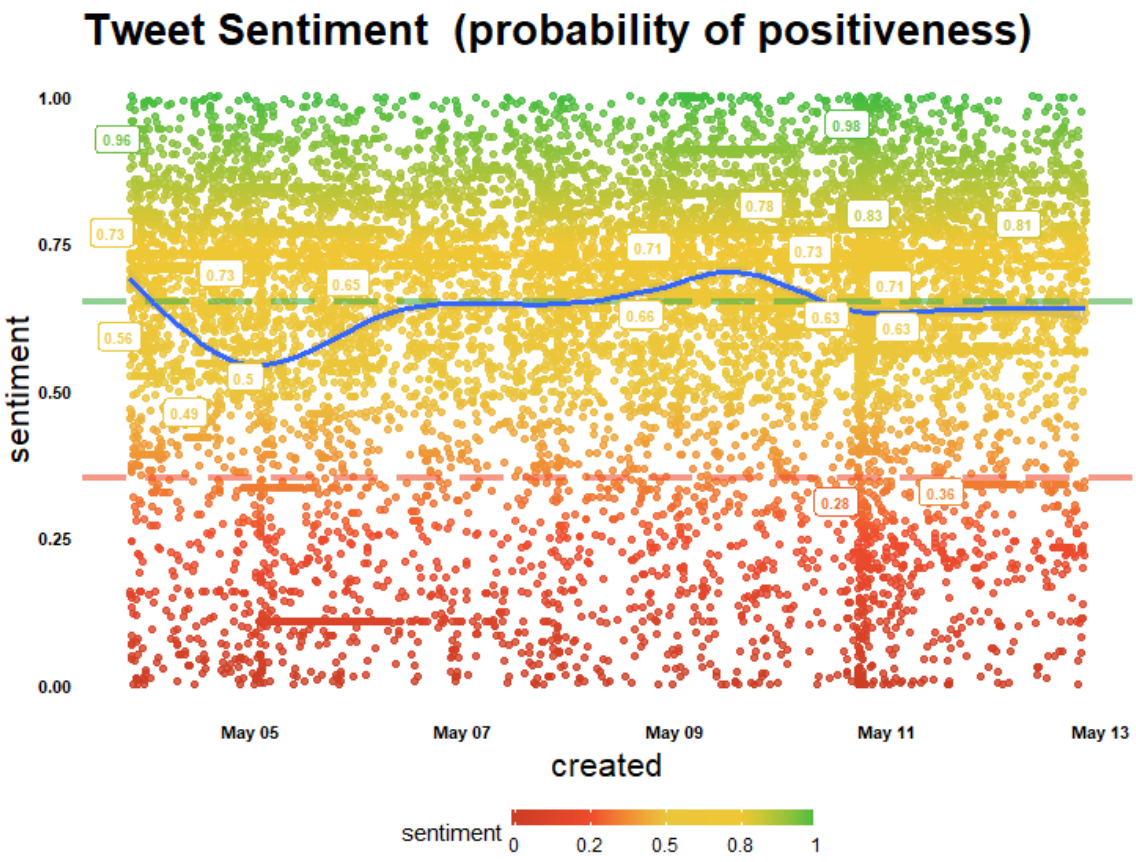

Figure 8 Twitter Analysis Result 13/5/18

We can observe a similar outcome in an earlier example of the same analysis, shown in figure 6. Once again there is an influx of tweets both negative and positive, that coincides with the release of a content update within the game on 5/5/18. Likely a result of the player base voicing grievances and support for the content.

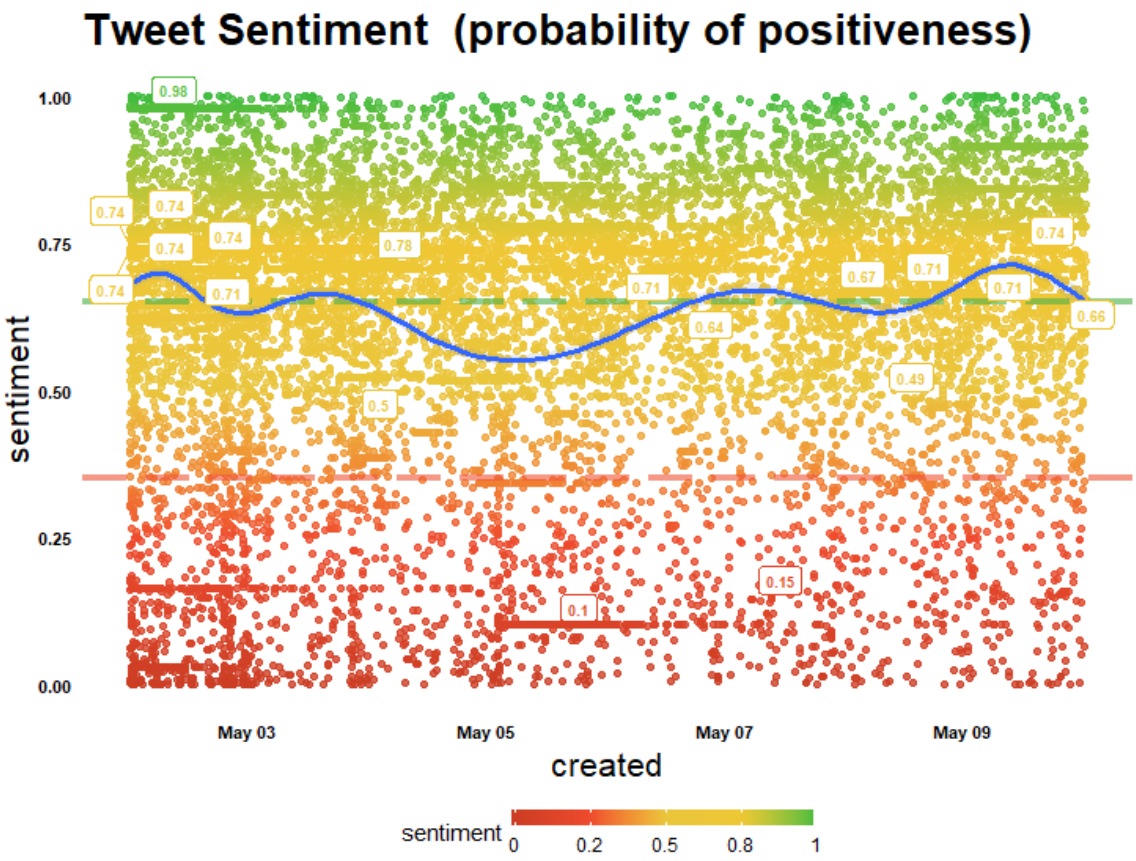

Figure 9 Twitter Analysis Result 10/5/18

Just from this result alone we can observe the validity of utilizing this method to observe consumer reaction to product releases. This is at least the case within the Gaming market where the average age is quite low which indicates a higher likelihood of a social media presence.

Reddit Headline Analysis

Over the course of the Reddit headline analysis several models were utilized. As our classification task is binary we cannot simply observe our accuracy score, its power as an evaluation metric is reduced. This is true as a random classifier will predict correctly 50% of the time, as such an accuracy of 65% tells us little. To avoid this, we will utilize two alternate evaluation methods: f1 score and confusion matrices. F1 score utilizes both the precision and the recall of the

model in order to provide a more effective accuracy score. The confusion matrix is a method for visualizing the performance of a binary classifier. It is composed of the predictions made by the classifier, graded according to accuracy.

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Figure 10 Confusion Matrix[14]

The confusion matrix grades are as follows:

- True Positive: Cases in which the classifier predicted 1 and was correct.
- True Negative: The classifier predicted 0 and was correct.
- False Positive: The classifier predicted 1 and was incorrect.
- False Negative: The classifier predicted 0 and was incorrect.

Shown below are the results obtained over the course of the project.

Multinomial Naïve Bayes

Average Accuracy: 74.07%

Average F1 Score: 81.03

Average Confusion Matrix:

[14] http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/ [Accessed 13/5/18]

[[19.6 14.6]

[13.15 59.65]]

Bernoulli Naïve Bayes

Avg. Accuracy: 74.35%

Avg. F1 Score: 81.94

Avg. Confusion Matrix:

[[17. 17.2]

[10.25 62.55]]

Logistic Regression

Avg. Accuracy: 73.04%

Avg. F1 Score: 80.13

Avg. Confusion Matrix:

[[19.6 14.6]

[14.25 58.55]]

Scholastic Gradient Descent Classifier

Avg. Accuracy: 72.71%

Avg. F1 Score: 80.20

Avg. Confusion Matrix:

[[18.4 15.8]

[13.4 59.4]]

Linear Support Vector Classifier

Avg. Accuracy: 72.57%

Avg. F1 Score: 79.51

Avg. Confusion Matrix:

[[20.25 13.95]

[15.4 57.4]]

Random Forest Classifier

Avg. Accuracy: 68.97%

Avg. F1 Score: 77.71

Avg. Confusion Matrix:

[[15.25 18.95]

[14.25 58.55]]

Multi-Layer Perceptron Classifier

Avg. Accuracy: 73.36%

Avg. F1 Score: 80.41

Avg. Confusion Matrix:

[[19.65 14.55]

[13.95 58.85]]

Surprisingly the considerably more sophisticated and advanced methods such as the Support Vector and Perceptron methods do outperform the less sophisticated Naïve Bayesian methods, in fact Naïve Bayes performed the best of all the classifiers. This is likely due to our small sample size and on larger samples the more sophisticated methods would begin to outshine the others. The f1 score of our models are rather satisfactory but could perhaps be improved with a variety of techniques.

Another observation garnered over the course of the project is the adherence of our data to Zipf's law, a statistical law that states the frequency of any words is inversely proportional to its position in the frequency ranking. As such the most popular word appears twice for every instance of the second most popular word and so on. We can observe this law in effect in the following figures.

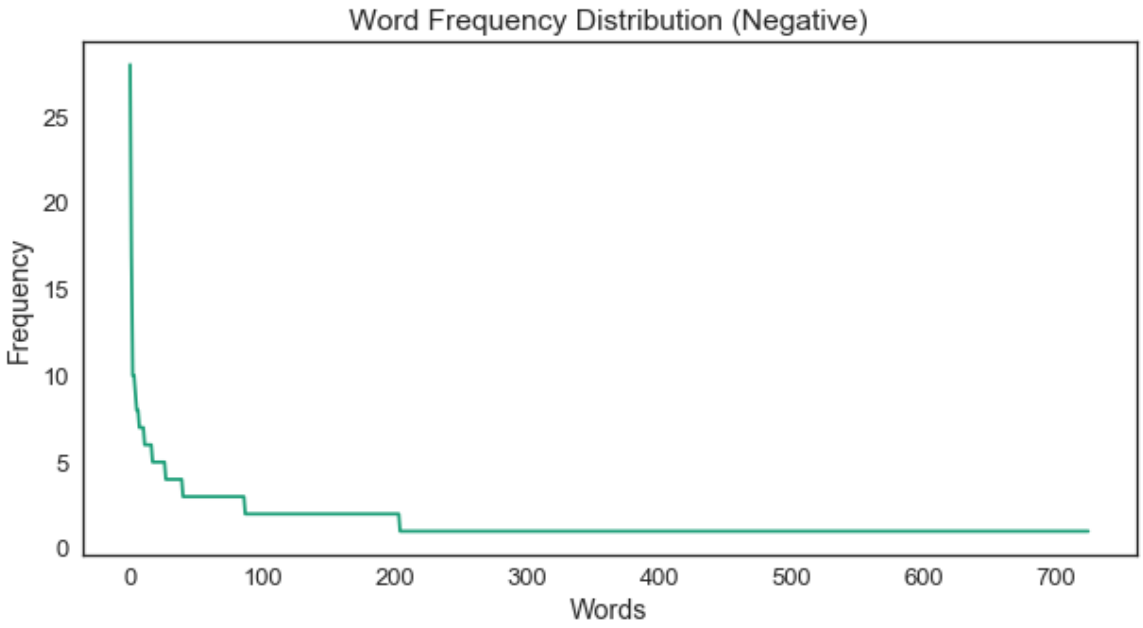

Figure 11 Word Frequency Distribution for Reddit Headlines

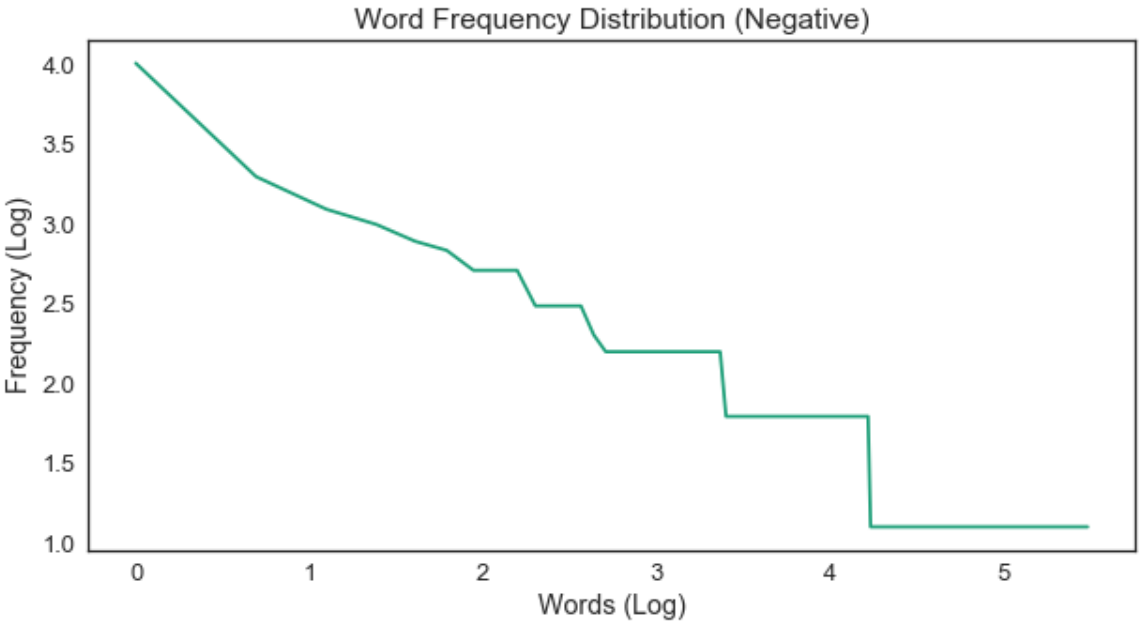

Figure 12 Zipf's Law in Action

Reddit Comment Analysis

The pipeline for analysis of the Reddit comments data was very similar to the pipeline for the Reddit headlines analysis.

Shown below are the results obtained over the course of the analysis.

Multinomial Naïve Bayes

Avg. Accuracy: 77.38%

Avg. F1 Score: 82.88

Avg. Confusion Matrix:

[[72.1 43.5]

[28.9 175.5]]

Bernoulli Naïve Bayes

Avg. Accuracy: 76.84%

Avg. F1 Score: 82.33

Avg. Confusion Matrix:

[[72.9 42.7]

[31.4 173.]]

Logistic Regression

Avg. Accuracy: 76.58%

Avg. F1 Score: 81.35

Avg. Confusion Matrix:

[[81.45 34.15]

[40.8 163.6]]

Scholastic Gradient Descent Classifier

Avg. Accuracy: 74.11%

Glenn Connell x14441832

Avg. F1 Score: 79.67

Avg. Confusion Matrix:

[[74.2 41.4]

[41.45 162.95]]

Linear Support Vector Classifier

Avg. Accuracy: 75.64%

Avg. F1 Score: 80.49

Avg. Confusion Matrix:

[[81.1 34.5]

[43.45 160.95]]

Random Forest Classifier

Avg. Accuracy: 71.31%

Avg. F1 Score: 78.58

Avg. Confusion Matrix:

[[59.75 55.85]

[35.95 168.45]]

Multi-Layer Perceptron Classifier

Avg. Accuracy: 76.50%

Avg. F1 Score: 81.51

Avg. Confusion Matrix:

[[78.85 36.75]

[38.45 165.95]]

As we can observe from the results above our accuracy and f1 scores are, in general higher than our results using the same models upon the Reddit headlines. This is likely due to our larger dataset increasing the efficiency of the models being run. Once again, we can also observe a lack of a clear victor when it comes to performance. As with our Headline dataset we can also observe Zipf's law in action upon our comments dataset.

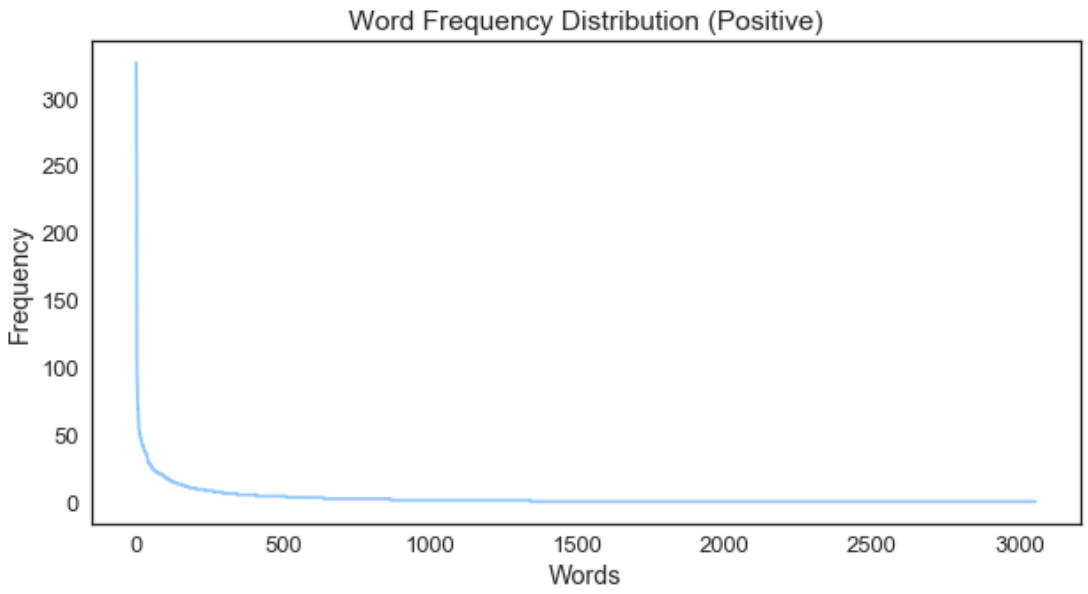

Figure 13 Word Frequency Distribution for Reddit Comments

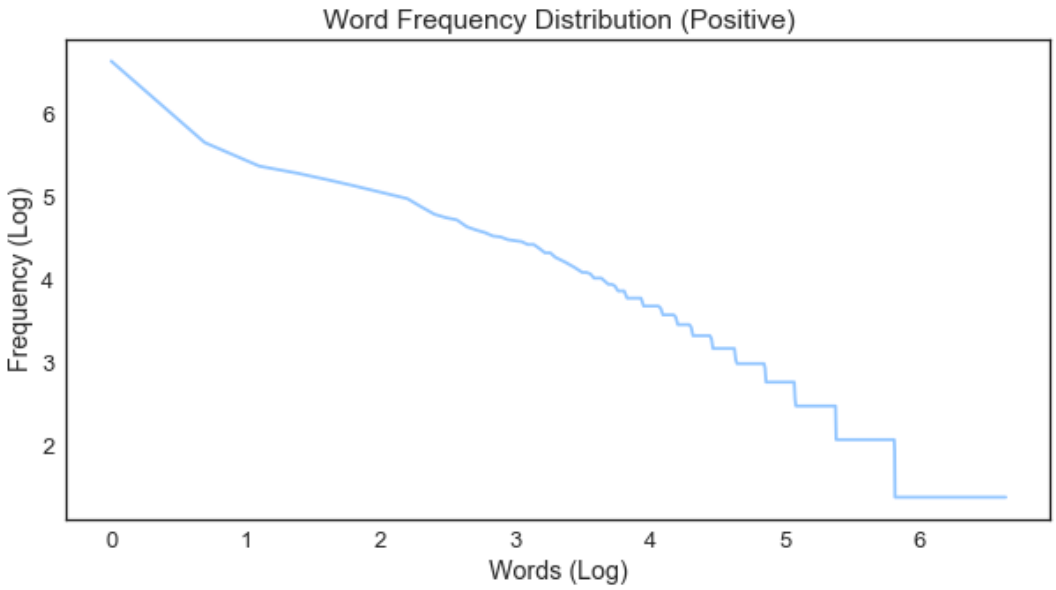

Figure 14 Zipf's Law in Action

2.5 Graphical User Interface (GUI) Layout

After careful deliberation, the conclusion reached is that a GUI will not add significantly to the project.

2.6 Testing

Over the course of the project the following testing methodologies were utilized, Black box and White box testing. The Black box testing focuses upon an examination of the functionality of the project with little emphasis upon the knowledge required to produce the functionality. White box testing instead focuses on the inner working of the project rather than the functionality aimed towards the user.

Supplied is a test script based on the suggested template supplied by Cambridge University press. The template follows the following layout:

AUT Name: the name of the Application Under Test.

AUT Version: the version information for the Application Under Test.

Iteration ID: the unique identifier for the iteration this test is being conducted in.

Date of Test: the start date of testing.

Test ID: the unique identifier for the test.

Purpose of Test: a brief description of the purpose of the test including a reference where appropriate to the requirement that is to be tested, as well as any dependencies from or to other Test Scripts/Test Cases.

Test Environment: a brief description of the environment under which the test is to be conducted.

Test Steps: concise, accurate and unambiguous instructions describing the precise steps the Tester must take to execute the test, including navigation through the AUT as well as any inputs and outputs.

Expected Result: a brief and unambiguous description of the expected result of executing the test.

Actual Result: a brief description of the actual result of executing the test.

Suggested Action: a suggested solution to the given problem.

Resolution: the final solution achieved for the given problem.

Black Box Test 1			
AUT Name	Twitter Pipeline	Version	6
It. ID	6.0	Date of Test	11/5/18

Test ID	BlackBox1
Purpose of Test	The test dataset is suitably modified for use.
Test Environment	A PC Specialist laptop with windows 10 and the Version 1.0.153 of RStudio.
Test Steps	From R studio the user should ensure the required file is within current working directory, successfully load the file into RStudio and proceed to apply the required pre-processing functions.
Expected Result	A set of data without symbols, correctly labelled columns and a sentiment score between 0-1.
Actual Result	The result is as expected.
Suggested Action	N/A
Resolution	N/A

Black Box Test 2			
AUT Name	Reddit H. Pipeline	Version	6
It. ID	6.0	Date of Test	11/5/18

Test ID	BlackBox2
Purpose of Test	The API call is correctly executed and iterated upon.
Test Environment	A PC Specialist laptop with windows 10 and Version 1.8 of Anaconda Navigator.
Test Steps	From Spyder IDE the user should be able to run the start of the script in order to begin communication with the PRAW.
Expected Result	A set of data that consists of up to 1000 headlines from the relevant subreddit.
Actual Result	The result is as expected.
Suggested Action	N/A
Resolution	N/A

Black Box Test 3			
AUT Name	Reddit C. Pipeline	Version	6
It. ID	6.0	Date of Test	12/5/18

Test ID	BlackBox3
Purpose of Test	The API Request is correctly iterated upon.
Test Environment	A PC Specialist laptop with windows 10 and Version 1.8 of Anaconda Navigator.
Test Steps	From Spyder IDE the user should be able to run the start of the script in order to begin communication with the PRAW.
Expected Result	An object containing a collection of 2000+ comments with no moreComments within.
Actual Result	An object with 450 comments and a moreComments within.
Suggested Action	Consult the documentation for PRAW to determine the nature of the moreComments object in order to determine a solution.
Resolution	The replace_more method was introduced to the code in order to replace the moreComments object with comments.

White Box Test 1			
AUT Name	Twitter Pipeline	Version	6
It. ID	6.0	Date of Test	11/5/18

Test ID	WhiteBox1
Purpose of Test	Data tokenization
Test Environment	A PC Specialist laptop with windows 10 and the Version 1.0.153 of RStudio.
Test Steps	From R studio the user should be sure of the presence of the data and should be able to run the tokenization script.
Expected Result	A correctly tokenized object derived from the original data.
Actual Result	The result is as expected.
Suggested Action	N/A
Resolution	N/A

White Box Test 2			
AUT Name	Reddit H. Pipeline	Version	6
It. ID	6.0	Date of Test	12/5/18

Test ID	WhiteBox2
Purpose of Test	Ensure vectorization is working as intended.
Test Environment	A PC Specialist laptop with windows 10 and Version 1.8 of Anaconda Navigator.
Test Steps	From Spyder IDE the user should be able to run the vectorisation function in the script and observe its effects.
Expected Result	A successfully vectorized output, available for examination.
Actual Result	The result is as expected.
Suggested Action	N/A
Resolution	N/A

White Box Test 3			
AUT Name	Reddit C. Pipeline	Version	6
It. ID	6.0	Date of Test	11/5/18

Test ID	WhiteBox3
Purpose of Test	Ensure ensembling is working correctly.
Test Environment	A PC Specialist laptop with windows 10 and Version 1.8 of Anaconda Navigator.
Test Steps	From Spyder IDE the user should be able to run the ensembling portion of the script and observe its effects.
Expected Result	A successful run of our ensemble classifier
Actual Result	An issue within the ensemble classifier.
Suggested Action	Review the code and inspect documentation to determine a solution.
Resolution	Now completes, but still contains an issue with evaluation metric calculation.

3 Conclusions

Over the course of this project and during the research undertaken to complete it, it has become evident that there is a niche within the Gaming market that can exploit ventures such as this. This is due in part to the demographic that the market tends to, it trends toward a younger consumer base causing an increased likelihood of an active social media presence. This combined with a vocal player base can lead to a vast resource of data to be accessed. This project is but a proof of concept that this source of data can be utilized. Take the Twitter analysis results as an example of this, we can observe that while for both content releases there was a large influx of tweets both positive and negative, each time content was released the sentiment of the tweets tended to decrease in score from the day before. Indicating perhaps a more vocal negative party on Twitter, as the majority of the comments on Reddit were perceived as Positive.

4 Further development or research

One of the constraints placed upon this project due to its nature is time, it removes the possibility of obtaining a dataset that could be considered “Big”. The limited time prevents a suitable window for a large data collection from the PRAW, as API calls are limited. If this project was to be revisited at a later date, two solutions to avoid this issue could be utilized: implement a streaming communication with the API or to use the pushshift API.[15]

A streaming API communication would provide the benefit of a large amount of data continuously in a relatively short period of time. The issue with this approach is that it sacrifices accuracy. If we wish to simply observe user reaction to content releases within the game then a method must be devised in order to filter the content, as streaming simply acquires all content posted to Reddit or a subreddit indiscriminately.

Pushshift is an alternate to simply scraping the data from Reddit itself, it enables users to connect to Reddit via an Endpoint and then perform requests from Reddit. This enables the user to remain “polite” in avoiding abusing the site via Web Crawler, while pushshift handles all communication with Reddit gracefully, respecting rate-limits and handling errors that occur in communication.

An application of either of these methods could vastly improve the quality of the results of this project if re visited in the future.

Another potential area of improvement is to potentially employ the use of:

- Bootstrap Aggregating, to decrease the variance of our predictions.
- Another method of boosting, rather than the majority rule method attempted over the course of the project.
- Stacking, to both increase prediction prowess and minimize variance.

[15] https://pushshift.io/ [Accessed 13/5/19]

5 References

Citeseerx.ist.psu.edu. (2017). [online] Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.198.5133&rep=rep1&type=pdf [Accessed 29 Nov. 2017].

Hastie, T. (2008). The Elements of Statistical Learning. 2nd ed. [ebook] pp.661-668. Available at: https://web.stanford.edu/~hastie/Papers/ESLII.pdf [Accessed 11 May 2018].

Chawla, N., Bowyer, K., Hall, L. and Kegelmeyer, P. (2002). Journal of Artificial Intelligence Research 16 (2002). [ebook] pp.321–357. Available at: https://jair.org/index.php/jair/article/view/10302 [Accessed 12 May 2018].

Bing Liu. (2018). Sentiment Analysis and Opinion Mining | Synthesis Lectures on Human Language Technologies. [online] Available at: https://www.morganclaypool.com/doi/abs/10.2200/S00416ED1V01Y201204HLT016 [Accessed 13 May 2018].

Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys, 34(1), pp.1-47. [online] Available at: https://dl.acm.org/citation.cfm?id=505283 [Accessed 13 May 2018]

Arlot, S., Celisse, A. (2010). [online] Available at: https://projecteuclid.org/download/pdfview_1/euclid.ssu/1268143839 [Accessed 13 May 2018].

G.I. Webb., Z. Zheng (2004). Multistrategy ensemble learning: reducing error by combining ensemble learning techniques - IEEE Journals & Magazine. [online] Available at: https://ieeexplore.ieee.org/abstract/document/1318582/ [Accessed 13 May 2018].

6 Appendix

6.1 Project Proposal

6.1.1 Objectives

The project I am proposing is a statistical study of the effectiveness of in game events upon the playtime of players. For this project, I have chosen to use Riot Games league of legends as my subject, due to its suitability for my study. It has a massive player base and as such a diverse dataset along with having a well-documented API.

I will be using the language R along with R-Studio for gathering along with cleaning my data. I have chosen this technology as I have been using it extensively for one of my modules which has allowed me to gain a good understanding of the use of the technology that will only continue to improve I gain experience with the technology. Another reason for this choice is that R is a popular package for statistics and is sure to have a large amount research material available which will benefit my project immensely as rather advanced statistics will be needed.

I will be using the Riot Games API as the main source of my dataset and will be drawing my tailored data from there. I will then clean and perform my calculations with R and within R-Studio.

I will then transport my dataset to an application, possibly tableau, to properly visualize my data. I also hope to incorporate a way to dynamically manipulate the data from this application.

6.1.2 Background

This *study will be undertaken with the goal of finding the "average player" and discovering the effects, if any, of the in game events within the game. This will then help to discover whether the events increase the average players playtime or*

affects their behaviour in any other determinable way. With the data gathered from my analysis we will be able to determine the worth of events to Riot Games and eventually other companies.

Once I have completed this project the same analysis can be applied to multiple games across several platforms, which is where the true value of this project is as the commercial value increases as we broaden the audience for the results.

My Inspiration:

I have always had an avid interest in computer gaming, and particularly I have always enjoyed viewing statistics and graphs representing data about the games I enjoy. This helped kickstart my interest in the field of data analytics as I would always ask myself the question of "How do they know that?" regarding the interesting statistics that Gaming companies would display, usually in a nice neat graph. It is with this in mind I delved into the process companies use to gather and calculate these metrics from their data and have in turn been inspired to undertake this project.

The main differentiating factor between my inspiration and my project is that I will be looking to garner some insight into the value such metrics can provide rather than eye-catching large statistics such as maximum player count and other such statistics.

6.1.3 Technical Approach

A brief description of the approach:

First step in the approach is research and literature review. I will need to read the documentation for the Riot Games API to discern the most effective ways to retrieve my data along with learning the general structure of my data, which will aid me in manipulating and cleaning my dataset.

Once I have completed my initial research the next step will be to determine the requirements needed to carry out the project to its fullest potential.

On completion of requirements capture I can then proceed with the capturing of my dataset. I will achieve this by collecting data via the Riot Games API over the course of a suitable event. Once suitable data has been gathered the first thing to do is to clean my data.

Once I have suitably cleaned my data the next step will be to perform analysis on the data to gather what an average players details are. Once this data has been obtained we can then carry out our study to determine if there are any effects upon the players behavior over the course of the event. To aid in the statistical side of this analysis I will be making use of the text An Introduction to Statistical Learning for reference.

Once the outcome of the study has been obtained I plan to visualize the data within a software, potentially Tableau, and build an application around it. Within the application there will be visualizations along with a dynamic manipulation of the data.

6.1.4 Special resources required

An Introduction to Statistical Learning. Available at http://www-bcf.usc.edu/~gareth/ISL/

6.1.5 Project Plan

See (6.2 Project Plan)

6.1.6 Technical Details

R

R is a software environment designed for computation and graphics based on statistics. It allows for a platform with a wide variety of packages and libraries.

R Studio

R Studio is an IDE for running and compiling R code.

Tableau

Tableau is software that provides the ability to enhance the impact of ones' data through powerful visualizations.

Riot Games API

The Riot Games API is a powerful and well documented API for all data pertaining to the game League of Legends.

6.1.7 Evaluation

Regarding the testing during the initial stages of the project, I will perform all testing until the time comes for the application to be created. Once the application comes into play I will be performing regular testing along with turning the application over to a group of my peers in order for them to rigorously use the application. This rigorous testing will ensure that the project is bug free once completed.

6.2 Project Plan

	Start Date	Duration(Days)	End Date
Review Feedback from the MidPoint	12/6/2017	2	12/8/2017
Make adjustments as necessary	12/8/2017	12	12/20/2017
Study Period	12/20/2017	20	1/10/2018
Project Development	1/10/2018	100	4/20/2018
Evaluation	4/20/2018	10	4/30/2018
Finalise Documentation	4/30/2018	5	5/4/2018
Submission	5/4/2018	1	5/5/2018

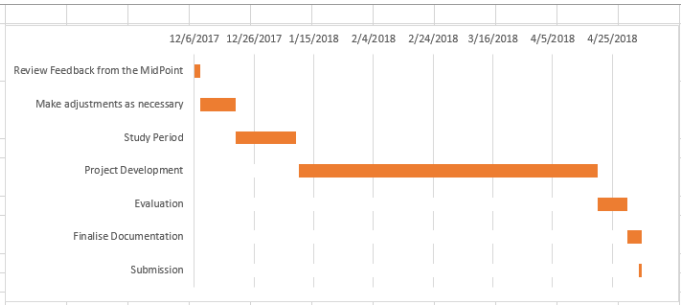

Figure 15 Post Mid-Point Project Plan

6.3 Monthly Journals

Month of September

Glenn Connell x14441832

September was a month of research, while I had devised the general gist of my idea earlier in the year, it was during this month that I really began to flesh out my idea into a fully formed project. This mostly involved online research into the potential technologies that I can use for my project. The result of this research was that I would most likely be using python or R in order to obtain my desired dataset. Another result of this research was that I will be using the Riot API in order to get my dataset. This all depends whether the project gets past the pitch. I have also begun to use Trello in order to manage tasks needed for my project which has proven to be a tremendous help in keeping my work on track already.

I have completed the pitch and have been approved to continue with this idea with one caveat which is that I must expand my dataset to encompass more than one game or that I change my dataset to encompass multiple game platforms and genres in order to generalize my dataset. I am leaning towards the second option as it will allow me to appeal to a wider audience rather than just Riot along with making my dataset easier to analyze as my analysis will be less dependent on the quality of my data which was a problem with the previous idea. All together I think the pitch went well and it has allowed me to sharpen my idea up into the project I will be working on for the rest of my year at NCI. Next month I will finalize my research and decide which of the two routes I wish to take. Once I have done that I will continue my work on my documentation and perhaps begins some work on my prototype.

Month of October

This month was a rather slow month in terms of progress on my Overall project, primarily due to the large number of deadlines over the course of this month. While the overall progress achieved an important milestone, which was the completion of my project proposal. Along with that I performed some research crucial to the Requirements Specification Document due next month. This was mostly wrapping

my head around the basics of the KDD and Crisp DM methodologies, I will be choosing one of these methodologies for my project and this research will be integral for the decision. Based on my research I am leaning towards using Crisp DM for the needs of my project but further deliberation is needed.

Over the course of this month I also performed one of my first attempts at pulling some data from the Riot API, which was partially successful. It was unsuccessful in the fact that while I was able to pull a fairly large amount of data the data I obtained was not necessarily the data I require, a lot of the data was not necessary and caused more hassle in the cleaning of my data, which has put me off gathering anymore data until I can perform more research. Next month shall be the month in which I will be putting a lot of my research into action, with my requirements specification and the building of my prototype.

Month of November

This month was focused on completion of the goals I had set for myself in order to perform well in the upcoming mid-point presentations. This is one of the big milestones of this semester and it is important that I can field a semi competent product in order to pass.

It has been a hectic month but I believe I am prepared for the midpoint presentation. I am happy I decided to work on this early as I have just realized the extent of the amount of work that is due in the next two months. It's going to be chaotic!

Month of December

I have completed the midpoint presentation and I am currently awaiting my result. I believe the presentation went quite well, I successfully presented the work I had completed and the examiners seemed pleased and did not ask too many questions. The month so far has been absolute bedlam with projects due left and

right. With exams coming up in January I am going to postpone any work on the Software project as I currently do not have the time to devote to it.

I passed the Midpoint! I was only 3% off a 1st too! I am quite happy to have passed and that's one milestone down for the year. I will be contacting Simon soon in order to get some feedback on the presentation hopefully.

Month of January

This month has been full focus on study so far, the exams are tough and there is quite a few of them so the pressure is on.

The exams are finished which is a great relief. I will be meeting Simon soon in order to get feedback on my Midpoint.

Month of February

I have restructured the idea for my Software project. The project is retaining a similar theme to my original idea in that it is focused on the impact of content releases within F2P games, League of Legends in particular. Instead of utilizing the game data gathered from the Riot Games API, I will be performing analysis on Social media data related to the League of Legends Social media presence. The rest of this month will be performing research and then beginning on the basics of the project.

Month of April

This month has been extremely hectic so far. Exams are coming up towards the end of this month and on top of that the deadline of the Software project is encroaching on me. So, the pressure is on. I will be splitting my time this month between intense study and intense work.

Month of May

This will be my last Journal and I could not be more relieved; the end is in sight! I am currently hashing out the final details of my project and beginning on my final

report. The journey has been a long one and a rewarding one, I have picked up a slew of knowledge and techniques that will no doubt be useful in the future.

6.4 Other Material Used

An Introduction to Statistical Learning. Available at http://www-bcf.usc.edu/~gareth/ISL/ .

Trap.ncirl.ie. (2017). Welcome to TRAP@NCI - TRAP@NCI. [online] Available at: https://trap.ncirl.ie/ [Accessed 30 Nov. 2017].