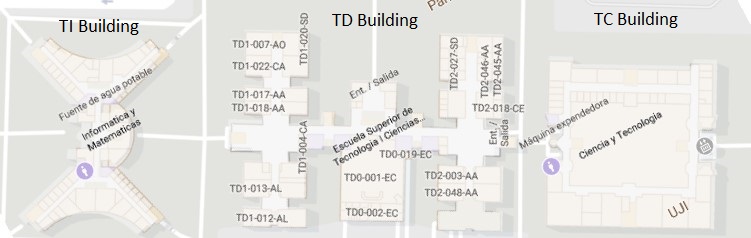
Introduction

Modern technology has made amazing strides in the field of object and personnel tracking for business and government use by means of GPS (Global Positioning System) receivable technology (smartphones, laptops, location assistance trackers/mappers). The technology operates on the fundamental process of triangulating an object’s location by using multiple orbiting satellites to accurately calculate the target’s location. And while GPS technology is surprisingly effective in providing location and local time information to its users it also suffers from some drawbacks - namely the inability to operate inside of buildings because of the signals from satellites being scattered or blocked by roofs and walls.

To remedy the lack of good quality indoor tracking new technologies in the form of IPS (Indoor Positioning System) are currently being developed. Unlike GPS, which is an outer space centric system operated by the United States government that has been fully developed and standardized, IPS is a burgeoning technology with no set standards for establishing an object or person’s location. At this time the majority of indoor positioning techniques involve using established wireless technologies, such as Bluetooth and Wi-Fi, to coordinate where something or someone is by using the respective signals produced by wireless devices. IPS is not limited to using Wi-Fi and Bluetooth signal producing devices, but a good amount of research and acceptance by organizations us them as their standard for indoor location tracking because of their already established use in businesses and homes and cost of implementation.

Kathy Narim-Nomstrom of IOT Analytics has requested that my team and I review and recommend a solution for the data they have collected to create a reliable IPS for the TC, TD, and TI buildings of Computer Systems and Computing Political University of Valence, Spain (see Figure 1)[[1]](#footnote-1). As a professional data analysis team we will follow a thorough process to analyze the data provided, correct or workaround any issues that may be found with said data, test algorithm predictive models, and build final productive models from the data. The outcome from this work will be a functional machine learning model, research documentation, and a comprehensive solution model for the client.

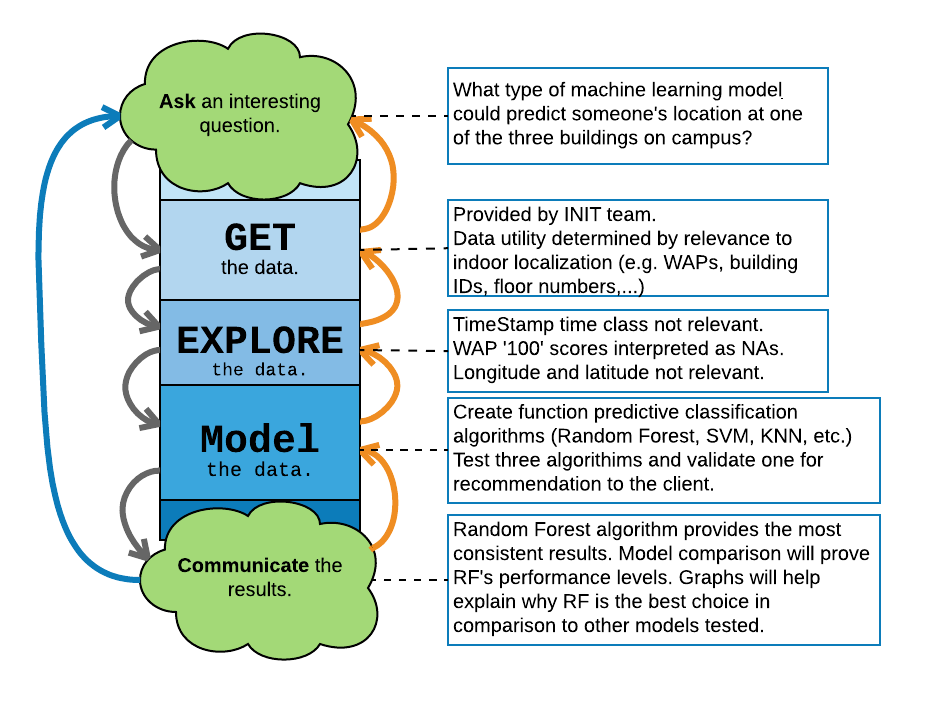


**Figure 1**.

**Note**: The initials were used for each building for the sake of clarity.

Data Science Process

After an initial search through the data and some prototype training and testing sets were created it became abundantly clear that it was time to implement a data science process to the project. Sometimes the starting of a data science process can begin at the start of a project, but I have found that there are cases where it is better to have an understanding of the data first before making a decision regarding a project’s strategy. The data science process that we decided to use for this project is based on the work of Blitzstein and Pfister for a Harvard data science class.[[2]](#footnote-2) Their process was chosen because it provided a straightforward guided approach to the work that we were doing without unnecessary complexity, which was very important to us early on in the project as there were many aspects of the data to consider and requests made for what the prediction algorithm could accomplish. To better understand the steps that have been taken to address questions and requirements for this task I have included a diagram detailing the overall topics that were covered (see Figure 2).



**Figure 2**.

Data Management

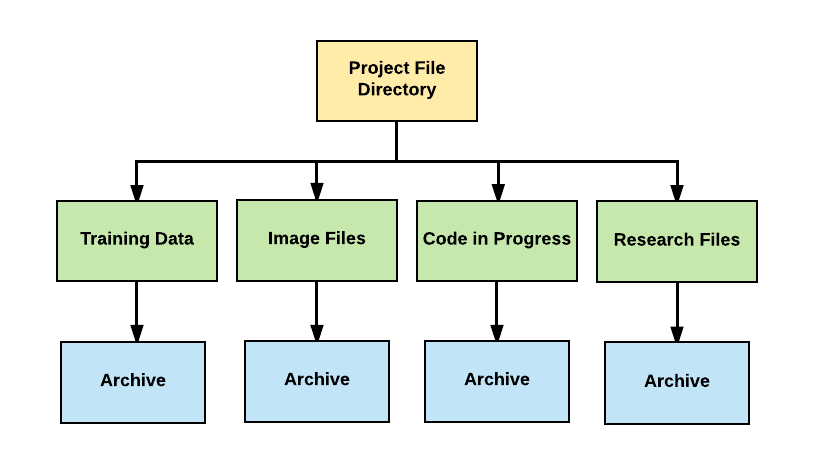
Data for this project is stored in a locally organized network of folders - each focused on a particular aspect of the assignment along with archive subfolders for retired files (see Figure 3). Backups of the folder are routinely made biweekly for preservation.

**Training Data** – Folder cover all training data set iterations.

**Image Files** – All image files (JPEG, BMP, PNG) pertaining to the project (screenshots, graphs, etc.).

**Code in Progress** – Repository of all current source code for the project with all previous versions stored in respective archive folder.

**Research** – All files containing notes, copies of files provided by the client, website bookmark backup, and research papers acquired throughout the project.

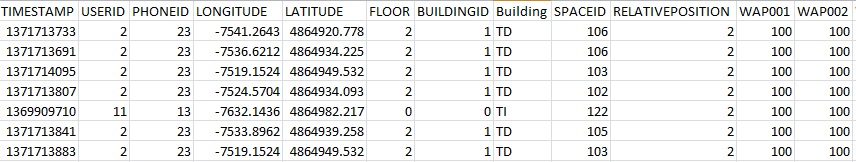


**Figure 3**. Defined layout of data structure for project.

Data Related Issues

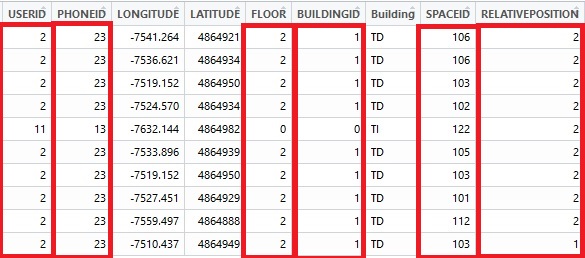
With every data set there are inevitably going to be some quirks in the data. Normally small things that cause problems that can be easily fixed, such as strange data class formatting or odd column headers. Other times it can be more severe problems, like missing values that may require serious consideration about their effect on the data set. Now with that in mind I will provide a detailed list of issues that were found in the data and what steps were taken to resolve them for this project.

**No unique data identifiers** – Normally a dataset for users would have a system for recording the specific results for them (see Figure 4). This may come in the form of a special identification code or usernames.

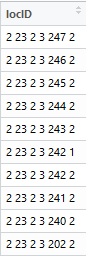


**Figure 4**. No distinctive naming or IDs used.

To ensure that each record may be easily identified a system was created to piece together each row’s data. The process involved joining each row’s UserID, PhoneID, Floor number, BuildingID, Space ID, and Relative Position (see Figure 5) into a new column called locID short for Location ID (see Figure 6) thus making each record more unique than what the original dataset allowed.

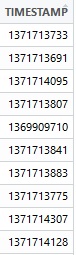


**Figure 5**.



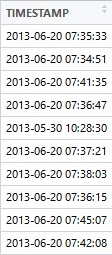
**Figure 6**.

**Time data not correctly formatted** – We found that the date time data was not correctly recorded for our purposes in that it did not show the actually day and time (see Figure 7).



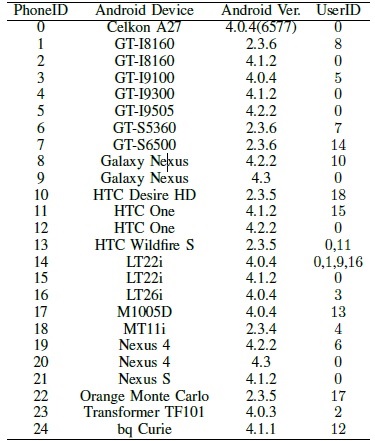
**Figure 7**. Time scale stored in nondescript data class of seconds.

This problem was fixed by changing the data type from an integer string to a date and time based one (see Figure 8).



**Figure 8**. Time scale changed to match ISO 8601 time standard.

**Unnecessary data** – After reviewing the data it was found that there were attributes that either had little or no value to the data analysis process. The longitude and latitude entered for each record was at first interesting to note for comparison of respective locations until it is considered that the data is most likely wrong because it was recorded inside buildings using GPS, which has been shown to be unreliable, thus requiring us to dismiss the data. This same issue was found with the PhoneID and UserID attributes in that they fail to provide the added detail to the data set that they are intended for in this project. Looking at Figure 9 we can see that the PhoneID and UserID are not related one-to-one. Instead users have switched phones on multiple occasions thus the data has been made utterly useless for analysis of which phones were the best at picking up WAP signals. Someone could argue that not all the UserIDs show as using multiple phones, but I think that the data has been tainted enough and knowing peoples’ behavior there may be more phones that were switched between people than we know.



**Figure 9**.

Prediction Model Comparison

Once the analysis and correction of the data that was provided to us was completed we moved on creating classification models to allow for prediction of the users’ location at each building. As was explained before the unique qualities of the data lead us to design the models to only interpret specific buildings and not all three within the same algorithm. It should be noted that to prevent confusion the Kappa scores for each model will not be reported, but may be found in the additional documentation provided with this report (See Model Resamples Summary).

**Random Forest**

Random forests or random decision forests is an ensemble learning method that works by creating a series of decision trees to analyze the data presented to it and returns a classification based on the mode (most often returned result). With the Random Forest accuracy numbers I found that the numbers for the three buildings stayed within a compact range of 70 - 85% (see Figure 10), which is the tightest and closest matching building range of the three models produced for this project. The accuracy numbers themselves are fair, but definitely not considered great results for these tests (85 – 95% at the highest would be ideal).

**Figure 10**. Random Forest accuracy plot.

**SVM**

To understand SVM it is probably best to get an general description of what it is and what it can do. I think Greg Lamp described it best:

“SVM is a supervised machine learning algorithm which can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.”

In other words the algorithm takes the data it is given, transforms it, and plots the data to determine the boundary between what is of a particular class of things and what is not. In our case what building, floor, and area someone or something is occupying.

The SVM model that was produced from the three buildings training data had the greatest variance of the three models produced. This can be seen by reviewing the results as we can see that the minimum and maximum accuracy scores for the TI and TD buildings have a difference of roughly 23% points (see Figure 11). It is very hard to recommend using the SVM model for predictions as the results vary so drastically with the TI building prediction accuracy being great, TC being fair, and TD being poor.

**Figure 11**. SVM accuracy plot.

**KNN**

K nearest neighbor is a simple algorithim that takes all existing data and arranges it into groups based on proximity to one another. Once this task is completed by the KNN model it applies this grouping scheme to all new data presented to it for categorization. The KNN model results are easily the most disappointing with the best results for the three buildings, TD, having a maximum prediction number of less than 61% (see Figure 12). The lower end of values is taken up by the TI building’s KNN results with the maximum prediction accuracy value of less than 41%, which means that the entire series of models for the buildings is unreliable more than half the time at its best.

**Figure 12**. KNN accuracy plot.

So far we have looked at the accuracy data from the machine learning model side, but I would like to now present the data from a different angle so that we may gain some different insight into how they perform from each of the buildings’ perspectives. And unlike before when I used charts to show performance levels we will let numbers show the highs and lows of our ML models.

For each of the buildings a chart was created, displaying the stratification of algorithm accuracy performance across all three models. We will again be using accuracy for discussion purposes instead of Kappa as the scores by and large match so closely. It should be noted that color coding was added to apply a value range every possible answer. Red is for accuracy that goes from less than 70%, yellow for greater than 70% to less than 80%, and green for 80% or higher achievement.

We can see from the chart that Random Forest and SVM worked fairly well for the TC building in predicting locations with close matching scores. While KNN failed to meet expectations with numbers in the 50% range (see Figure 13).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC Accuracy | Min | 1st Qu | Median | Mean | 3rd Qu | Max |
| Random Forest | 71.92% | 73.76% | 75.11% | 74.99% | 75.95% | 79.16% |
| SVM | 71.20% | 73.94% | 75.39% | 75.53% | 77.01% | 80.43% |
| KNN | 51.21% | 53.27% | 54.10% | 54.29% | 55.19% | 58.41% |

**Figure 13**. TC Building accuracy graph.

With the results for the TD building it is interesting to note that the Random Forest prediction results are the only consistently fair to good numbers (see Figure 14). The SVM model predictions fail surprisingly when compared to its performance with other buildings. And based on researching the dataset it is not caused by simple issues, such as sample size which was found to be nearly the same amount as the TI building’s training set. I am led to believe there may be a problem with the building itself.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TD Accuracy | Min | 1st Qu | Median | Mean | 3rd Qu | Max |
| Random Forest | 75.73% | 78.47% | 79.92% | 79.78% | 81.25% | 84.68% |
| SVM | 61.32% | 65.60% | 67.17% | 66.81% | 68.25% | 71.16% |
| KNN | 51.80% | 56.08% | 57.65% | 57.29% | 58.73% | 61.65% |

**Figure 14**. TD building accuracy graph.

Unlike before with the TD building’s metric data the TI building’s values show a reversal of fortune between Random Forest and SVM (see Figure 15). This time SVM has very good minimum through maximum numbers, while Random Forest has more tepid marks with averages between 70% and 78.43%. The KNN model continues to fail to perform with prediction numbers in the red between 32% and 40.61%.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TI Accuracy | Min | 1st Qu | Median | Mean | 3rd Qu | Max |
| Random Forest | 70.31% | 73.48% | 74.27% | 74.18% | 75.36% | 78.43% |
| SVM | 85.58% | 88.64% | 90.10% | 89.85% | 91.06% | 94.12% |
| KNN | 32.07% | 35.13% | 36.59% | 36.34% | 37.55% | 40.61% |

**Figure 15**. TI building accuracy graph.

Model Recommendation

After much consideration I believe the best option of the three predictive algorithms that we tested for this project is Random Forest. Initially I would have preferred to have recommended SVM, but because of its significant issues predicting the TD building locations accurately it makes suggesting the algorithm impossible. While I did consider other factors such as speed and ease of use of the various models, which is important from a development standpoint. Ultimately it came down to fundamental prediction ability and SVM has one too many glaring faults to make it recommendable. And while not the best performing prediction model Random Forest does a decent job across the board for what it needs to do. KNN on the other hand was a surprising disappointment for me in this evaluation as it has in the past been my go to solution for building effective and straightforward models, producing great prediction results.

Model Performance Suggestions

While the performance of the Random Forest algorithm was satisfactory it did not reach the highest level for prediction accuracy. To help workaround this challenge and improve the experience for users of the app I have thought of a couple ways to improve the odds of the RF model working in their favor.

**Perform an evaluation of WAP utilization** – One of the striking things I discovered when evaluating the data was the low use of WAPs (Wireless Application Protocols). From reviewing WAP use across all three buildings the average level of activity remains around 3% with less than 20 devices used (see figure 16, Training Data Research Base).

|  |  |  |
| --- | --- | --- |
| Building | WAP# | Utilization |
| TC | 19 | 3.84% |
| TD | 17 | 3.19% |
| TI | 16 | 3.04% |

**Figure 16**. WAP utilization across

One way to do this is to create a team project focused on evaluating the strength and positioning of WAPS throughout the TC, TD, and TI buildings. The data collected will need to be evaluated for completeness and accuracy as it is possible for human error to alter the results, lowering the usefulness of the end product. Also it will be beneficial to map the WAPs across the three buildings along with providing their radius of influence. [[3]](#footnote-3)Doing this will ensure that there are no weak spots in WAP coverage and if more devices were active, than that would allow for more attribute data to be collected which would improve the quality of data collected for model creation and tuning, and ultimately improve the findability of users with the locater app.

**Improve model performance with check points** – Based on my research I found that even with the best predictive models available the ability to triangulate someone or something’s position may still be limited without the assistance of onsite resources to help correlate the user’s location. My suggestion is that each of the three buildings be equipped with check points at every entrance. This would be accompanied with users either having RFID (Radio-frequency identification) chip devices or would use the buildings’ locator app to interface with the check points transmissions to define where the users are within the buildings. With the material I have read it is possible that the checkpoint data could be used in conjunction with a user’s smartphone and the prediction model to dramatically improve the locator app’s accuracy. [[4]](#footnote-4)

1. *UJIIndoorLoc*. Arnau, Benedito-Bordonau, Huerta. 2014. http://ieeexplore.ieee.org/document/7275492/. [↑](#footnote-ref-1)
2. *The Best Intro to Data Science Courses*. Venturi, David. 2017. https://www.class-central.com/report/best-intro-data-science-courses/. [↑](#footnote-ref-2)
3. # *Enhancing Wi-Fi fingerprinting for indoor positioning using human-centric collaborative feedback.* Chen, Hoeber, Luo. 2013. https://link.springer.com/article/10.1186/2192-1962-3-2

   [↑](#footnote-ref-3)
4. ## *A Novel Approach for Wi-Fi Fingerprinting Using Logical Sequences of Intelligent Checkpoints.* Hofer, Retscher. 2015. https://www.isis.tuwien.ac.at/node/17016.

   [↑](#footnote-ref-4)