Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Computer Science and Statistics

Boredom Classifier Predicting Boredom Of Users Based On Mobile Phone Usage

Jasmine Ajaykumar Kanav
Supervisor - Dr. Owen Conlan

A Dissertation submitted in fulfilment
of the requirements for the degree of
MSc (Computer Science - Data Science)

Declaration

I hereby declare that this Dissertation is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at `http://www.tcd.ie/calendar`.

I have also completed the Online Tutorial on avoiding plagiarism ‘Ready Steady Write’, located at `http://tcd-ie.libguides.com/plagiarism/ready-steady-write`.

Signed: ______________________ Date: ______________________

Abstract

The project is based around empathy based data analysis, and aims to investigate the correlation between users' emotional states and their mobile phone usage. This study's core objective is to identify the moments of boredom in a mobile phone user and build a model that can subsequently help find opportune moments for the delivery of notifications. The primary motive is to utilize moments of boredom to identify whether the user would interact with the notification based on his mobile phone usage.

A Machine Learning approach was taken to be able to utilize maximum features, and the machine learning algorithms were used to train a model that was then applied to a different dataset to predict boredom based on mobile phone usage patterns. For evaluation of the machine learning models, Confusion Matrix, AUC-ROC curve, recall, precision, accuracy and F1-Score were used.

It was possible to build a boredom classifier based on a user's mobile phone usage patterns from one dataset and use it to classify moments of boredom on users in a different dataset. Amongst the three Machine Learning Models used, Random Forest Classifier performed the best with an F1 score of 0.93 and Accuracy of 0.88. From the 5 features considered for building the ML models, Application Type gave the best predictions of boredom on the WeAreUs dataset with 11960 instances of boredom detected.

Acknowledgements

I would like to acknowledge and give my thanks to my supervisor, Dr. Owen Conlan, who made this work possible. His guidance and continuous support helped me through writing all stages of this research project.

I would also like to thank Kieran Fraser for providing the synthetic notification dataset which was crucial for evaluation of the research, and also for his advice and technical help during the project.

Contents

List of Figures

1 Introduction

1.1 Motivation

Notifications flood our mobiles continuously, as applications, emails, reminders, alarms, messages all send push-notifications to grab the attention of the mobile user, and since interacting with mobiles has become an integral part of people's lives, it is plausible that users check most of the notifications received on their cellular devices. A report by Deloitte (1), states that people in Ireland check their phones about 50 times within a single day, and majority of this is to check and engage with the notification alerts. Figure 1.1 depicts the distribution of mobile phones usage patterns in Ireland.

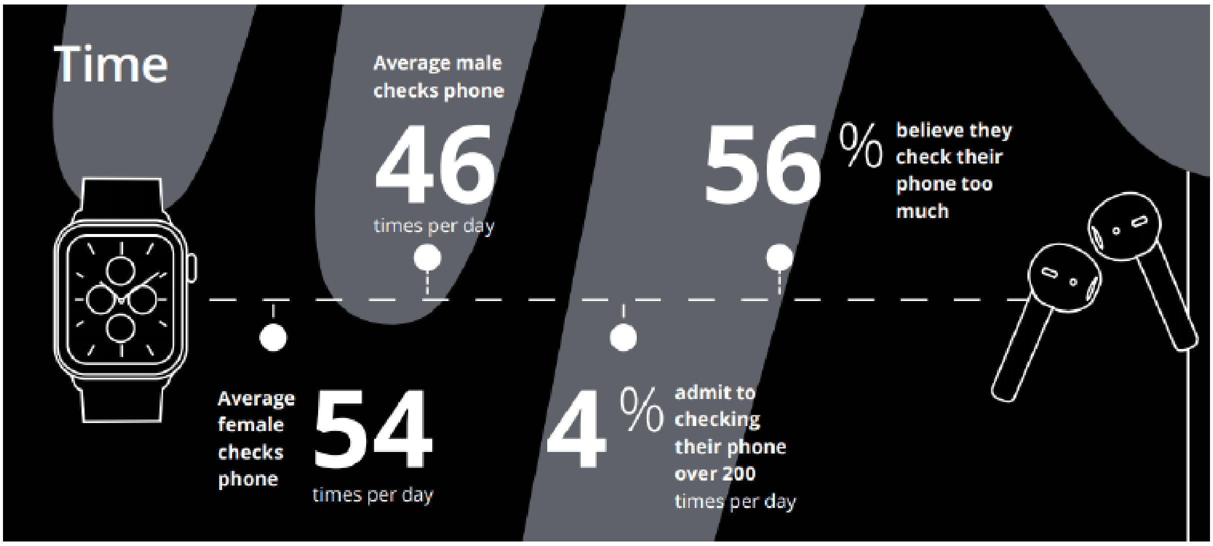


Figure 1.1: Distribution of Phone usage in Ireland

Push-notifications can be annoying (2), since there are multiple apps that send notifications throughout the day and looking at every notification can reduce a person's productivity (3).

Due to this reason, many people nowadays keep their phones on silent or the push-notifications turned off for apps that are not that important. This is a major loss for the companies associated with the apps, as notifications are a way for companies to interact with their customers and let them know about new offers or any exciting upcoming updates that could lead to an increase in their sales.

Another important thing to consider is users missing out on critical notifications, like an important reminder, or an emergency text received, due to the insurge of too many push notifications and no mechanism to filter out the important notifications from the others. This study's core objective is to identify the moments of boredom in a mobile phone user and build a model that can subsequently help find opportune moments for the delivery of a notification. The goal is to utilise moments of boredom to identify whether current factors may lead to the right moment for interacting with the notification quickly and hence increasing the engagement rate.

A human being goes through multiple emotions in a day (4), due to numerous factors, such as based on the time of the day, if it's after evening or the weekend the person could be perceived as being free and therefore bored assuming they have a day job, some type of content in messaging/ news apps could lead to humans being enraged or happy.

Hence, it becomes important to find opportune times at which the user can be sent push-notifications and quality content for them to interact with which would lead to eventual increase in sales and customer-base. An opportune time can be defined as moments when the user is bored, having lower productivity, and would be open to reading notifications and interacting with them instead of ignoring it.

Numerous factors come into play when deciding if a user is bored in a moment of time and will thus interact with the notification in given time. To better understand the behaviour of users we can utilise Data Analytics as well as Machine Learning models which will help in classifying moments of boredom of mobile users. This model will be useful for marketing companies as well as individuals.

The difficulty with this type of research is gaining access to gather in-the-wild and open data, as most of this data is controlled by a few large companies like Apple, Samsung, Google, who would not be willing to share the mobile usage datasets of their customers due to numerous reasons, a major one being privacy concerns, as location and message content can easily be misused if it falls in the wrong hands. Through this research we take a deeper look at different mobile phone usage features and try to gauge their impact on user's boredom levels.

A solution found for this problem was to utilise existing datasets available that gave information about users mobile phone usage with various parameters like number of notifications received, time of the day those notifications were received, type of apps used, and so on. While it was difficult to find such datasets that did not cause any privacy issues or violate GDPR, one such dataset was found that was part of an experiment conducted by Martin Pielot and his team.

This particular dataset categorises moments of boredom in users based on features like time of the day, day of the week, type of application, and so on, along with ESM (Experience Sampling Methodology) and impose it on a different synthetic dataset that has similar features as Martin's dataset such as time of day, day of week, type of app, etc. and find if user is bored in that moment of time. Although there is no way to be 100% sure of the boredom classification, it is assumed that the boredom classifier obtained from the learning model on Martin Pielot's dataset is accurate and would be an apt model to judge moments of boredom on other datasets having similar features.

If mobile phones are able to detect the moments of time when users are bored, then they could suggest alternate things that can be done to better utilise those idle moments,

- Proposing content, projects or utilities that may help in conquering boredom
- Encouraging catching up on TO-DO lists, to be read lists, or participation in research surveys
- Allowing users to use the downtime for contemplation to enable mental downtime, as it's of utmost importance to learn, reflect and foster creativity.

1.2 Research Question

The research question that this thesis aims to answer is :

"Is it possible to build a boredom classifier for mobile phone users based on one model with limited context features and use it to make boredom predictions on another model with the same features? "

1.2.1 Research Challenges

Dealing with mobile phone users data is a humongous task, as it involves numerous challenges. The first step in this process is dataset collection, and finding datasets with features similar to the features considered for the boredom prediction classifier task. It is important to use datasets that have been collected without violating the GDPR or any other privacy and security of end-users. Analysis of the datasets and finding out which features are important for classification of boredom is the next big task, as it is important to factor that some features that denote a user as being bored in that instant of time may be missing in the dataset that it needs to be implemented in. To make the original dataset and the dataset the model needs to be evaluated on similar, the features need to be homogenised.

1.2.1.1 Challenges in Obtaining Mobile Phone Usage Data and Estimating User Interactions with Applications

Mobile environments and their sensors, data prove to be very difficult for researchers, due to which many researchers focus on introducing new methods of collecting data from users instead of putting efforts in conducting studies that collect data from users "in-the-wild", meaning real world data. Some of the innovative ways in which researchers collect data from participants are : context-aware ESM (5, 6, 7, 8), video collection (9) and more. Some researchers concentrated on building sensory techniques (10, 11), while some built mobile sensing frameworks to ensure uninterrupted mobile usage and sensor data collection.

1.2.1.2 Feature Space

Another challenge with using a mobile phone usage dataset is the wide variety of input feature types collected using mobile sensors and metrics. It becomes difficult to find the exact features and the right number of features to be considered for building a boredom classifier such that an accurate model can be built, that can then be used to find out boredom in other users based on mobile phone usage patterns.

1.2.1.3 Privacy and Ethics

Mobile phone usage logs contain data such as location data, and more PII which if leaked could cause a breach of privacy for a user and thus an explicit permission is needed to collect data as well as store the data and use it. It is crucial to handle the user mobile phone usage data securely, by ensuring that data breach does not take place and only people who are trusted and at higher levels have full access to user data.

The dataset by Martin Pielot and his team (4) was collected from an app launched specifically for the research, Borapp on Google Play store. In the preliminary stages it was explained to the user what kind of data would be collected, the place where it would be stored, how it would be used, clearly stating any PII(Personally Identifiable information) being collected. Prior to this, users were asked for their consent to partake in the experiment. Only after all permissions were received did the ESM(Experience Sampling Method) take place, which is explained more in Chapter 4 Section 1.

1.2.1.4 Time of Delivery of Notification

While it is important to understand the impact the time when a notification is delivered has on user receptivity, there have been studies with contradicting results which will be discussed in this section. A study conducted by Iqbal and Bailey (12) demonstrated that emails delivered at moments when a user has just completed a task requiring some thinking, makes them more receptive to interacting quickly to the emails and also helps in reducing frustration.

In contrast, a field study conducted by Fischer with 11 of his co-workers (13) showed that whether or not a user would interact with a notification depends on the contents of the message, namely if the notification received is interesting, of relevance to the user, if it has certain entertainment value. However, for their study they came to the conclusion that the time of receiving the notification had no effect whatsoever on the reception of the notification by users.

Looking at these two studies and their results, we can not come to a conclusion about the impact time of delivery of notifications has on users' receptivity.

A possible reasoning, as found by Mehrotra et al. (14), for this discrepancy is that the notification content is a major deciding factor in whether the user will dismiss the notification or open it. In their study, the instances of users saying they dismissed the notification because they were busy with something at that moment are rare. Thus, it can be concluded that higher precedence was given to looking at the notification than whatever task the users were performing, if the notification contents were deemed valuable by the user looking at the value.

Thus, it can be concluded that higher importance is given to notifications providing value to users over any tasks or interruptions at hand. It further strengthens the suggestion that notifications that are not majorly important can be delivered at a time when the user is bored, thus ensuring effective management, and higher engagement with the notifications than with other tasks at hand.

1.2.2 Research Overview

Through these researches it can be inferred that attention and openness to interruptions can be evaluated from:

- time since recent usage of device
- using of specific services such as internet browsers, email inbox, calendar
- user activity context (differences between still and moving)
- time-such as the hour of the day or the day of the week
- proximity, i.e., if a mobile phone's screen is free or covered (indicating if the phone is stowed away)

In conclusion, these studies indicate that our interaction with technology is affected by level of attention, openness to interaction and boredom levels.

In the next chapters we will first take a look at the different research done around empathy based notifications for mobile phone users, then we discuss about the methodology used for the research, where we discuss about the different Classification Machine learning algorithms used for building a boredom classifier, then we will look at the Evaluation methodology used for evaluating performance of the ML models, then we talk about the Data preparation and feature analysis which talks about the Datasets used, Data pre-processing, analysis of the prepared data. Finally we talk about the results and findings for this research, taking a look at visual representations of the results. In the final chapter, we take a look at an overview of the research, the primary findings, limitations of the research and also discuss the future work that can be done based on the research.

2 Literature Review

Digitization has propelled researchers to focus more on the ever-increasing use of mobile phones and analyse the usage patterns, the impact on the emotional state of the user, with the ultimate goal of increasing the productivity of end-users. The following section will discuss recent research and past work in mobile phone usage and the impact on emotional and mental wellbeing of mobile users, based on mobile phone usage.

2.1 Boredom and its Detection

Boredom can be said to be "lack of stimulation or inability to be stimulated" (15) and the resulting displeasure. It can also be defined as a "pervasive lack of interest and difficulty concentrating on the current activity" (16). According to Eastwood (17), "a bored person is not just someone who does not have anything to do; it is someone who is actively looking for stimulation but is unable to do so".

Feeling bored inevitably goes with the desire to escape such a state (18). Likely benefits of boredom include the start of creative processes and self-reflection (16). There is a huge commercial value in knowing when a person is bored, as it has been learned that people who are bored crave stimuli and that human attention has become scarce and thus highly valuable (19).

The most popular way to detect boredom, according to Bixler and D'Mello (20), is through facial expressions, speech, text, and physiological signals. They explored boredom detection by taking a log of writing keystrokes during a task given to users to write, and came to the conclusion that although keystrokes alone had low predictive accuracy of boredom, around 11% for engagement-neutral and boredom-neutral states, when stable traits of participants were added to the model it helped improve prediction accuracy of boredom.

According to a study conducted by Guo et al. (21), when users are performing a web search, there are a number of events that allow the prediction of user's openness to be distracted from their primary task, such as movements of the mouse, clicks, page scrolls, and more, which might be indicators of boredom.

Recent studies have been made by Mark et al. (22) around variance of attention and subsequent boredom in the workplace. In an in-situ study over a period of 5 days, they kept a track of computer activity of 32 information workers and around 20 times a day they probed the effect. Their findings indicate that boredom is related to a large number of factors, some of which are the time of the day, computer interaction patterns like frequency of switching the window.

2.2 Mobile Phone Usage - Inferring Emotions

When we consider mobile phones, there have been numerous studies that show a definite link between emotions and mobile phone usage. Bogomolov et al. showed that it is possible to infer daily stress (23) and daily happiness (24), through mobile phone usage data, users personality traits and also data related to weather. LiKamWa et al. (25) showcased that by monitoring social interactions through SMS, Email, Phone calls and routine activities like application usage, day-to-day mood (valence and arousal) can be inferred. Exhaustive research has been done on utilising the sensors on a mobile device to help learn about the state of attention of the user, like the amount of interruptions a person is open to getting.

Some studies have shown that computing devices can be used to detect a person's willingness to receive office visits (26), emails (12), messages from desktop instant messengers (27), SMS and Mobile phone usage (28), Mobile phone alerts (29), and phone calls (30, 31).

Through this research(4) it can be inferred that attention and openness to interruptions can be evaluated from: time since recent usage of device](26, 27); using of specific services such as internet browsers, email inbox, calendar (22), time-such as the hour of the day or the day of the week (26, 27, 30); and proximity, i.e., mobile phone screen being covered or in use (21, 24).

In conclusion, these studies indicate that our interaction with technology is affected by level of attention, openness to interaction and boredom levels.

2.3 Feature importance for predictions

A major challenge with analysing boredom levels of mobile phone users is the wide variety of input feature types present in the mobile phone usage dataset. In order to determine mobile phone usage patterns and infer boredom, mobile sensor data can be used. Many of these sensors were determined to be predictors of boredom in user by Martin Pielot and his team (4), these features are “screen on/off, hour of the day, day of the week, ringer mode, location, user activity context, number of notifications received, time since last outgoing call, time since last SMS sent/ received, gender, proximity, light, age, time since last notification was received, apps per minute, time since last unlock, battery level, battery drain, bytes received/transmitted”. It’s possible to use a machine learning model to use these features effectively to build a boredom classifier that can accurately predict boredom based on mobile phone usage. ML models can use large feature spaces, and provide high performing models with good accuracy.

2.4 Ethics and Privacy Considerations

There are privacy and ethical difficulties surrounding the accumulation of datasets for mobile phone usage. Mobile phone usage logs can contain sensitive information such as location data, and more PII and thus require ethics permission for collection and storage.

Storage of this sensitive data requires high levels of data security and strict user access restrictions so that only individuals who are authorised to use that data have access to it. If a data breach were to occur with mobile user data it would infringe on the privacy of the users who participated in data collection.

The dataset by Martin Pielot and his team (4) was collected from an app launched specifically for the research, Borapp on Google Play store, wherein participants were asked for explicit consent to their participation in the study. The background of the study, kind of data being collected, how and where the data would be stored, and how it would be used was all explained in the preliminary stages. Once consent was obtained, the app collected data and triggered probes via ESM(Experience Sampling Methods).

Since this dataset is available publicly, no explicit ethics and privacy requests were required from this research’s perspective. The second dataset used to test the boredom hypothesis obtained from Pielot’s dataset, is a synthetically prepared dataset by Kieran Fraser (32), which does not impact users’ privacy in any way as the PII like location of user and notification context was omitted from the synthetic dataset.

2.5 Synthetic Notifications Dataset

An artificial or synthetic dataset was used for the evaluation of the boredom classifier developed as a part of this research. This dataset was created by Fraser (32), as a part of his study. For evaluating the dataset, three machine learning models were used namely Linear SVC, Logistic Regression and a random forest classifier that were used to run the classifier on the synthetic dataset. To evaluate the performance of these models on an "in-the-wild" dataset collected by Martin and his team (4), different metrics like accuracy, precision, F1-score and AUC-ROC curve were used.

According to Fraser (32), the possible disadvantages of using a synthetic dataset is that it is not able to capture subtle variances in the data as opposed to a real-world dataset. Compared to the real dataset, the dataset has fewer unique subjects, places, and apps, suggesting that the generative model could not learn a holistic view of these characteristics.

3 Methodology

This section discusses about the flow of the research, the different datasets used, the data pre-processing, data transformations, model training, hyperparameter tuning and result visualization. We also take a brief look at the different Classification Machine Learning Models used,and the evaluation metrics used for measuring the performance of these algorithms.

3.1 Flow of the process

The Literature Review helped us come to the conclusion that attention, boredom and proneness to receiving notifications or interruptions depend on different factors which can be collected from mobile phone usage patterns of users. Some of the features affecting proneness to interacting with notifications are: time of the day, day of the week, type of application from which user receives notification, type of activity being performed by user at time of reception of the notification, if the mobile screen is locked or unlocked at the time notification is received.

As can be seen from the flow diagram below Fig 3.1, we use two datasets one is the training dataset, Borapp dataset, on which the three ML models are trained and the other dataset is the WeAreUs dataset which is the test dataset. Primary aim of this study is to build a boredom classifier on the training dataset and use it to classify moments of boredom in the test dataset. Firstly, We perform Data pre-processing on both the datasets to make them homogenised with each other, handle missing values and null values, one-hot encode the categorical variables to use them for ML models. Then we train the model on the training dataset. After saving the model for the 3 different classification models, we perform hyperparameter tuning on the models to improve their performance. The final step on the training dataset is to perform Result evaluation, which is done using confusion matrix, F1-Score and accuracy, AUC-ROC curve. All these metrics help in understanding the performance of the trained model. Finally, we predict the test dataset, and visually represent the results of moments of boredom in WeAreUs dataset.

The diagram below shows the flow of the study,

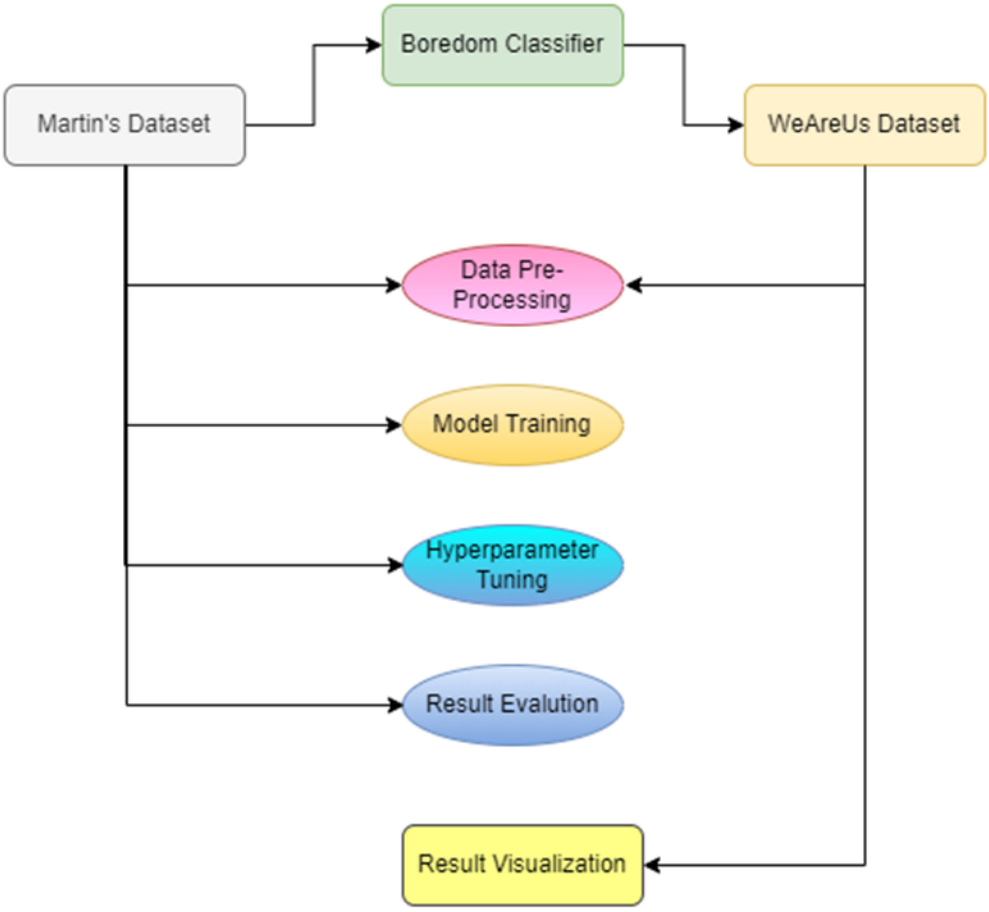


Figure 3.1: Process flow - Boredom Classifier

3.2 Classification Algorithms

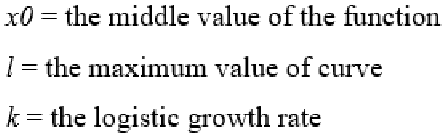
3.2.1 Logistic Regression

Logistic regression is a kind of supervised learning, statistical model that is very often used for classification as well as prediction analysis. The model estimates the chances of some event occurring, on the basis of the dataset provided having some independent variables. As the Logistic regression calculates probabilities, the dependent variable which is the outcome variable can have binary discrete values only. (33)

Logistic regression can be represented by the following function:

$$f(x) = \frac{l}{1 + e^{-}k(x - x_0)}$$

where,



x0 = the middle value of the function

l = the maximum value of curve

k = the logistic growth rate

Logistic regression estimates the relationship between one dependent and one or many independent variables, but it is majorly used for estimating the predictions of categorical variables, which can have only values like 1 or 0, Yes or No, True or False, and so on. Logistic regression makes use of negative log-likelihood as the loss functions using gradient descent process that helps in finding the global maximum.

Some drawbacks of logistic regression are that it can be subject to overfitting, especially when there is a large number of predictors in the dataset. To avoid overfitting, Logistic regression models are regularised in order to penalise the large coefficient parameters to make sure the model does not suffer from high dimensionality.

3.2.2 Random Forest Classifier

Random forest is a type of ensemble machine learning algorithm used to combine output received from multiple decision trees to form a singular result. It is very flexible and easy to use because of which the algorithm is very popularly used for classification as well as regression problems. (34)

Random forest requires the hyperparameters node size, number of trees and features size to be set before training the model, and the model can then be used to perform classification or regression problems. The Random Forest contains many decision trees and each tree in this ensemble contains a small number of datapoints taken from the training dataset with minor changes, these data points are called bootstrap samples. After this, a train-test split is performed on the training data and is used as the test dataset for validation of the trained model. Feature bagging adds some more randomness to the data which helps in reducing the correlation between the different branches in the decision tree. Finally, cross validation is performed to form the predictions.

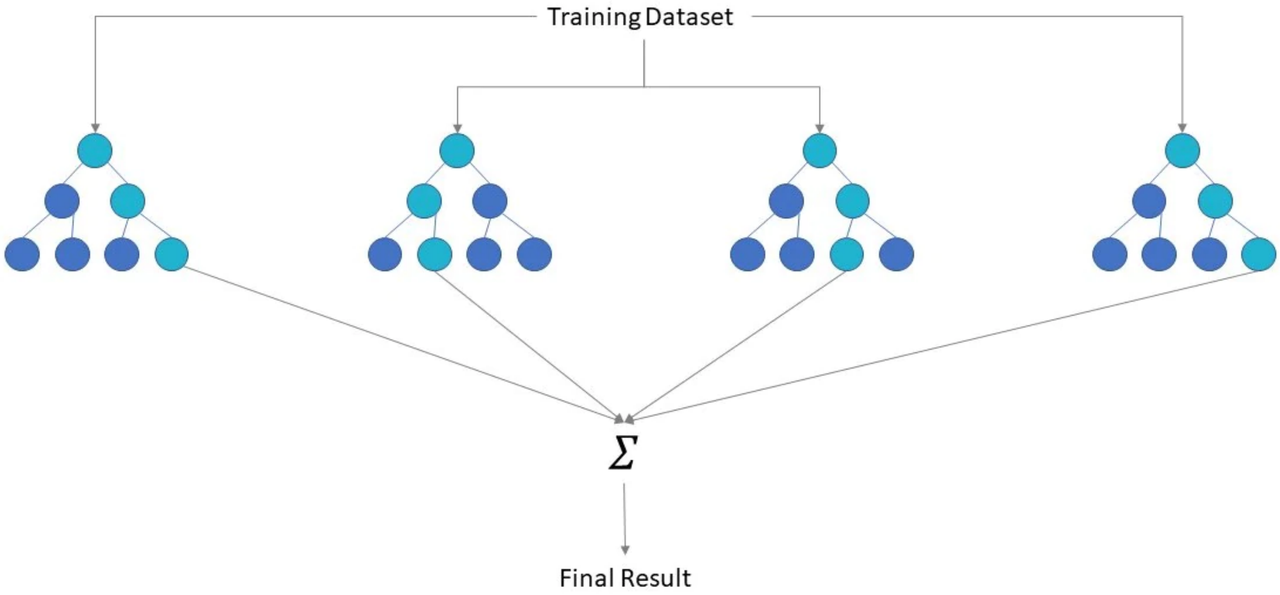


Figure 3.2: Random Forest Classifier

3.2.3 Linear Support Vector Classifier

Support Vector Classifier maps data for easier categorization into a high-dimensional feature space. It looks for a separator in the categories formed and then transforms the data using the separator as a hyperplane. Subsequently, the features of the new data are used to make predictions of the group in which to categorise the new record. (35)

SVC is a vigorous model that performs classification and regression in such a way as to maximize the prediction accuracy, and avoids overfitting the training data. SVC works best with big datasets having a large number of predictor variables. SVC has many different applications such as CRM(Customor Relationship Management), Image recognition and face recognition, bioinformatics, intrusion detection, speech recognition and many more.

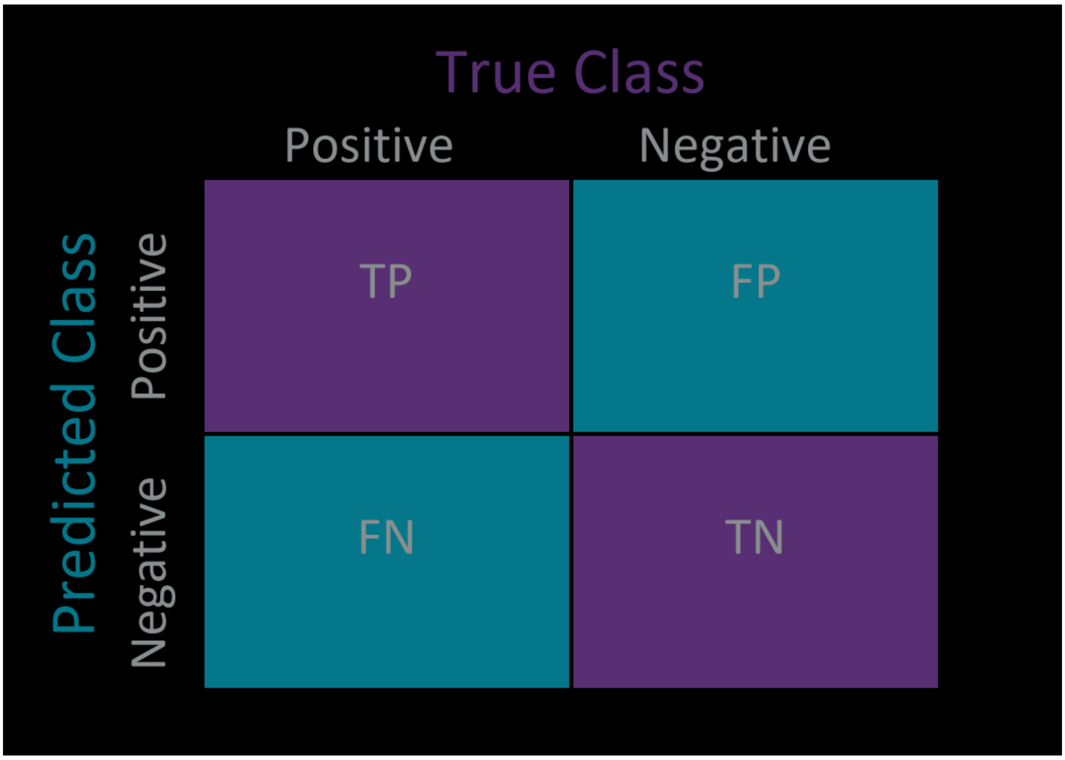
A kernel function is used for transformations, which is a mathematical function. SVM supports the following kernel types:

- Linear
- Polynomial
- Radial basis function (RBF)
- Sigmoid

3.3 Evaluation Methodology

It is crucial to measure the quality of machine learning algorithms models. It depends on the type, implementation, the hypothesis to be tested and context of the model to decide which metric will be used. Improved predictions lead to better metrics score and due to this reason it becomes necessary to tune the model correctly. The next section discusses evaluation metrics for the classification models we used in our study.

3.3.1 Confusion Matrix



		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Figure 3.3: Components of Confusion Matrix

Confusion matrix is an evaluation method used in classification problems, and it is a square N sized matrix, where N is the number of dependent variable classes. It is a convenient way to find out the expected and predicted value counts based on the type of category.

The matrix cells can be explained as follows:

- True positive (TP): When expected and predicted values are both true.
- When expected and predicted values are both negative.
- When expected is false, but prediction is categorised as true.
- When expected is true, but prediction is categorised as false.

3.3.2 Recall or Sensitivity

The recall is a measure of actual positive cases correctly predicted by the model divided by the true positives and false negatives. Recall is beneficial when we have a scenario where false negatives are a lot, and we want to identify the highest possible number of true positives. When used alone, Recall is not a good enough evaluation metric of the model.

The recall fraction is:

$$\text{Recall} = \frac{\text{True Positives (TP))}}{\text{True Positives (TP))} + \text{False Negatives (FN))}}$$

3.3.3 Precision or Positive Prediction Value (PPV)

Precision is a measure of positive instances in our model that have been accurately predicted out of the total number of false positives and true positives. This measure is beneficial when there is a high cost of false positives and low cost for false negatives. It can be represented by a formula as follows:

$$\text{Precision} = \frac{\text{True Positives (TP))}}{\text{True Positives (TP))} + \text{False Positives (FP))}}$$

3.3.4 F1 Score

F1 score is the harmonic mean of recall and precision and it gives us a value that is the best amongst the two. The reasoning behind choosing harmonic mean instead of arithmetic mean is because the harmonic mean provides better penalization of extreme values.

The equation for F1-score is given below:

$$F_1 = \left(\frac{\text{Recall}^{-1} + \text{Precision}^{-1}}{2} \right)^{-1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

3.3.5 Area Under the Receiver Operating Characteristic Curve (AUC-ROC)

ROC or Receiver Operating Characteristic Curve is a plot between sensitivity and specificity (ratio of true negative cases) . AUC or Area Under the Curve is a singular representation of ROC, with values lying between 0.5 (failure of model) and 1 (overfitted model).

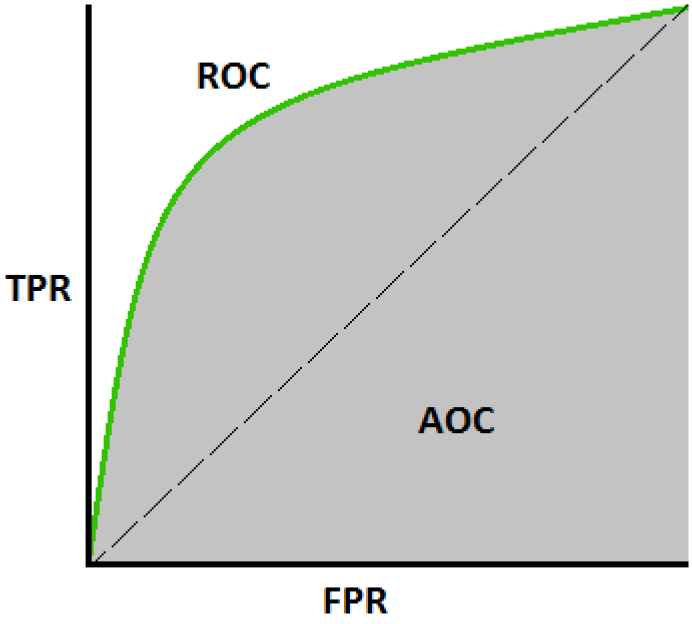


Figure 3.4: AUC-ROC Curve Example

4 Data Preparation and Feature Analysis

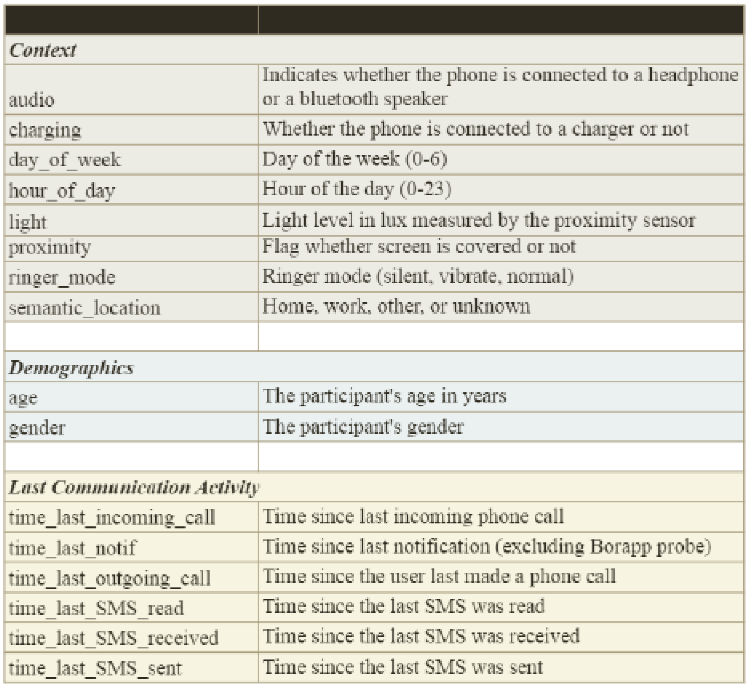
4.1 Datasets Used

For this study two datasets were used, the statistics of both datasets are discussed below.

4.1.1 Borapp Dataset

Martin Pielot and his team led an experiment in 2014 with 54 participants to find out if mobile phone usage is affected by boredom levels of users or them seeking stimulation, and if this is true finding out the accurate indicators of boredom in mobile phone usage. The participants had to install a dedicated application called Borapp from the PlayStore, which was free to download, under the condition that they had to use it for a minimum of 14 days. The mobile phone usage patterns were collected from the phones sensors as well as the event listeners. The app was designed to run on Android phones with OS 4.0 and above.

There were two kinds of data being collected, one type was collected from the phone when the screen was unlocked and turned on, and the phone was being used, the other type of data was background data, which was collected all the time. In the preliminary stages it was explained to the user what kind of data would be collected, the place where it would be stored, how it would be used, clearly stating any PII(Personally Identifiable information) being collected. Prior to this, users were asked for their consent to partake in the experiment. The final dataset collected contained around 43 million mobile phone usage records taken from 54 mobile phones.

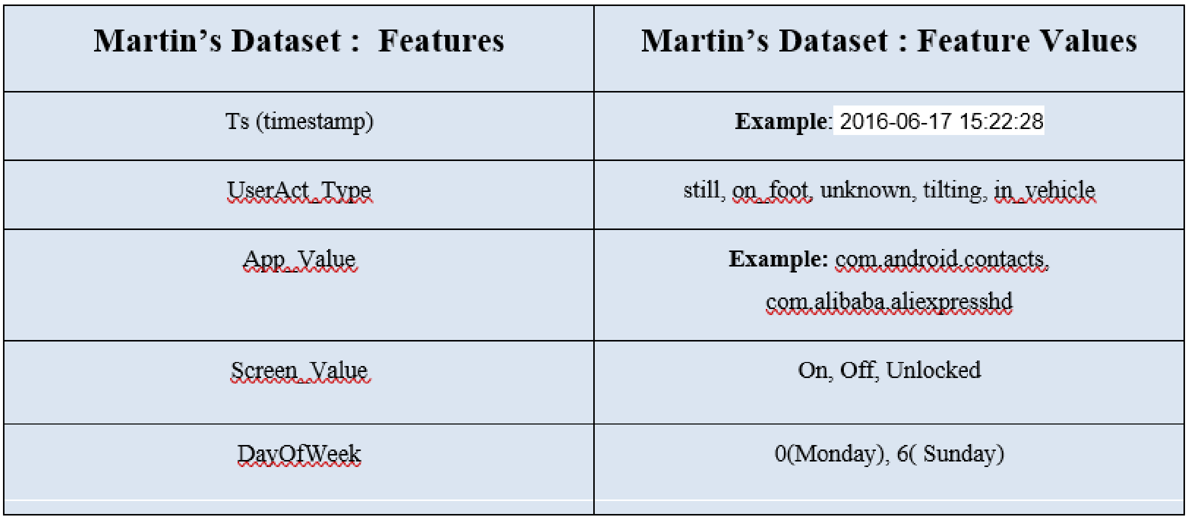


Context	
audio	Indicates whether the phone is connected to a headphone or a bluetooth speaker
charging	Whether the phone is connected to a charger or not
day_of_week	Day of the week (0-6)
hour_of_day	Hour of the day (0-23)
light	Light level in lux measured by the proximity sensor
proximity	Flag whether screen is covered or not
ringer_mode	Ringer mode (silent, vibrate, normal)
semantic_location	Home, work, other, or unknown
Demographics	
age	The participant's age in years
gender	The participant's gender
Last Communication Activity	
time_last_incoming_call	Time since last incoming phone call
time_last_notif	Time since last notification (excluding Borapp probe)
time_last_outgoing_call	Time since the user last made a phone call
time_last_SMS_read	Time since the last SMS was read
time_last_SMS_received	Time since the last SMS was received
time_last_SMS_sent	Time since the last SMS was sent

Figure 4.1: List of features related to context, demographics, time since last activity

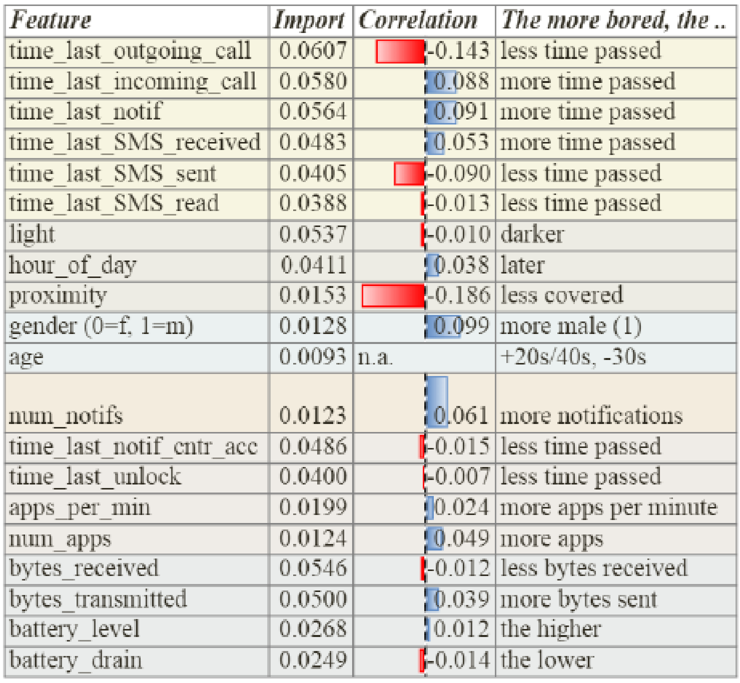
35 Features were extracted that were directly related to phone-usage patterns in 7 categories: context, demographics, time since last activity, intensity of usage, external triggers, type of usage, idling. (4) The figure above lists some features related to context, demographics and time since last activity.

Out of these 35 features, the 5 features selected for this research are as follows:



Martin's Dataset : Features	Martin's Dataset : Feature Values
Ts (timestamp)	**Example**: 2016-06-17 15:22:28
UserAct_Type	still, on_foot, unknown, tilting, in_vehicle
App_Value	**Example:** com.android.contacts, com.alibaba.aliexpresshd
Screen_Value	On, Off, Unlocked
DayOfWeek	0(Monday), 6(Sunday)

Figure 4.2: Table of Martins Dataset Features and their values



Feature	Import	Correlation	The more bored, the ..
time_last_outgoing_call	0.0607	-0.143	less time passed
time_last_incoming_call	0.0580	0.088	more time passed
time_last_notif	0.0564	0.091	more time passed
time_last_SMS_received	0.0483	0.053	more time passed
time_last_SMS_sent	0.0405	-0.090	less time passed
time_last_SMS_read	0.0388	-0.013	less time passed
light	0.0537	-0.010	darker
hour_of_day	0.0411	0.038	later
proximity	0.0153	-0.186	less covered
gender (0=f, 1=m)	0.0128	0.099	more male (1)
age	0.0093	n.a.	+20s/40s, -30s
num_notifs	0.0123	0.061	more notifications
time_last_notif_cntr_acc	0.0486	-0.015	less time passed
time_last_unlock	0.0400	-0.007	less time passed
apps_per_min	0.0199	0.024	more apps per minute
num_apps	0.0124	0.049	more apps
bytes_received	0.0546	-0.012	less bytes received
bytes_transmitted	0.0500	0.039	more bytes sent
battery_level	0.0268	0.012	the higher
battery_drain	0.0249	-0.014	the lower

Figure 4.3: Most important features obtained from the primary dataset sorted by their mean impurity decrease score. More positive (blue) correlation values are interpreted as "higher the value more bored"

Figure 4.3 describes the most important features found from the Borapp dataset by using Random Forest machine learning models implicit feature importance by mean impurity decrease scores. The blue side indicates positive values meaning more correlation that can be interpreted as "Higher the value more bored". Whereas, the red side is used for indicating negative values, meaning low correlation which denotes "Higher the value less bored".

4.1.2 WeAreUs Dataset

The WeAreUs dataset was made using "in-the-wild" notifications that were collected from real mobile users. An app named WeAreUs (32) was used to collect the notification data, which used two ways to collect the user data, one was an ESM (Experience Sampling Method) and the other way was to use background sensors (32). The background sensors used Android SDK's Notification Listener Service (Google LLC) and the mobile sensors hourly logs that collected data with no human interaction at all. A majority of the data was collected using the 2nd method.

An active interaction between user and the ESM was needed for the data collection, which prompted users by asking them questions about notifications and the current context (32) after there was an interaction between the user and notification. It also prompted the user to share some insight into their state of mind at the moment and immediate context by asking them questions through ESM on screen being unlocked. (32)

The data collected through ESM was lower in quantity than the background data collected, but this data was rich in quality of depth provided with respect to the context of the users and their receptivity to notifications.

A combination of these 2 data collection methods from "15 participants (2 Female and 13 Male), ranging in ages from 21 to 64"[18] collected "Over 30,000 in-the-wild notifications [. . .] as well as 4,940 smartphone general-usage logs and a total of 291 ESM questionnaires [. . .] answered by participants" (32)

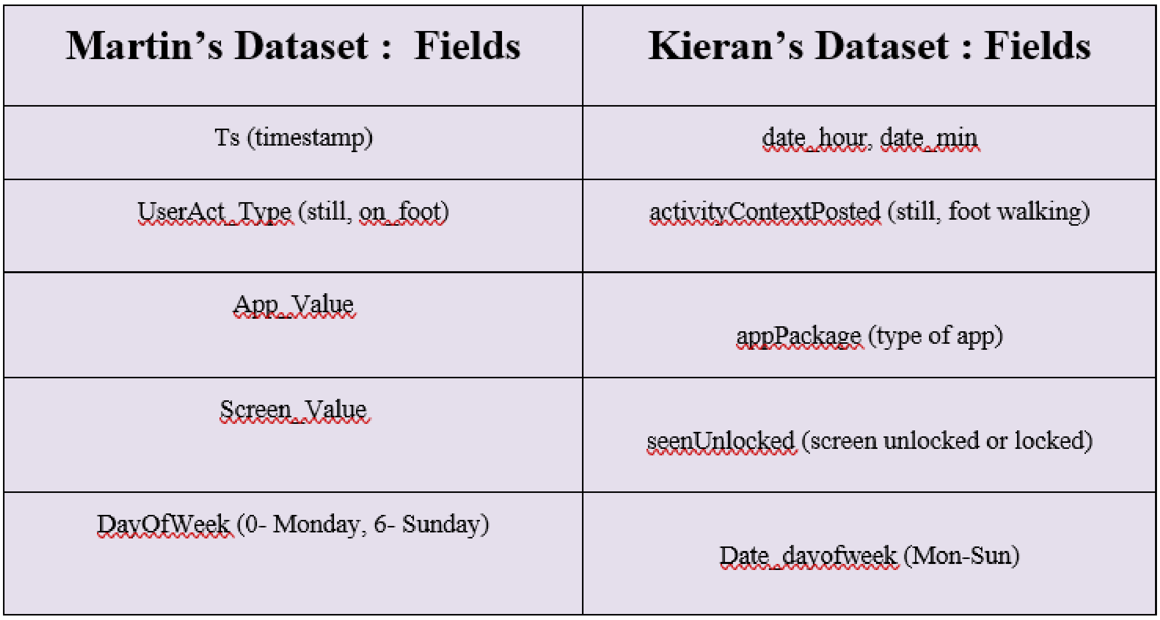
Using the WeAreUs dataset as real data, synthetic notifications were generated through the generator component of a Generative Adversarial Network (GAN), giving real and generated values in alternate patterns (32). Care was taken to have the synthetic data be as close to real data as possible so much so that the discriminator of the GAN would be unable to differentiate between the real and synthetic data.

4.2 Data Preparation

Moments of Boredom, according to Martin Pielot's study are:

- More time has passed since receiving phone calls, SMS, or notifications, and less time has passed since making phone calls and sending SMS. This finding suggests that being contacted by others is generally correlated with being less bored. Contacting others, however, is more likely to happen while being bored.
- Boredom further correlated with the intensity of mobile phone use. In general, higher the usage intensity, the higher the boredom.
- Boredom positively correlated with the time of the day and darker ambient lighting conditions. This finding means that there are boredom levels that vary throughout the day, boredom tends to increase as the day progresses.
- Apps that most strongly correlated with being bored are Instagram, email, settings, the built-in browser, and apps in the 'other' category. Apps that correlated most strongly with not being bored were communication apps, Facebook, SMS, and Google Chrome.

While reading data from the Borapp and WeAreUs datasets, we filtered out the columns that were present and similar in both. This gave us the following features:



Martin's Dataset : Fields	Kieran's Dataset : Fields
Ts (timestamp)	date_hour, date_min
UserAct_Type (still, on_foot)	activityContextPosted (still, foot walking)
App_Value	appPackage (type of app)
Screen_Value	seenUnlocked (screen unlocked or locked)
DayOfWeek (0- Monday, 6- Sunday)	Date_dayofweek (Mon-Sun)

Figure 4.4: Table of Martins Dataset features and Kieran's Dataset features

These parameters can be taken into consideration while gauging the boredom, as listed by Martin's moments of boredom.

- More time for received call duration than calls made, would mean less degree of boredom.
- Weekends and night time would mean more free time, and a higher degree of boredom.
- Appvalue or the type of applications would also influence the measurement of boredom of the users.
- If users are still, they have a higher chance of being bored than when they are in motion.

4.2.1 Data Cleaning and Exploratory Data Analysis

For making the datasets, Borapp and WeAreUs, homogeneous, we have done a number of data cleaning activities listed below:

First we will have a look at the data cleaning techniques done on both the datasets:

Data cleaning done on both datasets

4.2.1.1 Filtering out unneeded columns while reading data

For both datasets (Martin and Kieran), we only take into consideration certain columns that are common or similar in both and therefore useful in our goal of finding the boredom metric in users. The columns are as follows:

- Hour of the day
- Day of the week
- Type of App
- Mobile screen locked or unlocked
- Type of activity performed by user

4.2.1.2 Used One-Hot encoding for categorical data

After doing a train-test split on Martin's dataset, and obtaining the training and testing data, we performed One-Hot encoding on all the columns. All of our feature columns are categorical data, meaning they are variables containing labels instead of numbers as value. However, there is a fixed set of combinations of values possible for each of the variables. The downside with using categorical data directly is that many machine learning models do not support direct operations on labelled data, as they need the input as well as output variables to be numbers. For converting categorical data to numeric, we use one-hot encoding, which instead of adding integer encoded variables adds a new binary variable for every unique int value.

We use the function:

pandas.get_dummies(df) - *which converts categorical variables into dummy or indicator variables.*

The algorithm used for One-Hot Encoding was as follows:

1. one_hot_cols = [Set of all the Features that are being considered for building the model]
2. Rename the columns in WeAreUs dataframe with column names in BorApp dataset to ensure One-Hot encoding can be done as the column names are mapped while doing one-hot encoding.
3. We set inplace = true, so that the column names get replaced in the original dataframe.

4. Using difference(), we first find out the columns missing in the training dataset that are present in the test dataset and save it in a dataframe.
5. Again using difference(), we find the columns missing in the test dataset that are present in the training dataset and save it in a dataframe.
6. Using drop(), we drop the columns in the training dataset obtained from Step 4, and are left with columns in the training dataset that are synchronised with the test dataset columns.
7. Similarly, we use drop() to drop the columns in the test dataset obtained from Step 5, and are left with columns in the test dataset that are synchronised with the training dataset columns.
8. Finally, we use One-Hot encoding with the function pd.get_dummies and pass the dataframes that have been modified in Steps 6 and 7, to be synchronous, passing columns as mentioned in Step 1.

4.2.1.3 Missing Data Imputation in both datasets by handling null values, missing categories

After performing One-Hot encoding on both the datasets by homogenising the columns, there is one more step necessary to be performed before the training of the model can be done. This step is finding out the user entries that have missing values such as NaN, and missing categories after doing one-hot encoding and subsequently replacing the missing values by handling them as follows:

1. After one-hot encoding done on both datasets, we check using the function difference(), the columns in the test dataset that are missing in the training dataset, as the test dataset is the one on which we will be evaluating the trained model.
2. Using drop(), we remove the columns present in the one-hot encoded dataframe of the test dataset that are not present in the one-hot encoded columns for the training dataset.
3. In the training dataset, we use the difference() function to find one-hot encoded columns missing in the training dataset present in the test dataset and store it in a dataframe.
4. We then take the list obtained in Step 3 and replace the extra columns in the test dataset by 0.
5. This is how we handled the missing data and missing categories in the test dataset.

4.2.1.4 Categorising apps to achieve homogeneity in both datasets

Based on the type of app, we classify the different apps in the dataset into categories like “Social Media”, “Utility”, “Mail”, “Unknown” and more.

Since the number and type of applications being used by different users was a huge list that was making it difficult to accurately classify the apps, I decided to categorise the apps for easier classification. A crawl script was run using the app names on PlayStore that helped extract the metadata containing categories of the apps. This was done for the apps used by the users in Martin’s dataset, and a total of 32 categories were retrieved, which was added as a new column “type” through code in the data frame containing information about all the users mobile phone usage.

Since the data collected by Martin was in 2016, many of the applications being used by the users had following issues while trying to extract type metadata from PlayStore:

1. The information was not available on the PlayStore or

2. The app is no longer available for download or

3. The app package was changed, and so we can no longer link it to a PlayStore listing.

All of these apps were put into the “Unknown” category. Many records did not have any app entries, and the number of such records was actually quite significant around 56.7%. These NaN values were also replaced with “Unknown” as the app type.

For the WeAreUs dataset, there was a column/feature called “topic”, that tells us the category of the application, which was very close to the apps category obtained from PlayStore. So through code, we categorised the apps into 28 app types for the WeAreUs dataset.

Data cleaning done on the WeAreUs dataset

4.2.1.5 Eliminating “Activity” having unknown values

For the column “activityContextPosted” in WeAreUs dataset, there are three possible values that are indicative of the type of activity being performed by the user at the time of the record being taken, we have “Still”, “foot walking” and “unknown”. We eliminate the rows of data for all users having “unknown” as the entry for activity context, as this is not helpful in determining the boredom of a user.

Gauging boredom from activity being performed by user at that instant of time is done as follows: If a user is stationary then there is a high chance that they will check their phone for any notification. On the other hand, if a user is walking and in motion, then the chance that they will check their phone is low. So, "still" is classified as bored state, and "foot walking" as a not-bored state.

4.2.1.6 Changing date_dayofweek from Sun-Mon to 0-6 in WeAreUs dataset

The day of the week is indicative of boredom in the sense that people are more busy on the weekdays usually, i.e., from Monday to Friday and are less likely to check their phones much whereas on weekends people are usually glued to their phones and the chances of them using their phones is higher.

Since the Day of week had different values in WeAreUs dataset as Sun, Mon, Tue and so on, whereas the Day of the week was in format 0 to 6, where 0 stands for Sunday, 1 for Monday, etc., we replaced the date_dayofweek field values in WeAreUs dataset for all the users.

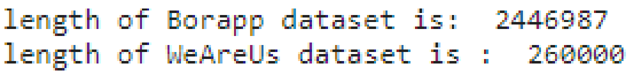
4.2.1.7 Replacing seenUnlocked from 0 to Off and 1 to Unlocked in WeAreUs dataset

The data column seenUnlocked is indicative of whether the screen of the user is locked or unlocked at the time of the record being taken on the user's mobile phone. The WeAreUs dataset has the values of screen unlocked as 0 for Off and 1 as unlocked, so we replaced these 0 and 1 with Off and Unlocked respectively, to make it similar to the BorApp dataset.

Data cleaning done on the Borapp dataset

4.2.1.8 Took 60 users data at random instead of full dataset having 342 users' data

Initially, I tried to use the full dataset provided by Martin that consisted of 342 users' mobile phone usage which amounted to around 30 million records in total. I used the free version of Kaggle for reading in the data, training the Machine Learning Models and predicting boredom in the WeAreUs Dataset. Due to the limited computing capacity provided by the free version of Kaggle, including a 16GB RAM, using the full dataset (around 2 GB) resulted in hitting the memory limit of Kaggle, while running the ML models.

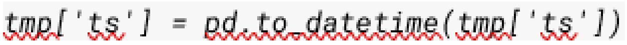


```
length of Borapp dataset is:  2446987
length of WeAreUs dataset is :  260000
```

To avoid this issue, I used a reduced dataset consisting of mobile phone usage data of 60 users, which was around 3 million records. Since the test dataset (WeAreUs) only contained 2,60,000 records, reducing the training data size did not have any significant impact on the research.

4.2.1.9 Converting time column from array to pandas datetime object for ease of use of the time date column

Using the function provided by pandas library, we converted the String datetime to Python datetime object as follows:

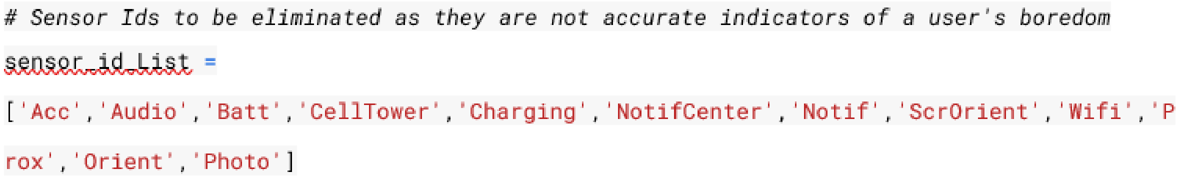


```
tmp['ts'] = pd.to_datetime(tmp['ts'])
```

4.2.1.10 Sensor Ids eliminated that are not accurate indicators of a user's boredom

After going through the sensors used to capture a user's mobile phone usage, we decided to eliminate the sensors that were not indicative of a user being bored in the given time frame.

The sensors data that was eliminated is as follows:



```
# Sensor Ids to be eliminated as they are not accurate indicators of a user's boredom
sensor_id_List =
['Acc','Audio','Batt','CellTower','Charging','NotifCenter','Notif','ScrOrient','Wifi','P
rox','Orient','Photo']
```

For example the Screen Orientation, Audio playing or not on the user's phone are not indicative of a user being bored at that moment, and thus these sensor values were not beneficial for our research and eliminated.

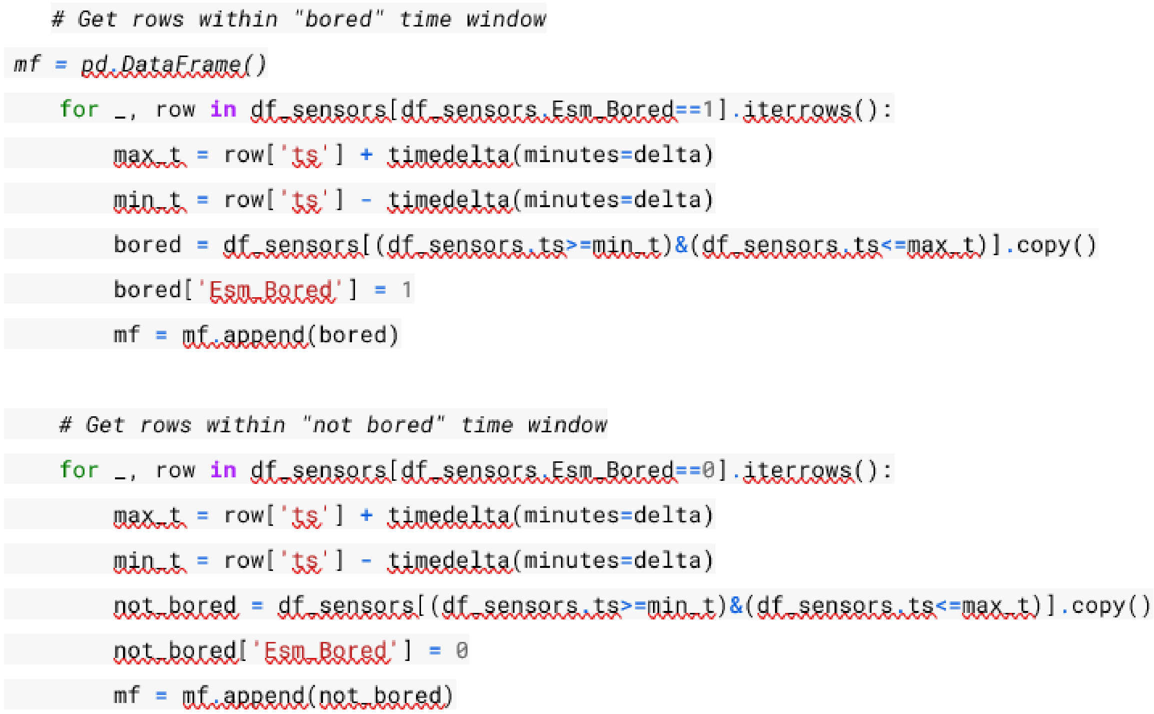
4.2.1.11 Based on the Esm_Bored column in Martin's dataset, we consider values 0 and 1

These are user inputted values on the mobile application that the user's installed as a part of Martin Pielot's research, where:

- 0: User inputted they are not bored in that instant
- 1: User inputted they are bored in that instant

We filter out these rows from Martin's dataset and then assume that the user would continue to be in the state of boredom for some time period in the vicinity of the time he manually admitted to being bored, so it can be safe to put in the same value for ESM_Bored for certain data points below and above that particular data point, thus replacing the non-null rows.

4.2.1.12 Replacing data points above and below in dataset with ESM_Bored values at that instance of time



```
    # Get rows within "bored" time window
mf = pd.DataFrame()
    for _, row in df_sensors[df_sensors.Esm_Bored==1].iterrows():
        max_t = row['ts'] + timedelta(minutes=delta)
        min_t = row['ts'] - timedelta(minutes=delta)
        bored = df_sensors[(df_sensors.ts>=min_t)&(df_sensors.ts<=max_t)].copy()
        bored['Esm_Bored'] = 1
        mf = mf.append(bored)

    # Get rows within "not bored" time window
    for _, row in df_sensors[df_sensors.Esm_Bored==0].iterrows():
        max_t = row['ts'] + timedelta(minutes=delta)
        min_t = row['ts'] - timedelta(minutes=delta)
        not_bored = df_sensors[(df_sensors.ts>=min_t)&(df_sensors.ts<=max_t)].copy()
        not_bored['Esm_Bored'] = 0
        mf = mf.append(not_bored)
```

Figure 4.5: Replacing datapoints below and above with value in ESMBored columns

As per the assumption mentioned in Section 4.2.2 1, we replaced data points in time windows of 5 minutes above and below the current data point with the ESM_Bored value at the current instant. The code used for this is as per the above image.

Assumptions

A user bored in a moment of time would continue to be bored for a period before and after that instance of time. We have taken this time window to be 5 minutes, as considering time windows lesser than 5 minutes gave us models with lower accuracy and increasing the time window to 10 minutes, 15 minutes lowered the accuracy of the model as the assumption was that a user would be bored for 10 minutes after he had recorded as being bored with the ESM (Experience Sampling Methodology). This was not the case as seen by the data retrieved from Martin's experiment.

4.3 Analysis of Prepared Data

In the following section we break down the performance based on different experiments performed by varying the features, number of features, number of users, and different Machine Learning algorithms.

4.3.1 Feature Selection - Choosing the features for building an accurate boredom classifier

Out of the 35 features listed by Martin Pielot as being accurate indicators of boredom, it was necessary to chose a subset of the input features that could be easily mapped to the WeAreUs dataset and at the same time keep the features that would help improve accuracy of the model and decrease the computational complexity. Choosing the correct features that would help in accurate prediction of the target variable, Boredom in our scenario, becomes important.

It was necessary to select features similar to the test dataset as in order to test the Machine learning model built by training the model on the BorApp dataset, we need a dataset that has the similar columns as well as the same number of values in the columns. The final list of features we chose were, Application type, Hour of the day, Day of the week, Activity context of the user, and whether the Mobile screen is unlocked or not.

Using Feature Importance provided by Random Forest Classifier, we were able to find out the top 5 features that would be accurate indicators of boredom on user.

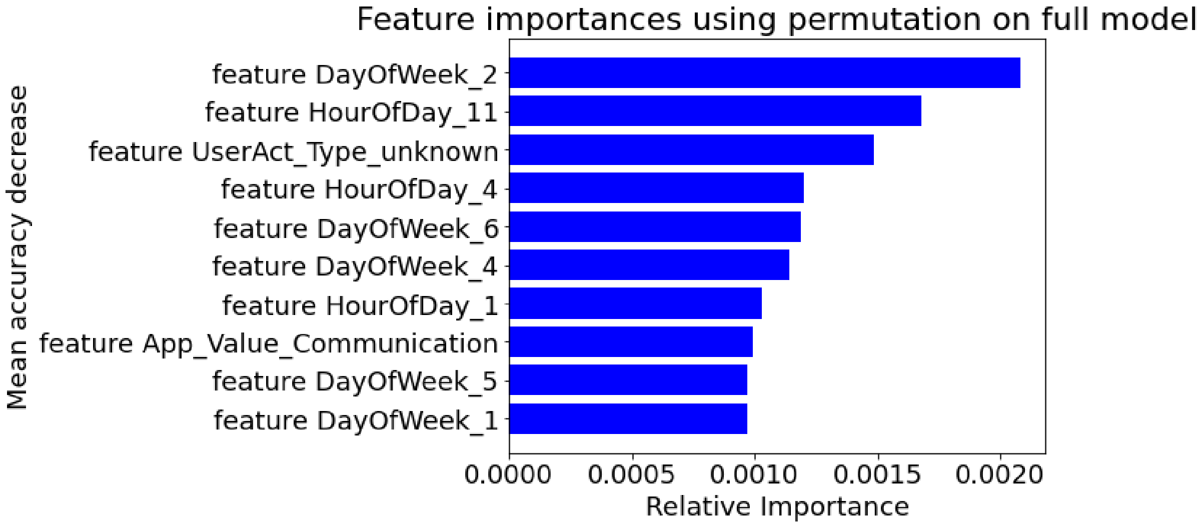


Figure 4.6: Feature Importance using Permutation on full model

Feature importance gives us Day of the week, Hour of Day, User Action type, Application value as the top features on which Boredom depends.

For the 60 users data, it can be seen that Tuesday and Saturday are the days when users are most bored and 11am and 4am are times of the day when users can be said to be in a bored state.

4.3.2 Choosing the optimum number of features for the boredom classifier model

One important thing to keep in mind when selecting the number of features to choose for training a Machine Learning model is that using the optimal number of features helps reduce the computational costs of the model thereby increasing the performance of the model. In 4.3.3 we discuss in detail about the different combinations of number of users and features that were considered to build a model giving high accuracy.

4.3.3 Choosing the number of users to consider for the boredom classifier model

The WeAreUs dataset was synthetically made on 13 users, whereas the BorApp dataset had 342 users as a part of the research experiment. Therefore, it was important to choose the correct number of users in the training dataset such that it did not overburden the computational resources and at the same time ensuring that model we got had high precision and accuracy, so that the boredom classifier to be used on the test dataset gave an accurate measure of boredom of users.

Following are some of the combinations used for finding the optimum combination of number of users and features to build a boredom classifier:

4.3.3.1 1st Iteration: One feature (App_Value) and One user

We got a very high F1- score and accuracy for the train-test split on Martin's data set(80:20) Looking at the Linear and non-linear classifiers used, Linear SVC model and Logistic Regression, both performed the same with equal F1-Scores and same accuracy as seen below.

Machine Learning Model	**F1-Score**	**Accuracy**
Linear SVC Model	*0.903*	0.933
Logistic Regression Model	*0.903*	0.933

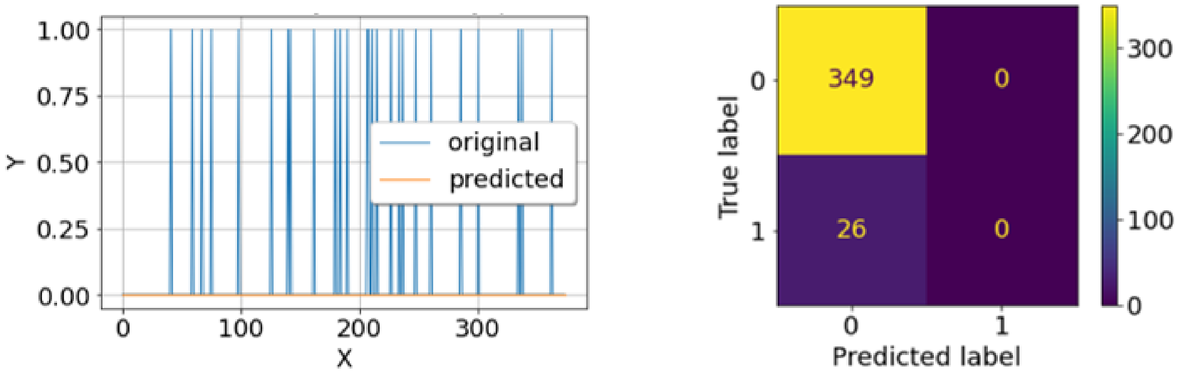


Figure 4.7: Plot of Predicted vs Actual values For Logistic Regression Model and Confusion Matrix

Looking at the Ensemble classifier used, Random Forest Classifier model, both F1-Scores and accuracy were higher than the other 2 models as seen below.

Machine Learning Model	F1-Score	Accuracy
Random Forest Classifier	0.933	0.976

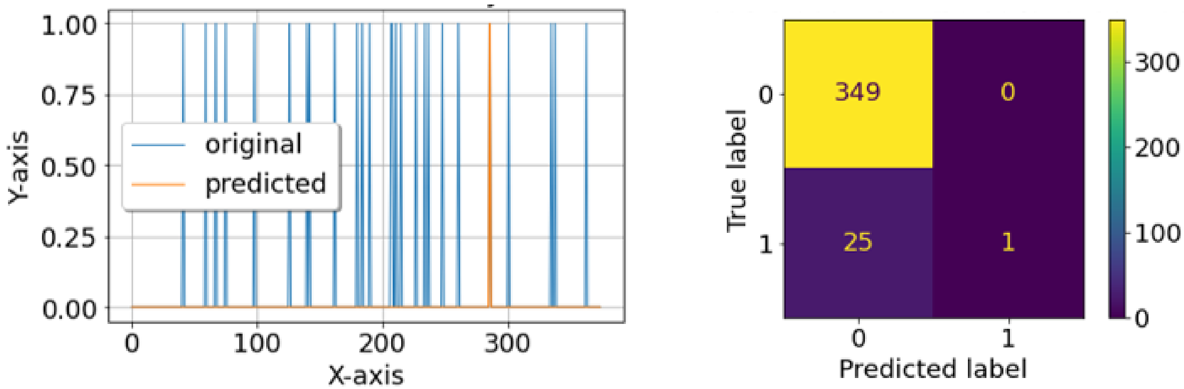


Figure 4.8: Plot of Predicted vs Actual values For Random Forest Classifier Model and Confusion Matrix

4.3.3.2 2nd Iteration: Two features (Application Value, Hour of Day) and One user

Increasing the features to 2, namely, App_Value and Hour_of_Day we get no significant difference in the performance for the machine learning models chosen. Looking at the Linear and non-linear classifiers used, Linear SVC model and Logistic Regression, both performed similarly with close F1-Scores and accuracy as seen below

Machine Learning Model	F1-Score	Accuracy
Linear SVC Model	*0.897*	0.*930*
Logistic Regression Model	*0.899*	0.*923*

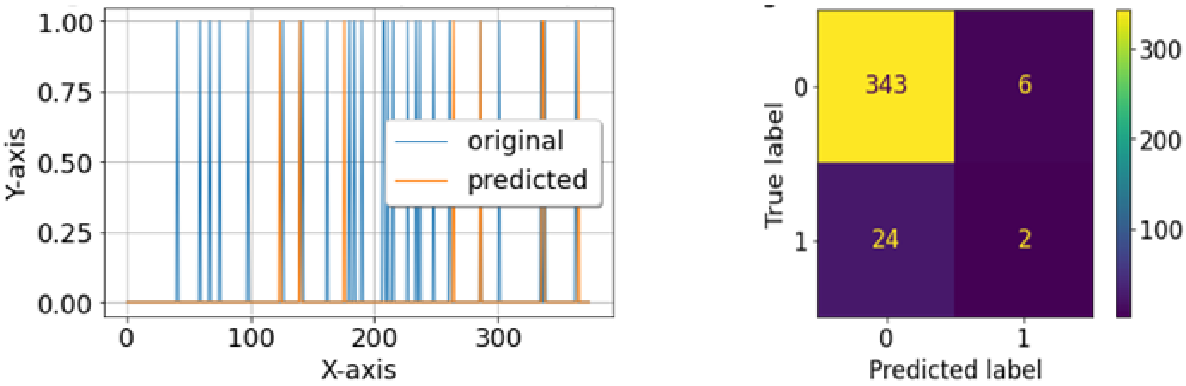
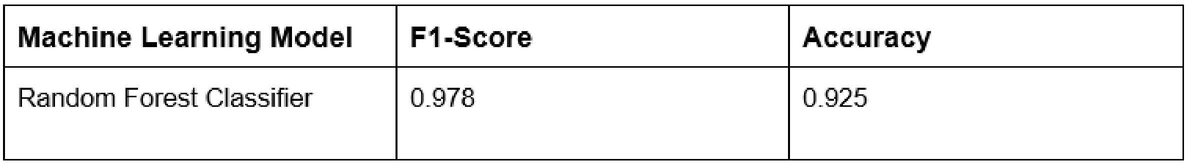


Figure 4.9: Plot of Predicted vs Actual values For Logistic Regression Model and Confusion Matrix - 2 Features 1 User

Looking at the Ensemble classifier used, Random Forest Classifier model, both F1-Scores and accuracy were higher than the other 2 models as seen below.



Machine Learning Model	F1-Score	Accuracy
Random Forest Classifier	0.978	0.925

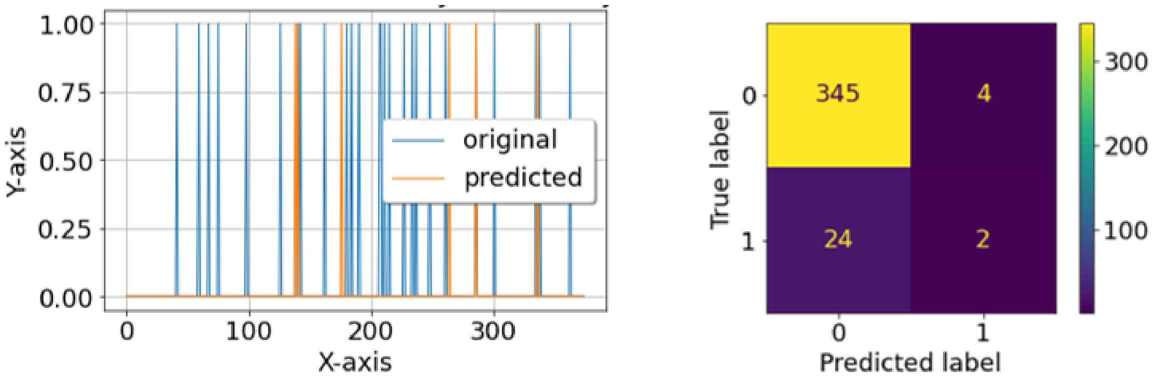


Figure 4.10: Plot of Predicted vs Actual values For Random Forest Classification Model and Confusion Matrix - 2 Features 1 User

4.3.3.3 3rd Iteration: 5 Features (Application Value, Hour Of Day, Day Of Week, User Action Type , Screen Locked or Unlocked) and 60 Users

Choosing all 5 features from the dataset, we ran the 3rd iteration using all 3 ML models. Looking at the Linear and non-linear classifiers used, Linear SVC model and Logistic Regression, both performed similarly with close F1-Scores and accuracy as seen below

Machine Learning Model	**F1-Score**	**Accuracy**
Linear SVC Model	*0.979*	0.976
Logistic Regression Model	*0.975*	0.986

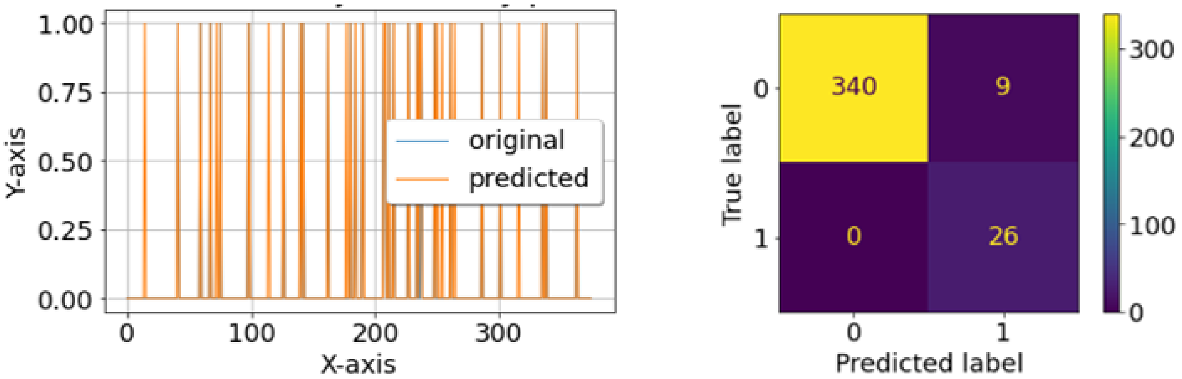
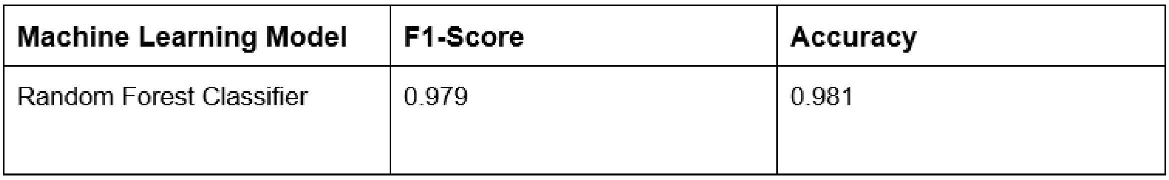


Figure 4.11: Plot of Predicted vs Actual values For Logistic Regression Model and Confusion Matrix - 5 Features 60 Users

Note: *Since we are using a small dataset with limited features, we cannot really consider the results received from the Linear SVC and Linear Regression model to be accurate as it best works for huge datasets.*

Looking at the Ensemble classifier used, Random Forest Classifier model, both F1-Scores and accuracy were higher than the other 2 models as seen below.



Machine Learning Model	F1-Score	Accuracy
Random Forest Classifier	0.979	0.981

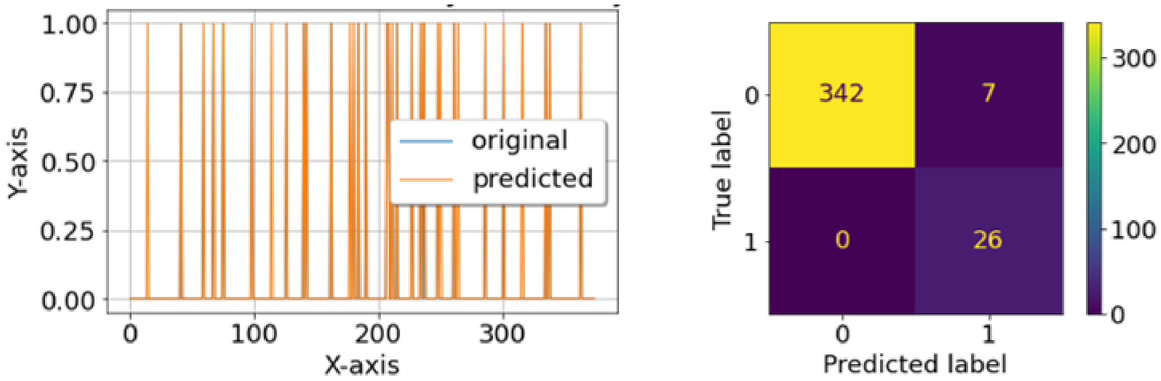


Figure 4.12: Plot of Predicted vs Actual values For Random Forest Classification Model and Confusion Matrix - 5 Features 60 Users

4.3.4 Choosing the optimum Machine Learning model for building the boredom classifier model

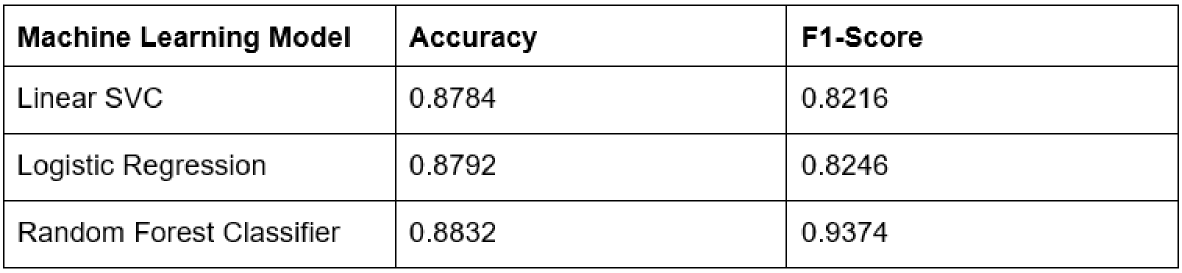
One of the Machine Learning models we have used is Random Forest Classifier, which performs feature selection automatically during training the model, as Random Forest selects optimum features intrinsically. RF only uses the features that help in maximising the accuracy of the model. From the initial tests on 1 User, Random Forest Classifier gave the best performance with highest Accuracy, F1 scores and best confusion matrix. RF performed best for all iterations of features and number of users. Thus for the prediction of the WeAreUs dataset, we used the Random Forest trained model on 5 features and 60 participants, which is discussed in the next section.

5 Results and Findings

In this chapter we will discuss about the accuracy of the boredom classifier built using the 3 different ML models, using the optimum number of features that were synchronous in the training dataset and test dataset, and the classification of boredom found in the test dataset, WeAreUs.

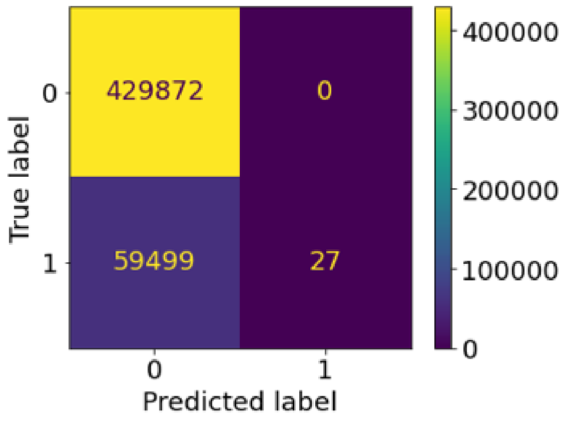
5.1 Best performing ML model

Finding opportune moments for delivery of notifications to the users based on their boredom is a binary classification problem, as the user can be in one of the 2 states: Bored or Not Bored. LinearSVC and Logistic Regression Models gave similar results for the boredom classification. Random Forest Classifier performed significantly better than the other two models.

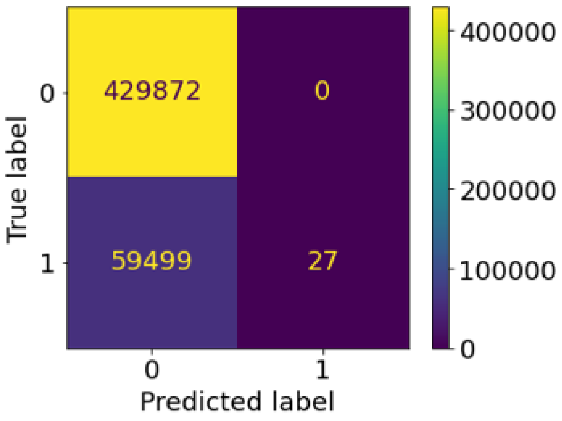


Machine Learning Model	**Accuracy**	**F1-Score**
Linear SVC	0.8784	0.8216
Logistic Regression	0.8792	0.8246
Random Forest Classifier	0.8832	0.9374

The Confusion matrices for all 3 Models are shown in the figure below.



Linear SVC - 5 Features Confusion Matrix



Logistic Regression - 5 Features Confusion Matrix

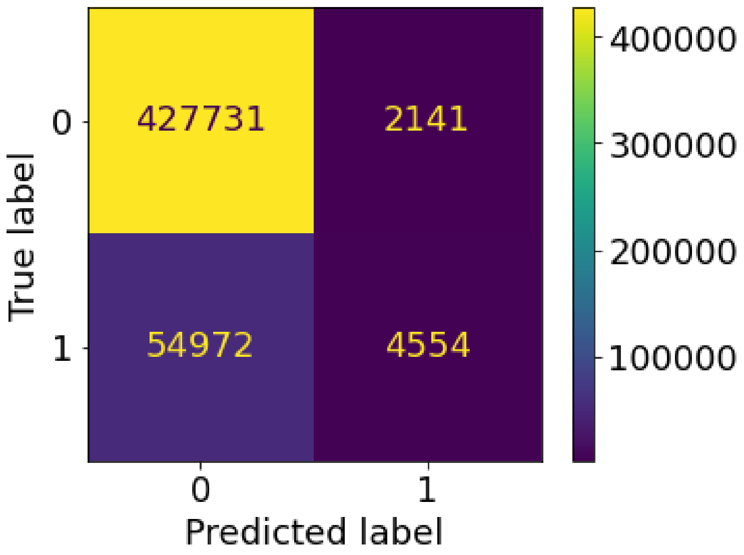
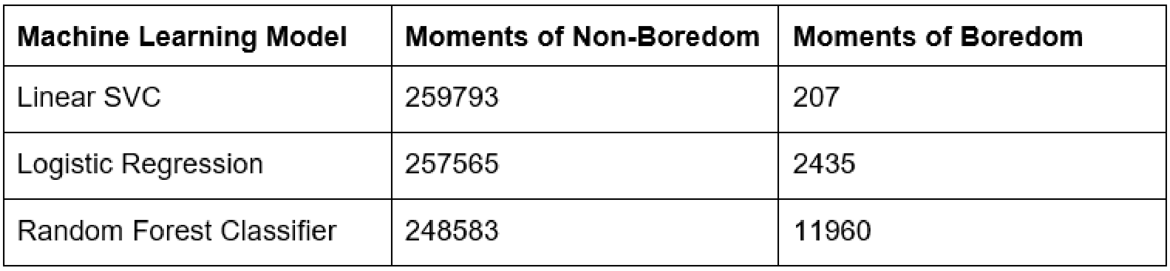


Figure 5.1: Random Forest Classifier - 5 Features Confusion Matrix

5.2 Categorising moments of boredom in WeAreUs dataset

After training the three ML models on different number of features and evaluating the model based on the test data obtained using train-test split on BorApp dataset, we use the test dataset, WeAreUs dataset, to predict moments of boredom.

The Figure 5.2 lists down the different ML models and the boredom classification (binary class classification) done on the WeAreUs dataset

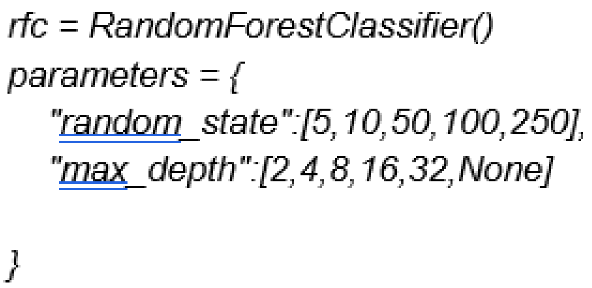


Machine Learning Model	Moments of Non-Boredom	Moments of Boredom
Linear SVC	259793	207
Logistic Regression	257565	2435
Random Forest Classifier	248583	11960

Figure 5.2: ML Model and Moments of Boredom count

As seen from Figure 5.2, the best performing ML model is Random Forest Classifier identifying 248583 moments of non-boredom and 11960 moments of boredom.

Below is the range of random_state and max_depth used for finding the best parameters for Random Forest Classification Model using GridSearchCV().



```
rfc = RandomForestClassifier()
parameters = {
    "random_state":[5,10,50,100,250],
    "max_depth":[2,4,8,16,32,None]

}
```

5.3 Visualising moments of boredom in WeAreUs dataset

We visualised each of the 5 features used for this research, based on the moments of boredom in the WeAreUs dataset and found the following outcome :

5.3.1 Application Type

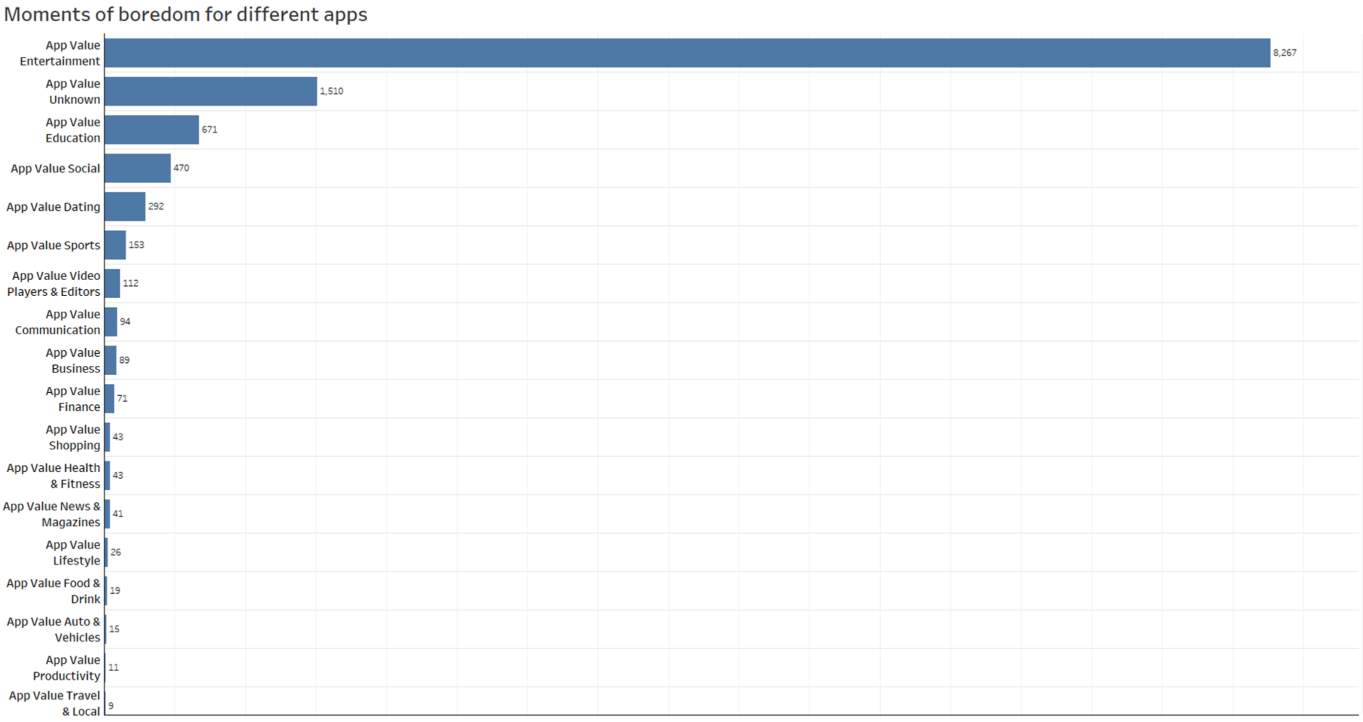


Figure 5.3: Moments of Boredom for Different App types

Varying the application types on the 11,970 moments of boredom found using the RF model on the WeAreUs dataset we got the bar chart in the figure below. As is evident from the bar chart, Apps having values as "Entertainment", "Unknown", "Education", "Social" and "Dating" had the highest levels of boredom, meaning that when the users where using these types of apps they can be deemed to be bored at that moment and looking for stimulation of some kind. So, it can be concluded that when users are using apps in these categories, they would be more likely to interact with any notifications they receive and be more receptive to the interruptions.

The second highest app type is "Unknown", this is because of the large number of missing apps in the PlayStore for Martin's dataset, due to which they could not be categorized into app types and subsequently the trained model used on the WeAreUs dataset, had many unknown app types.

5.3.2 Screen Locked/Unlocked

Based on if the mobile screen was unlocked and on or whether it was off, we found the following bar chart depicting the boredom levels. From the bar charts it can be concluded that when the Screen is Unlocked and On, there are 6491 moments of boredom recorded, which is plausible as the user would be using their phone when they are bored, actively looking for some kind of entertainment to get out of their bored state.

Screen value Off also had a significant number of bored instances, which goes on to show that Screen Unlocked/Locked is not quite an accurate indicator of boredom in a user. As ideally, the phone being unlocked and on means the user is bored, as backed up by Martin Pielot's research (4).

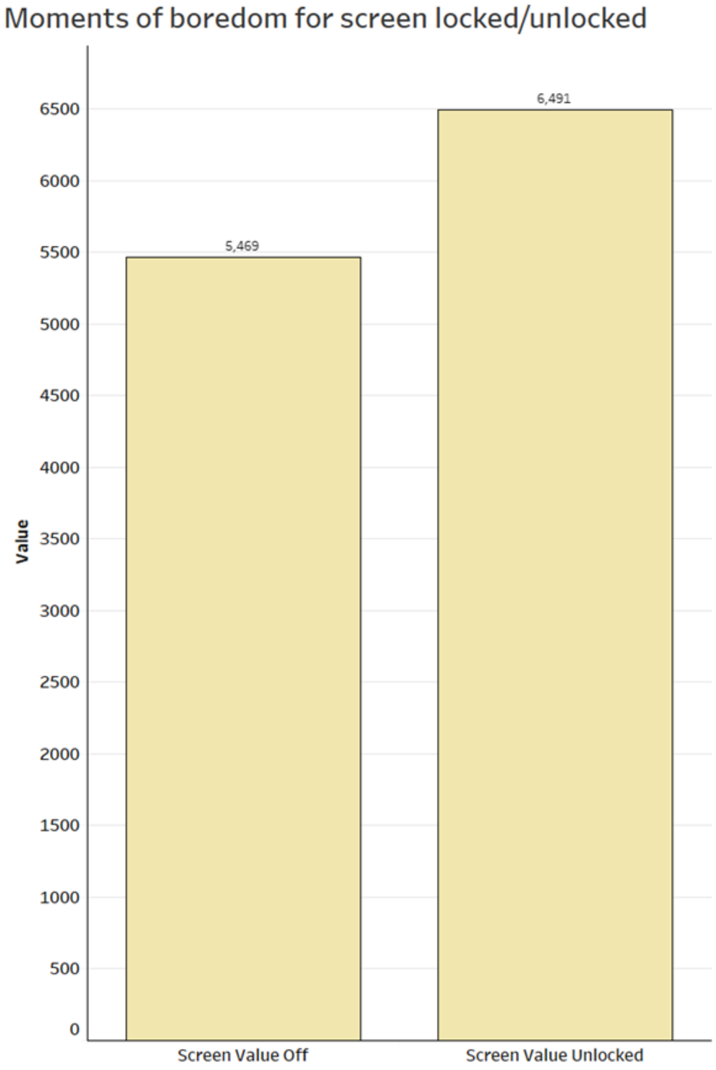


Figure 5.4: Moments of Boredom for Screen Locked/Unlocked

5.3.3 Day of the week

According to Martin Pielot's research, people are more likely to be bored on the weekdays when they have days off at their workplace, and this was seen in the WeAreUs dataset with users being more bored on Saturday, Sunday and Friday and being least bored on the weekday. The bar chart below shows the same. Day of the week can be said to be an accurate indicator of boredom levels in a user.

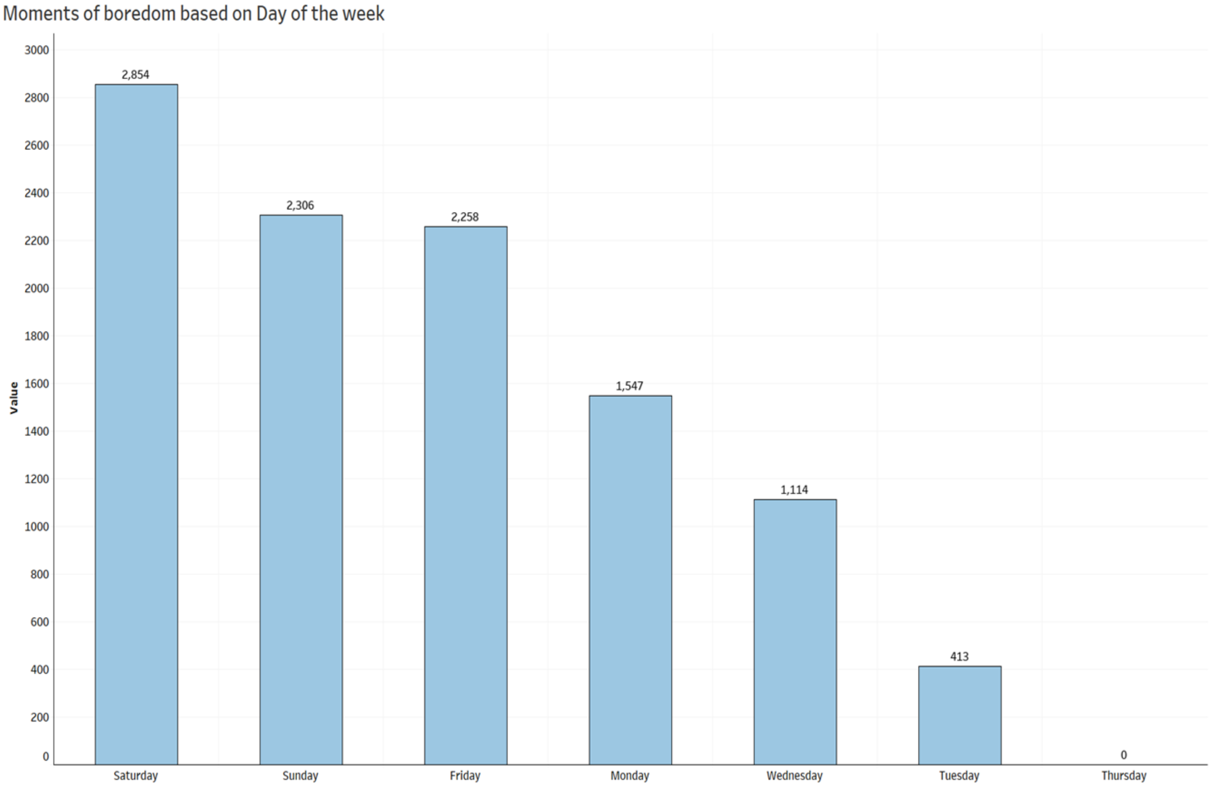


Figure 5.5: Moments of Boredom based on Day of the Week

5.3.4 Hour of the day

The line chart below shows that there is a decrease in the trend of users being bored based on the time of the day they are using their phones. When the users are off work, at night after 10pm and early mornings, it can be seen that the highest moments of boredom are recorded in the users in the WeAreUs dataset. This supports the findings of Martin's research, where they found that when the daylight hours are over, or during the afternoon when people are on a break, highest instances of boredom were recorded in users.

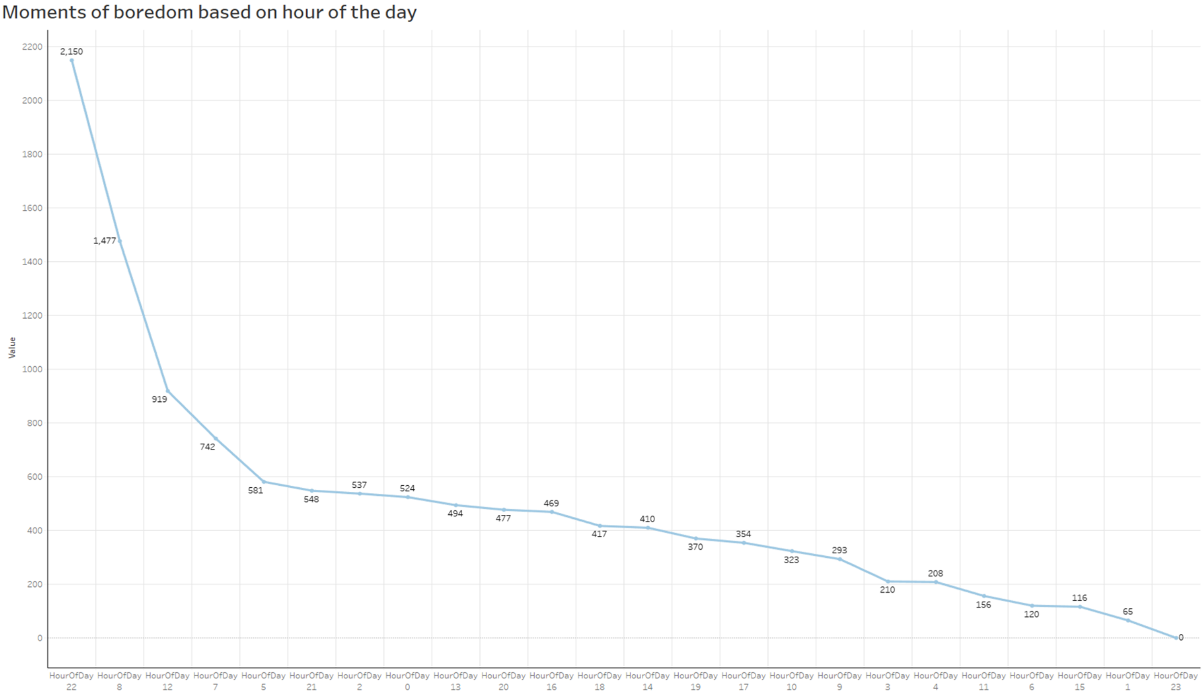


Figure 5.6: Moments of Boredom based on Hour of the Day

6 Conclusion

6.1 Overview

This study helped build a boredom classifier using a dataset that collected mobile phone usage patterns in 342 users, after installation of an app, Borapp, from the Play Store. The data collected as a part of this experiment carried out by Martin Pielot and his team followed all ethical guidelines. The second dataset used for the evaluation of this research, is a synthetically generated dataset, WeAreUs, that allowed us to use a larger variety of data.

Additionally, three machine learning models were used for predicting moments of boredom in a mobile phone user based on his mobile phone usage patterns. The best performing model, Random Forest Classifier, was able to predict 45% moments of boredom in the test dataset through a binary classification model. It was not possible to achieve exact moments of boredom due to a limitation in the features in the test dataset, and a substantial number of missing values in the training dataset.

6.2 Primary Findings

- It was possible to build a boredom classifier based on a user's mobile phone usage patterns from one dataset and use it to classify moments of boredom on users in a different dataset
- Out of the three Machine Learning Models used, Random Forest Classifier performed the best with an F1 score of 0.93 and Accuracy of 0.88.
- Out of the 5 features considered for building the ML models, Application Type gave the best predictions of boredom on the WeAreUs dataset with 11960 instances of boredom detected.

6.3 Key Learnings

- It was possible to take one dataset, Martin's dataset, which had a boredom classifier implemented in the data through Experience Sampling Methods, and do important data transformations on this data to then build a machine learning model with high performance, low computational costs and considerable accuracy.
- To then take this trained model with Martin's data, and use it to predict boredom on a different dataset having similar features as Martin's dataset, we were able to build a boredom classifier, which could be validated by adding the Bored column received through the predictions in the WeAreUs dataset, and using it to visualise the boredom classification model on each of the 5 features.

7 Limitations

- Due to a significant amount of missing data and unknown values in Martin Pielot's dataset, the model built was not as accurate as we had imagined it to be originally. The experiment conducted by Martin Pielot required users to install the Borapp on their devices and use it for a certain amount of time. The users also got notifications for an ESM(Experience-Sampling Methodology) which had a questionnaire asking users if they were feeling bored at that particular instance of time. A possibility is that users could have answered falsely to the questions of them being bored, which could lead to an inaccurate model being built and thus the results received after applying the boredom classifier on the test dataset could be biased.
- Since the experiment conducted by Martin Pielot and his team took place in 2015/16, one of the features for which data was collected from the users, application package, had applications that were either missing in PlayStore or their information was unavailable as they had been removed. This made identifying the category of the app from PlayStore difficult due to which these applications were classified as "Unknown".
- The research by Martin Pielot and his team does not take into account the diversity in the dataset, such as time zones of the participants, or different work shifts meaning different people would be deemed busy at different times of the day. It does not consider the participants having any underlying or medically diagnosed psychological conditions, making them use their phone more compulsively than other users.
- The research by Kieran Fraser, did not capture the gender and Age of the participants, which could have been used as features in training the model, as these two features had some importance in Martin's boredom classification, as is evident from the Feature Importance by Mean Impurity Decrease score figure in Section 4.1.

8 Future Work

- **Improving the Machine Learning Model by adding more features.**

  The Borapp dataset by Martin Pielot has many features and out of the 35 features they found to be most effective indicators of boredom in users, we were only able to use 5 features as the rest were not present in the test dataset, WeAreUs. In Machine Learning, there is a certain number of features that are thought to be the apt number to achieve a high performing model. Since our model was only able to classify 11,960 moments of boredom in the test dataset, which was roughly 44% of the dataset, with more number of features added during training of the model there could be a chance of achieving higher accuracy, F1-score and a high performing model that can then be used for boredom classification.

- **Using more datasets to test the boredom classifier.**

  For our research, we only considered 1 test dataset, the WeAreUs dataset which had Synthetically generated data with some features that mapped to the training dataset. In the future, it would be beneficial to look for more datasets that could be used to test the trained model for boredom classification. As stated in Chapter 1.2 Research challenges, it is difficult to find datasets collected "in-the-wild" that contain mobile phone usage patterns of real users, thus if possible synthetic datasets can be used instead.

- **Utilising moments of boredom to engage users with custom content through notifications.**

  One of the most useful outcomes of finding moments of boredom in mobile phone users is that these moments of boredom can be used to send tailored notifications to the users, to increase their productivity, to remind them of important tasks, to encourage them to utilise their time by doing exercise or meditation.

- **Transferability of Learnings**

  Martin's dataset was collected in 2014, it contains mobile phone usage patterns including types of apps used by users and which apps were most interacted with through their notifications. However, there is a question of the validity of this data in today's age, 2022 and whether this data collected in 2014 holds valid in terms of the usage patterns, as after the COVID pandemic, many people started working/studying virtually. Thus, it would be interesting to see how the usage patterns have changed, and if the newly derived mobile phone usage patterns can be used to build a boredom classifier using Martin's research as a basis, to have a more accurate and uptodate Boredom Classifier.

Bibliography

[1] Deloitte. Irish people check their phones on average 50 times a day, 2019. URL `https://www2.deloitte.com/ie/en/pages/technology-media-and-telecommunications/articles/mobile-consumer-stats.html`.

[2] Microsoft. Notification, Disruption, and Memory: Effects of Messaging Interruptions on Memory and Performance, 2001. URL `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/Interact2001Messaging.pdf`.

[3] R Kelly Garrett and James N Danziger. Im= interruption management? instant messaging and disruption in the workplace. *Journal of Computer-Mediated Communication*, 13(1):23–42, 2007.

[4] Martin Pielot, Tilman Dingler, Jose San Pedro, and Nuria Oliver. When attention is not scarce-detecting boredom from mobile phone usage. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 825–836, 2015.

[5] Mauro Cherubini and Nuria Oliver. A refined experience sampling method to capture mobile user experience. *arXiv preprint arXiv:0906.4125*, 2009.

[6] Karen Church, Mauro Cherubini, and Nuria Oliver. A large-scale study of daily information needs captured in situ. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(2):1–46, 2014.

[7] Trinh Minh Tri Do, Jan Blom, and Daniel Gatica-Perez. Smartphone usage in the wild: a large-scale analysis of applications and context. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 353–360, 2011.

[8] Joel E Fischer. Experience-sampling tools: a critical review. *Mobile living labs 09: Methods and tools for evaluation in the wild*, page 35, 2009.

[9] Barry Brown, Moira McGregor, and Donald McMillan. 100 days of iphone use: understanding the details of mobile device use. In *Proceedings of the 16th international*

conference on Human-computer interaction with mobile devices & services, pages 223–232, 2014.

[10] Scott Carter, Jennifer Mankoff, and Jeffrey Heer. Momento: support for situated ubicomp experimentation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 125–134, 2007.

[11] Jon Froehlich, Mike Y Chen, Sunny Consolvo, Beverly Harrison, and James A Landay. Myexperience: a system for in situ tracing and capturing of user feedback on mobile phones. In *Proceedings of the 5th international conference on Mobile systems, applications and services*, pages 57–70, 2007.

[12] Shamsi T Iqbal and Brian P Bailey. Oasis: A framework for linking notification delivery to the perceptual structure of goal-directed tasks. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 17(4):1–28, 2010.

[13] Joel E Fischer, Chris Greenhalgh, and Steve Benford. Investigating episodes of mobile phone activity as indicators of opportune moments to deliver notifications. In *Proceedings of the 13th international conference on human computer interaction with mobile devices and services*, pages 181–190, 2011.

[14] Abhinav Mehrotra, Veljko Pejovic, Jo Vermeulen, Robert Hendley, and Mirco Musolesi. My phone and me: understanding people's receptivity to mobile notifications. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 1021–1032, 2016.

[15] Otto Fenichel. On the psychology of boredom. 1951.

[16] Stephen J Vodanovich. On the possible benefits of boredom: a neglected area in personality research. *Psychology and Education: An Interdisciplinary Journal*, 2003.

[17] John D Eastwood, Alexandra Frischen, Mark J Fenske, and Daniel Smilek. The unengaged mind: Defining boredom in terms of attention. *Perspectives on Psychological Science*, 7(5):482–495, 2012.

[18] Thomas Goetz, Anne C Frenzel, Nathan C Hall, Ulrike E Nett, Reinhard Pekrun, and Anastasiya A Lipnevich. Types of boredom: An experience sampling approach. *Motivation and emotion*, 38(3):401–419, 2014.

[19] David A Bray. Conceptualizing information systems and cognitive sustainability in 21st century'attention'economies (includes syllabus). *Piedmont Project*, 2007.

[20] Robert Bixler and Sidney D'Mello. Detecting boredom and engagement during writing with keystroke analysis, task appraisals, and stable traits. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, pages 225–234, 2013.

[21] Qi Guo, Eugene Agichtein, Charles LA Clarke, and Azin Ashkan. In the mood to click? towards inferring receptiveness to search advertising. In *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 319–324. IEEE, 2009.

[22] Gloria Mark, Shamsi T Iqbal, Mary Czerwinski, and Paul Johns. Bored mondays and focused afternoons: the rhythm of attention and online activity in the workplace. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3025–3034, 2014.

[23] Andrey Bogomolov, Bruno Lepri, Michela Ferron, Fabio Pianesi, and Alex Pentland. Daily stress recognition from mobile phone data, weather conditions and individual traits. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 477–486, 2014.

[24] Andrey Bogomolov, Bruno Lepri, and Fabio Pianesi. Happiness recognition from mobile phone data. In *2013 international conference on social computing*, pages 790–795. IEEE, 2013.

[25] Robert LiKamWa, Yunxin Liu, Nicholas D Lane, and Lin Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 389–402, 2013.

[26] James Fogarty, Scott E Hudson, Christopher G Atkeson, Daniel Avrahami, Jodi Forlizzi, Sara Kiesler, Johnny C Lee, and Jie Yang. Predicting human interruptibility with sensors. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(1):119–146, 2005.

[27] Daniel Avrahami and Scott E Hudson. Responsiveness in instant messaging: predictive models supporting inter-personal communication. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 731–740, 2006.

[28] Martin Pielot, Rodrigo De Oliveira, Haewoon Kwak, and Nuria Oliver. Didn't you see my message? predicting attentiveness to mobile instant messages. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 3319–3328, 2014.

[29] Stephanie Rosenthal, Anind K Dey, and Manuela Veloso. Using decision-theoretic experience sampling to build personalized mobile phone interruption models. In *International conference on pervasive computing*, pages 170–187. Springer, 2011.

[30] Eric Horvitz, Paul Koch, Raman Sarin, Johnson Apacible, and Muru Subramani. Bayesphone: Precomputation of context-sensitive policies for inquiry and action in

mobile devices. In *International Conference on User Modeling*, pages 251–260. Springer, 2005.

[31] Martin Pielot. Large-scale evaluation of call-availability prediction. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 933–937, 2014.

[32] Rowan Sutton. Evaluating synthetic notification trained reinforcement learning for mobile notification management systems.

[33] IBM. Logistic Regression, 2020. URL `https://www.ibm.com/topics/logistic-regression`.

[34] IBM. Random Forest Classifier, 2020. URL `https://www.ibm.com/cloud/learn/random-forest`.

[35] IBM. SVC, 2021. URL `https://www.ibm.com/docs/en/spss-modeler/saas?topic=models-how-svm-works`.

A1 Appendix

A1.1 Application Category Table

Some examples of the categorization done for app packages from Play Store.

app package	app type
com.alibaba.aliexpresshd	Shopping
mmapps.mirror.free	Beauty
com.Worktoday	Business
com.tennistemple	Sports
com.ea.game.nfs14_row	Games
com.google.android.projection.gearhead	Auto And Vehicles
mmapps.mirror.free	Beauty
wp.wattpad	Books And Reference
com.okcupid.okcupid	Dating
com.snapchat.android	Communication
com.nike.plusgps	Health And Fitness
com.google.android.play.games	Entertainment