Springboard Data Science Career Track
Capstone Project 2: Final Report

Predicting Check-ins of Foursquare Users in Tokyo

Kevin Limkrailassiri

1 Introduction

Foursquare's slogan is "Foursquare helps you find the places you'll love, anywhere in the world". Since the launch of the Foursquare mobile app in 2009, Foursquare has helped 60 million users discover new exciting places worldwide[1]. The app provides personalized recommendations of places to visit in the vicinity of a user's current location based on "previous browsing history, purchases, or check-in history".[2] As a result, the Foursquare app has gained popularity for helping users to discover brand new places that match their interests.

Given the widespread use of Foursquare, the tens of millions of Foursquare check-ins accumulated since 2009 present an interesting collection of data from which useful insights and predictions may be unearthed. Each check-in contains data about the user, time and day of check-in, venue, and GPS coordinates. It is conceivable that by studying the past check-in activity of each user, specifically the types of venues visited in certain parts of a city at certain times of the week, predictions may be made for the types of venues the user may be interested to visit in the future, including venues the user has never visited before. This has potential benefit both for consumers and businesses, as consumers can receive targeted recommendations for places they may want to visit before they even start making any plans, and businesses can anticipate periods of high traffic and tailor promotions for the exact times when more users are expected to check-in to their store.

Toward this end, we study Foursquare check-in data for the city of Tokyo and attempt to predict the type of venue that a user checks into based on the time of day, day of the week, their GPS location, and their past usage history.

2 Data Acquisition and Cleaning

The data set of Foursquare check-in data for Tokyo is obtained from Kaggle in .csv format. The data set contains 573,703 individual check-ins for the time period of April 4, 2012 to February 16, 2013. Each check-in is described by eight features: the user who checked-in (`userId`), the specific venue visited (`venueId`), the category of this venue (`venueCategoryId` and `venueCategory`), the GPS coordinates of this venue (`latitude` and `longitude`), the time zone offset from Coordinated Universal Time or UTC (`timezoneOffset`), and the time of check-in according to UTC (`utcTimestamp`). First, the index is set to datetime using the local time for Tokyo based on `utcTimestamp`. Next, redundant columns `venueCategoryId` and `timezoneOffset` are removed. Then, the column names are renamed to be shorter and with lowercase characters only. This results in a dataframe containing five features (`userid`, `venueid`, `venuecat`, `lat`, and `long`). There are no NaNs in this dataframe.

Each venue is categorized into one of 247 unique venue categories. Given that the ultimate goal of predicting venue type is a multiclass classification problem, it would be beneficial to somehow reduce the number of venue categories as this will likely help us reduce the number of features to be analyzed by the machine learning model. Therefore, we group the 247 venue categories into 9 root categories: Arts & Entertainment, College & University, Food, Nightlife Spot, Outdoors & Recreation, Professional & Other Places, Residence, Shop & Service, and Travel & Transport. The mapping of venue categories to these primary root categories was done with the aid of a look-up

table provided in JSON format from Foursquare. While the majority of the venue categories were successfully categorized into the root categories, those venue categories that were not needed to be mapped to the appropriate root category manually.

3 Exploratory Data Analysis

We begin exploratory data analysis by plotting the coordinates of all venues visited for the entire time span of the recorded data. The map of Figure 1 denotes each visited venue as a white point. For reference, the locations of the ten busiest train stations in Tokyo are labeled. It is clear that much of the activity is concentrated in the heart of Tokyo and associated with these train stations. This is consistent with the fact that trains and subways are the primary mode of transportation in

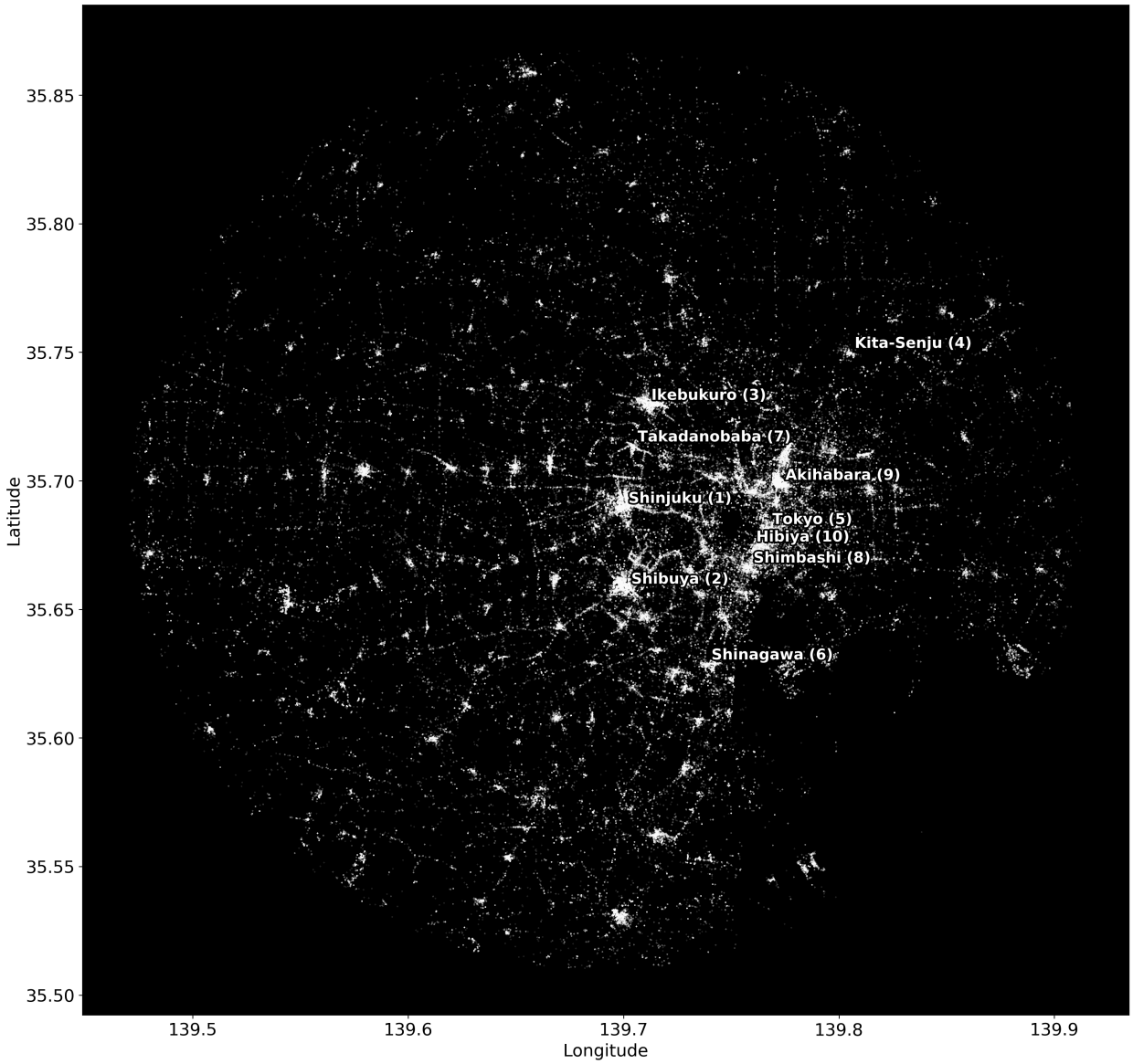

Figure 1: Coordinates of all venues visited in Tokyo. The 10 busiest train stations are also plotted.

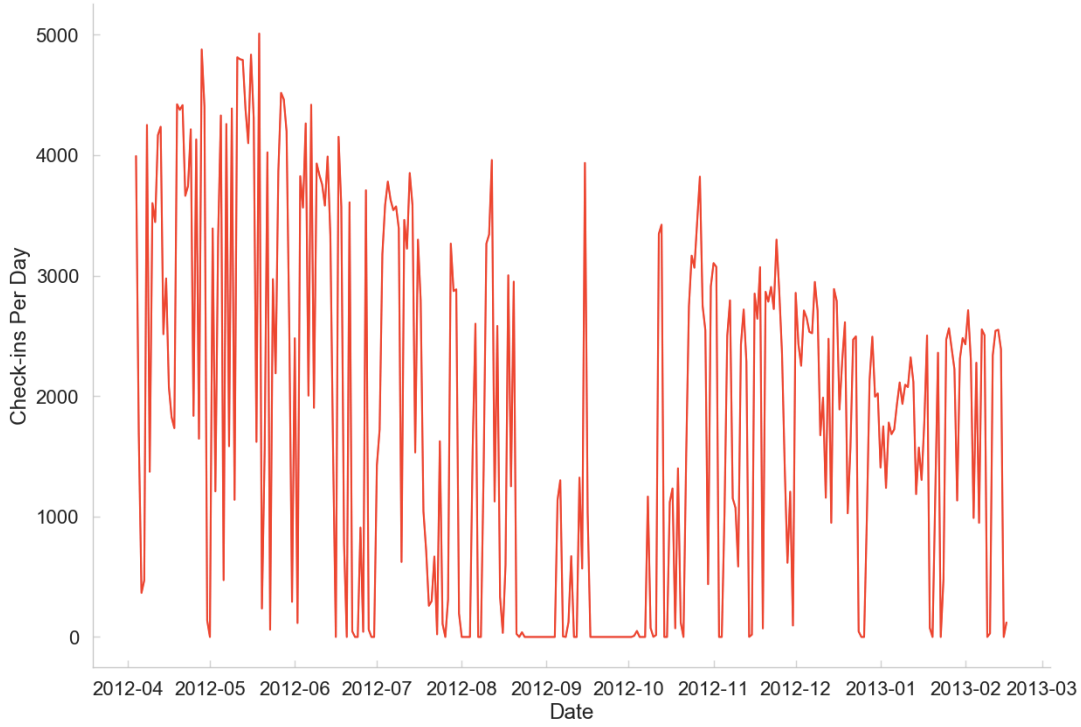

Figure 2: Average daily check-ins for the entire time span of the data set.

Tokyo, and these train stations are located near commercial and administrative centers where users likely frequent for work or leisure. Other prominent clusters of white dots are also likely transportation hubs.

Next, we consider the usage of the Foursquare app over time. Figure 2 plots the average daily check-ins for the entire time span of the recorded data. The data appears to fluctuate wildly, reaching as high as 5,000 check-ins to as low as zero. There is a general trend of declining usage from mid-May 2012 through February 2013. Notably, there are lengthy sequences of days with zero reported check-ins from mid-August 2012 to mid-October 2012. It is unlikely that all 2,293 Tokyo users collectively decided not to use the Foursquare app on these days. Possibly, it was the decision of Foursquare to exclude the check-ins for these days from this data set. Given the likelihood of missing data, it would not be helpful to analyze the monthly trend of check-ins considering that the monthly averages as calculated from the data would not reflect the true monthly averages. Therefore, we will restrict our time series analysis to weekly and hourly trends.

Figure 3 compares the total number of check-ins by day. We note that check-ins for the weekdays are fairly consistent with the most check-ins taking place on Mondays, while the check-ins for Saturday and Sunday are noticeably higher than for weekdays. This makes intuitive sense; we would expect users to follow a certain pattern of usage during the weekdays that is distinctly different from the weekends due to their working schedule. It may be helpful, then, to distinguish weekday and weekend activity in subsequent exploratory data analysis in order to isolate trends that are unique to weekdays and weekends.

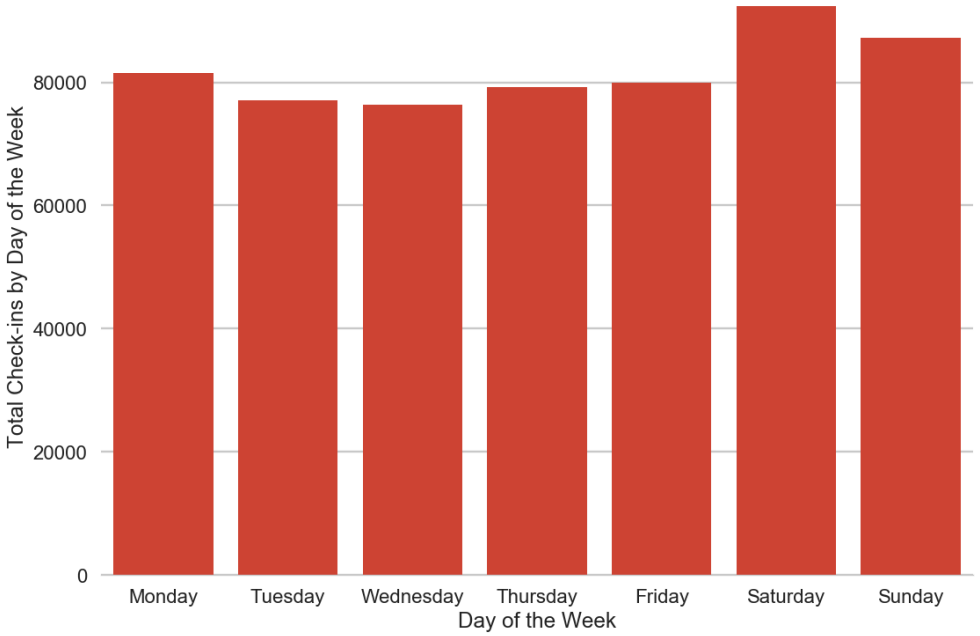

Figure 3: Total check-ins by day of the week for the entire time span of the data set.

Figure 4 shows the distribution of check-ins according to hour of the day. There are notable peaks at 8am, 12pm, and 6pm, likely corresponding to morning commute, lunch break, and evening

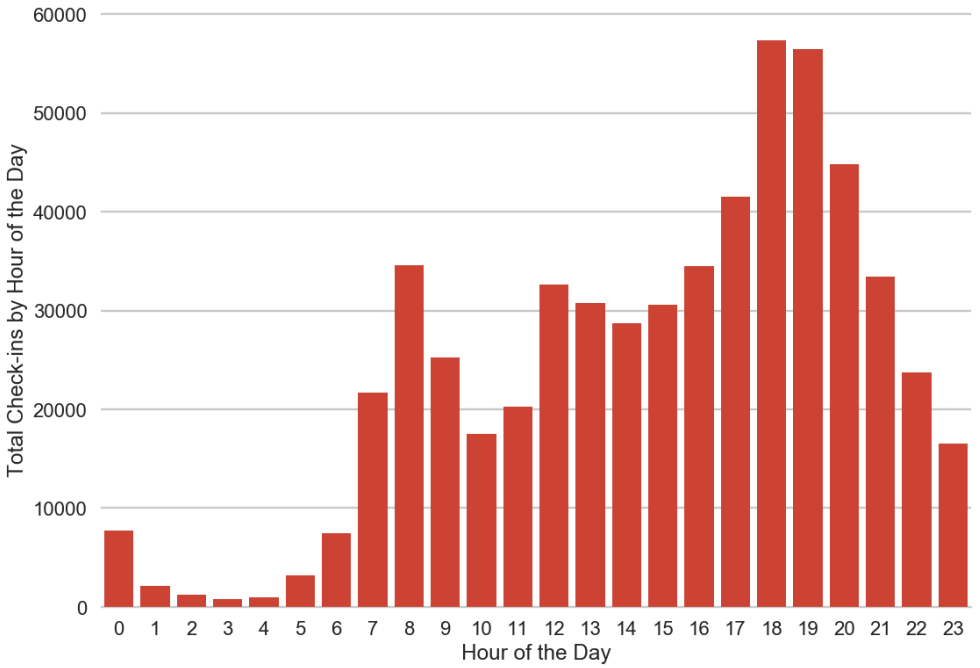

Figure 4: Total check-ins by hour of the day for the entire time span of the data set.

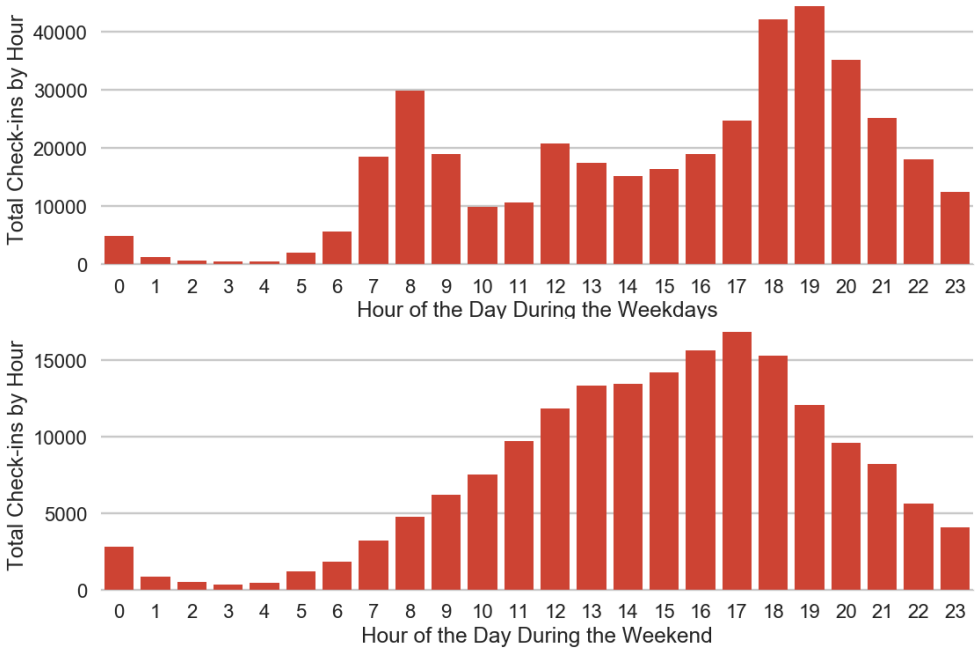

Figure 5: Comparison of check-ins by hour of the day for weekdays and weekends.

commute hours during the weekdays. The hour of least activity is at 3am. However, these trends may not be generally representative of every day of the week. Therefore, to surface check-in activity specific to weekdays and weekends, we plot hourly activity separately in Figure 5. Indeed, we find that the check-in activity is distinctly different between weekdays and weekends. Once again, the peaks noted in Figure 4 are present in the distribution for weekday check-in activity, as expected. However, the distribution for weekend activity does not show any of these peaks, but rather a smooth increasing trend peaking at 5pm with the lowest point at 3am. Indeed, after a long week of work, it is good to sleep in and get recharged before a night out on the town!

Given these check-in trends, what kinds of venues do users visit? Figure 6 provides a comparison of check-ins for the 9 root venue categories during the weekdays and weekends. It is evident that Travel & Transport is the most popular venue category, followed by Shop & Service and Food. A comparison of the weekday and weekend check-ins shows that users' check-in priorities remain unchanged from weekdays to weekends, and there is little change in the proportion of check-ins

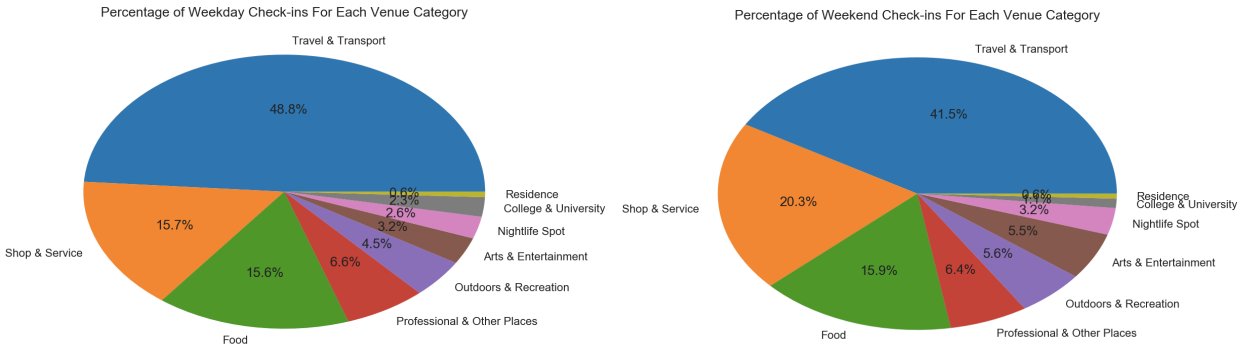

Figure 6: Comparison of check-ins on weekdays and weekends by root venue categories

for each of the venue categories. The fact that the data set is clearly imbalanced is something to keep in mind when choosing classifiers for multiclass classification.

To understand where these venues may be located, the coordinates of these venues are plotted in Figures 7 and 8. It is difficult to discern the contrast between a plot of all venues visited during the weekdays and another plot of all venues visited during the weekends in the style of Figure 1 because both plots show similar clusters of white points. However, it is much easier to visualize the difference in check-in activity if we were to plot the locations of venues visited only during the weekdays and another plot of the locations of venues visited only during the weekends. Figures 7 and 8 serve this purpose. Remarkably, two distinct patterns emerge. The venues visited only during the weekends tend to be more concentrated in the heart of Tokyo, implying that many users find work in this area and would not frequent these specific locations during the weekends, while the venues visited only during the weekends are uniformly distributed across all of Tokyo.

While the above visualizations provide an overview of overall check-in activity for all users, the insights from these visualizations may not be generally applicable for every user. Each user will have his own preferred shortlist of venues along with preferred times and days to visit these venues. Therefore, it would be helpful to zoom-in on the check-in activity of individual users to see if we may uncover some patterns in check-ins that will help us design a supervised learning model.

Taking userid 718 as an example, we note check-in activity on Friday, May 18, 2012 and Friday, May 25, 2012, as depicted by the red dots on the maps of Figure 9, and similarly, check-in activity on Saturday, May 19, 2012 and Saturday, May 26, 2012. One observation common to all maps is the sparse spatial distribution of check-in locations. Taking the example of the map for May 18, 2012 in Figure 9, the map shows seven red dots representing nine total check-ins spanning a significant portion of Tokyo. It would not be surprising if other users' check-ins were similarly distributed spatially, as most people living in Tokyo are heavy users of the train and subway systems and would then be expected to check-in to various venues along their daily itinerary. This clear spatial separation of check-ins can be leveraged for predicting future check-ins since the list of likely venue types to be visited in the future can be narrowed down significantly by knowing what venues the user visited in the past at around the same time of the week and in the vicinity of his current location. Comparing the maps for May 18 and May 25 as well as May 19 and May 26 confirms that some venues are indeed frequented on a weekly basis.

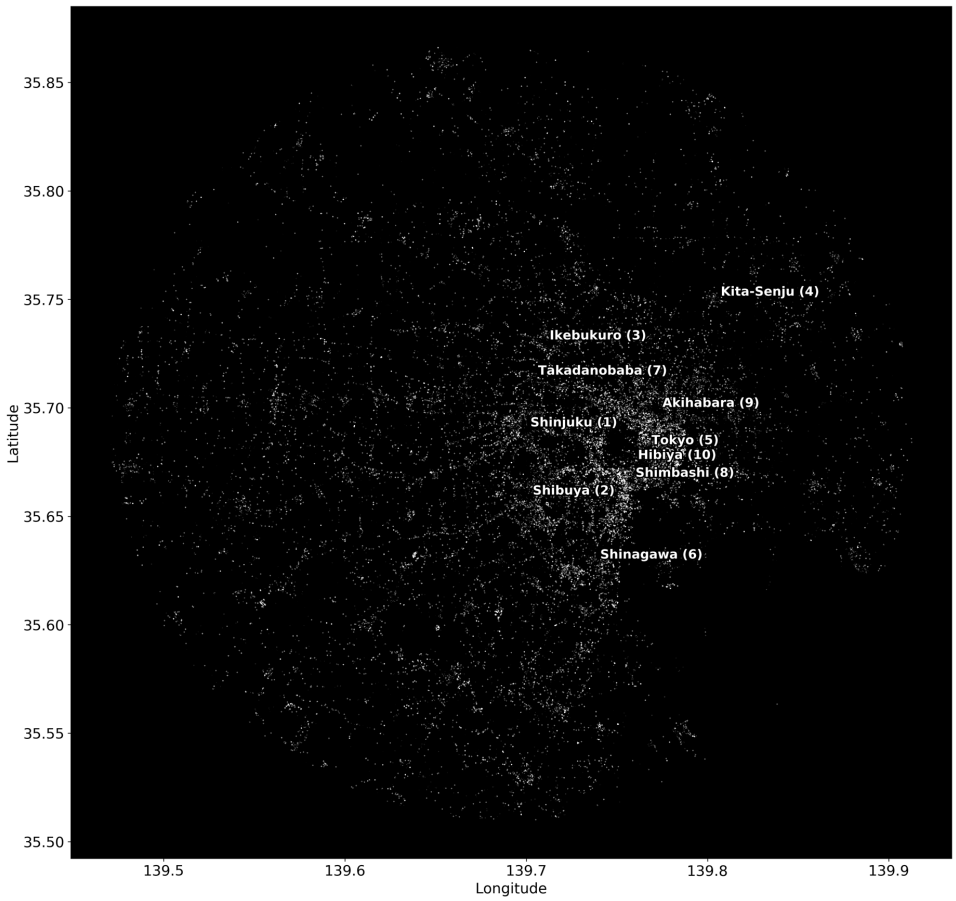

Figure 7: Coordinates of all venues visited only during weekdays.

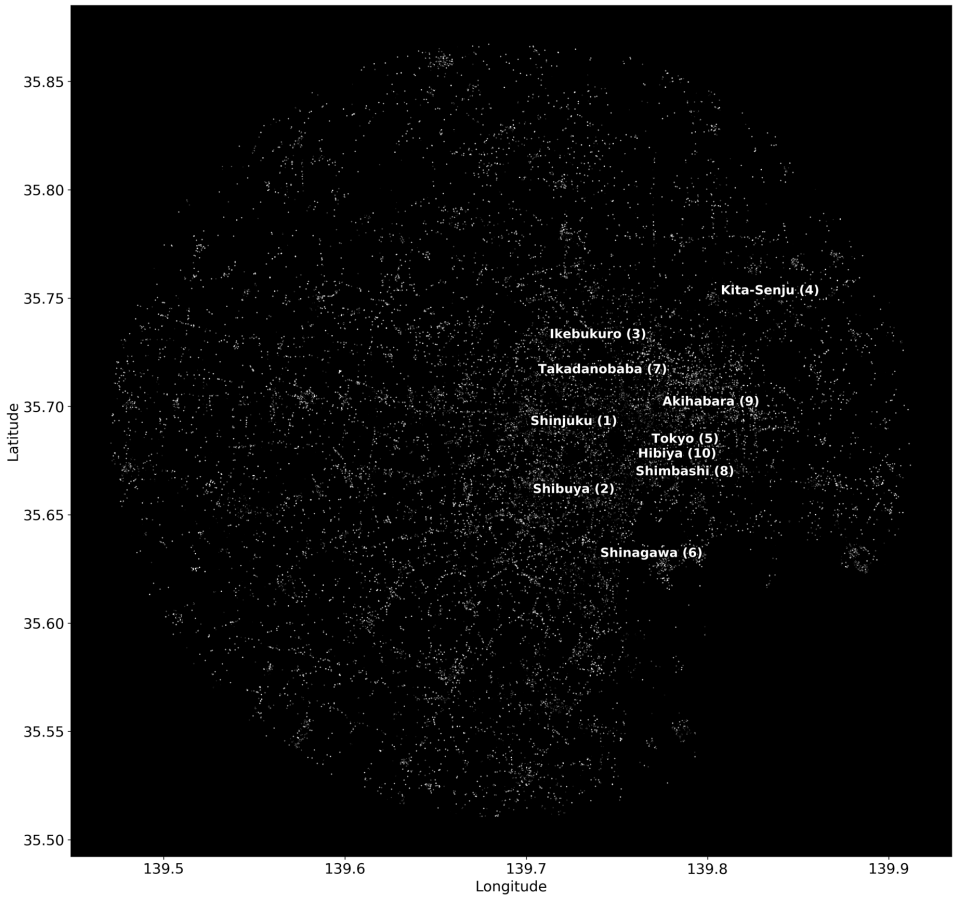

Figure 8: Coordinates of all venues visited only during weekends.

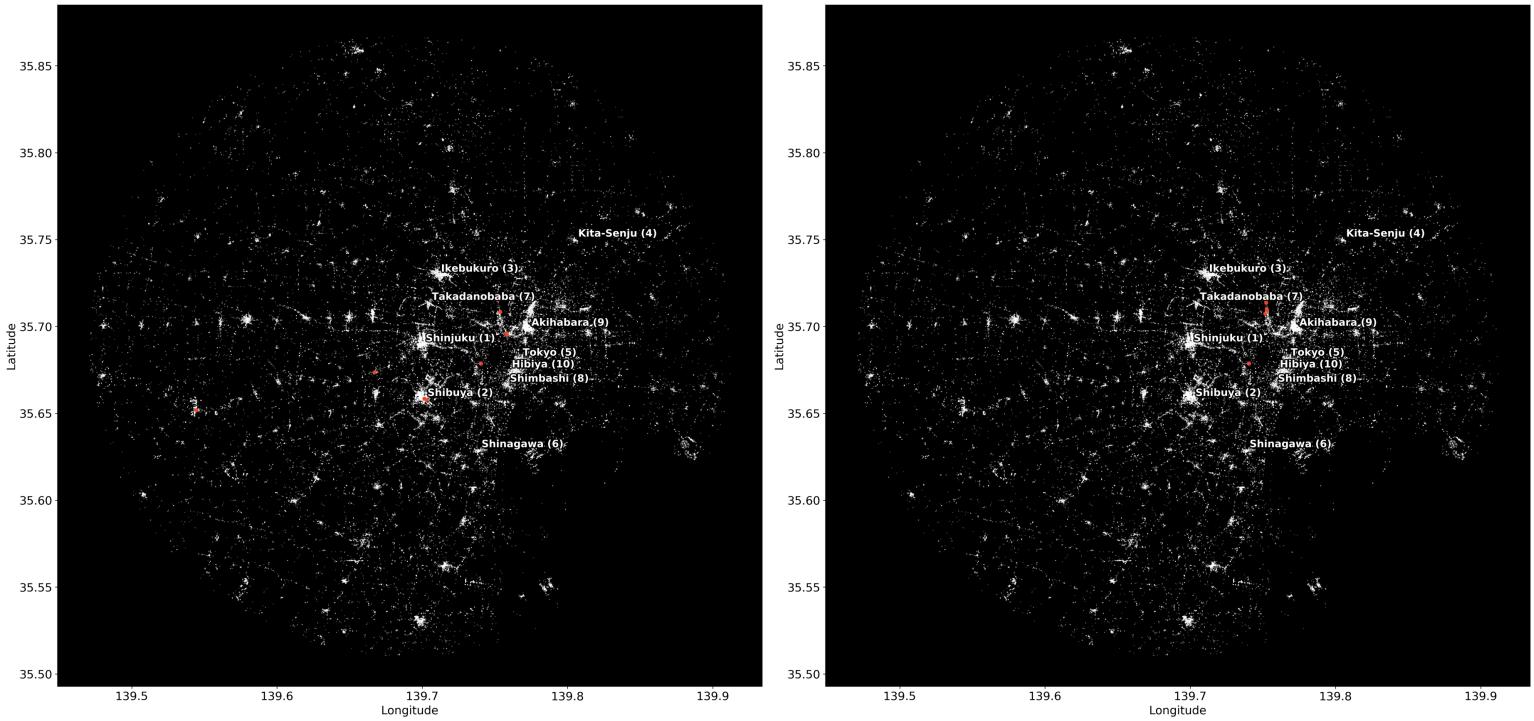

Figure 9: Venues visited on Friday, May 18, 2012 (*left*) and on Friday, May 25, 2012 (*right*) by userid 718.

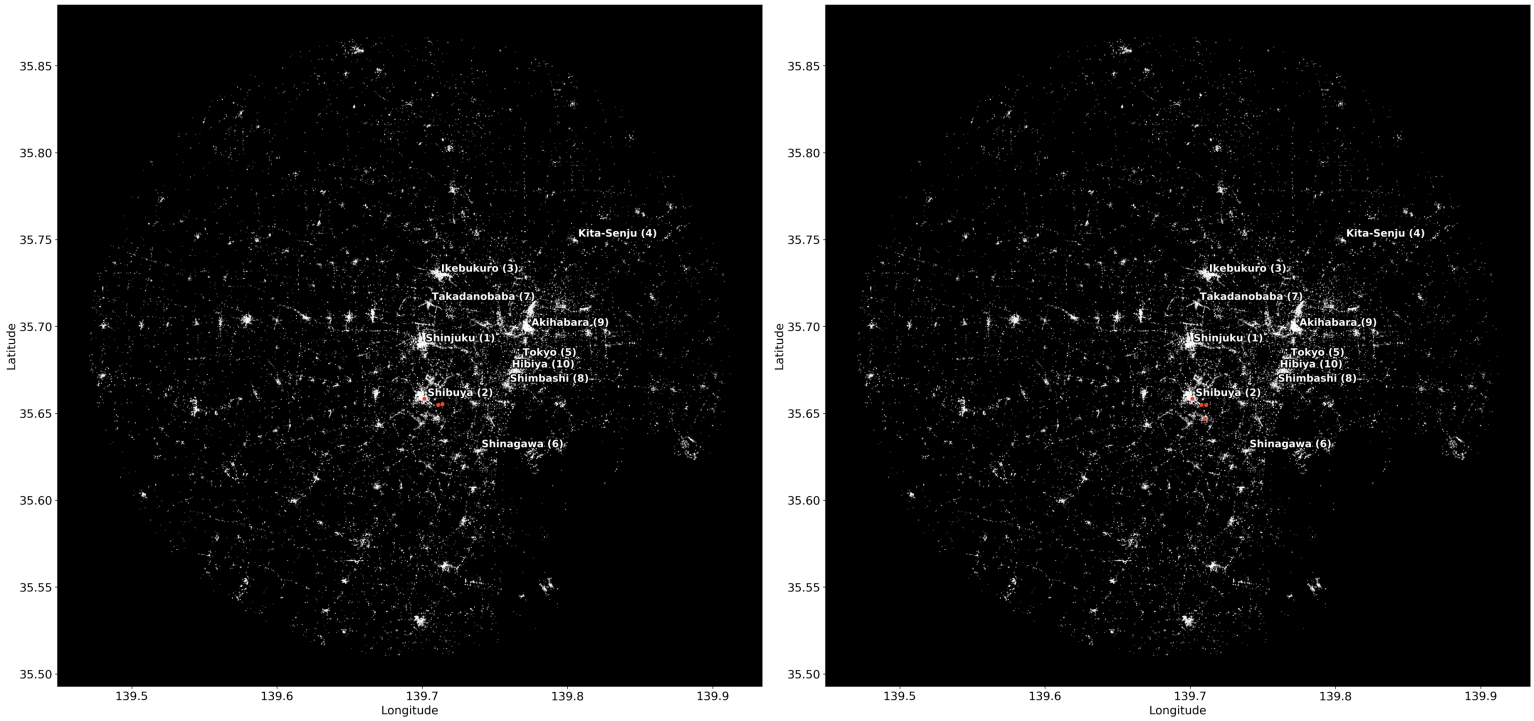

Figure 10: Venues visited on Saturday, May 19, 2012 (*left*) and on Saturday, May 26, 2012 (*right*) by userid 718.

4 Modeling

Based on the observations gained from visualizing check-in activity for individual users, we devise the following method for designing a multiclass classification model for supervised learning. First, eight features will be engineered according to each of the eight root venue categories. Specifically, for every observation, a check-in described by unique timestamp, userid, and GPS coordinates, each feature will contain the number of check-ins previously made by the user for a specific root venue category within a 1-kilometer radius of the current GPS location and within a 2-hour window of the current time from previous weeks. The frequencies of these features will then be normalized from zero to one. For example, considering the check-in of a Food venue by userid 868 on Wednesday, April 4, 2012 at 3:22am at 35.715 latitude and 139.800 longitude, all previous check-ins by this userid taking place on weekdays between 2:22am and 4:22am and within a 1-km radius of 35.715, 139.800 will be counted and categorized into each of the eight root venue categories. The target variable will contain the root venue categories encoded from 0 to 8 as shown in Table 1. Several multiclass classification algorithms will be used, and for each algorithm the data will divided into a train-test split of 75:25. Since the last 100,000 check-ins of the data set will be used for machine learning, this amounts to 75,000 check-ins in the train set and 25,000 check-ins in the test set. The last 100,000 check-ins were chosen by considering the history of check-ins up to that moment in the data. Capturing all 573,703 check-ins is not necessarily helpful, as many of the check-ins in the earlier portion of the data have very little if any prior history, which means the features would not contain very much information for a classification model to make a well-informed prediction. It was observed that at around the last 100,000 check-ins of the data set, the number of check-ins with no prior history began to level off. Therefore, it was decided that this would be a good starting point for the train set. Hyperparameter tuning will be performed with a grid search algorithm. Recalling that the data set is imbalanced, the performance metrics of interest for each classifier will be precision, recall, and F_1 score. These metrics will be calculated both by class and overall using macro averaging.

Root Venue Category	**Label**
Arts & Entertainment	0
College & University	1
Food	2
Nightlife Spot	3
Outdoors & Recreation	4
Professional & Other Places	5
Residence	6
Shop & Service	7
Travel & Transport	8

Table 1: Label encoding of root venue categories.

4.1 Logistic Regression

A logistic regression model was employed yielding accuracy of 0.606, precision of 0.598, recall of 0.398, and F_1 score of 0.457. The confusion matrix of Figure 11 provides deeper insight into these scores. The darker cells of the last column indicate that many false positives were predicted

```
Tuned hyperparameters:
{'C': 3.3598182862837809, 'penalty': 'l2'}

Accuracy: 0.606

              precision    recall  f1-score   support

           0       0.58      0.21      0.31       930
           1       0.61      0.50      0.55       318
           2       0.56      0.37      0.45      3988
           3       0.61      0.24      0.35       738
           4       0.67      0.40      0.50      1167
           5       0.55      0.23      0.33      1610
           6       0.64      0.35      0.45       153
           7       0.52      0.38      0.44      4220
           8       0.63      0.90      0.74     11876

         avg       0.60      0.40      0.46     25000
```

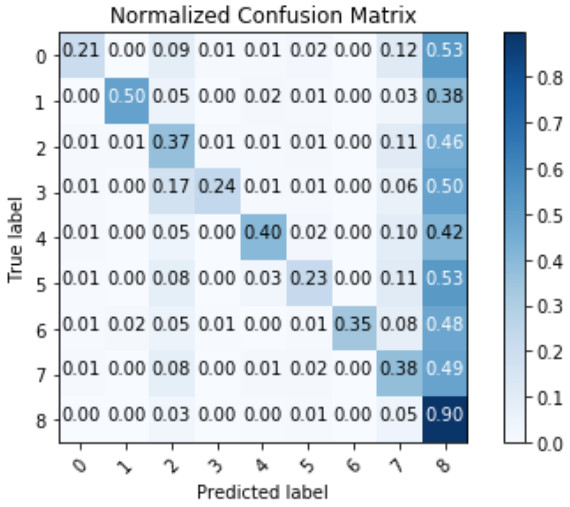

Figure 11: Results of logistic regression model

for Label 8, which corresponds to Travel & Transport. This means that observations that are truly Travel & Transport are not linearly separable from all other observations. The fact that all other categories have a large proportion of observations mislabeled as Travel & Transport is supported by the fact that the data is imbalanced with Travel & Transport composing nearly half of all check-ins. Many observations that are not actually Travel & Transport but bear similar feature values would then be mislabeled as Travel & Transport because the accuracy score tends to benefit from predicting this majority category. Similarly, the model also mislabels many check-ins as Labels 2 and 7, corresponding to Food and Shop & Service. Therefore, it is evident that accuracy alone is not a good indicator of the performance of the logistic regression model for this case. Rather, precision, recall, and F_1 score will be more accurate predictors of a classifier's ability to handle imbalanced data sets.

4.2 Gaussian Naïve Bayes

The Gaussian Naïve Bayes model produced a nearly identical accuracy of 0.601. The precision, recall, and F_1 scores were also very respectable at 0.505, 0.492, and 0.485, respectively. Similar to the logistic regression model, the confusion matrix of Figure 12 shows many observations mislabeled as Travel & Transport, Food, and Shop & Service. A closer look into the scores shows that the performance of the models appears roughly similar and differs most for the venue categories with the least number of actual check-ins: Label 1 (College & University), Label 3 (Nightlife Spot), and Label 6 (Residence). For these three categories, the logistic regression model exhibits better precision, while the Gaussian Naïve Bayes model exhibits better recall. This implies that the Gaussian Naïve Bayes model has more difficulty with imbalanced data sets, preferring to label actual minority classes as other classes. Therefore, some pre-processing step to handle the imbalanced data set should hopefully help boost the precision score. It is worth noting that Naïve Bayes is an especially useful classifier owing to its simplicity and ability to generate predictions quickly. The other classifiers in this study required anywhere between a few minutes to several hours to run, while the Gaussian Naïve Bayes model required less than one second while providing very competitive metric scores.

Accuracy: 0.601

	precision	recall	f1-score	support
0	0.44	0.33	0.38	930
1	0.49	0.70	0.57	318
2	0.60	0.33	0.42	3988
3	0.39	0.38	0.38	738
4	0.56	0.49	0.52	1167
5	0.46	0.37	0.41	1610
6	0.42	0.62	0.50	153
7	0.53	0.37	0.43	4220
8	0.65	0.85	0.74	11876
avg	**0.51**	**0.49**	**0.49**	**25000**

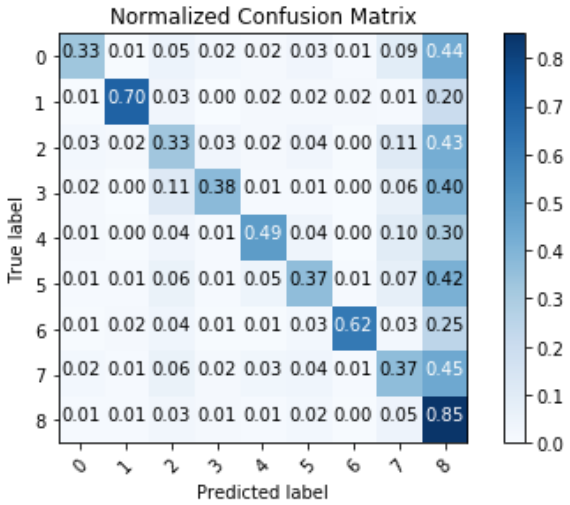

Figure 12: Results of Gaussian Naïve Bayes model

4.3 k-Nearest Neighbors

The k-nearest neighbors model achieves an accuracy score of 0.605 with precision, recall, and F_1 scores comparable to logistic regression and Gaussian Naïve Bayes. Given the classification algorithm for k-nearest neighbors, the premise for making predictions is simple: a check-in will typically be of the same venue category as other neighboring check-ins in the feature space, and given the design of the features, the most likely venue category is the one that has been checked into most frequently in the past. Among the three classifiers studied thus far, the close resemblance in confusion matrices as well as in the values for accuracy, precision, recall, and F_1 score suggest that further gains in the performance of these classifiers will hinge upon improving the existing features and even introducing new ones, thereby providing the classifiers more nuanced information to make better informed predictions. The optimal value for hyperparameter n_neighbors, obtained through grid search, was 55.

```
Tuned hyperparameters:
{'metric': 'euclidean', 'n_neighbors': 55}
```

Accuracy: 0.605

	precision	recall	f1-score	support
0	0.52	0.27	0.35	930
1	0.54	0.67	0.60	318
2	0.55	0.38	0.45	3988
3	0.54	0.27	0.36	738
4	0.63	0.43	0.51	1167
5	0.50	0.27	0.35	1610
6	0.58	0.42	0.49	153
7	0.50	0.42	0.46	4220
8	0.65	0.86	0.74	11876
avg	**0.56**	**0.44**	**0.48**	**25000**

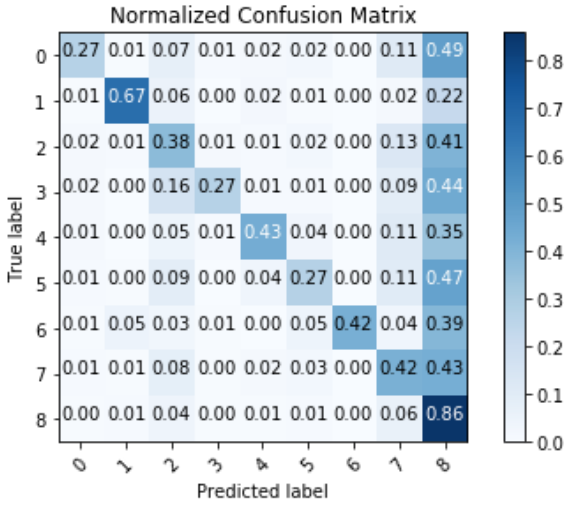

Figure 13: Results of k-nearest neighbors model

4.4 Ensemble Models

In this section, we discuss the results of three ensemble models: random forests, extreme gradient boosting, and extremely randomized trees. Interestingly, the metric scores for all three classifiers are almost exactly identical, and the confusion matrices of Figure 14 look very similar. Although extreme gradient boosting and extremely randomized trees are usually expected to outperform random forests, there is nothing further for these two classifiers to mine from the features apart from what random forests is able to find already. As mentioned earlier, a critical future work for this project is more careful engineering of features and addition of new ones, but for the time being, the features present a firm bottleneck that presents the classifiers from distinguishing themselves from one another in terms of performance. The precision, recall, and F_1 scores of the ensemble models are ~0.57, ~0.43, and ~0.47, respectively.

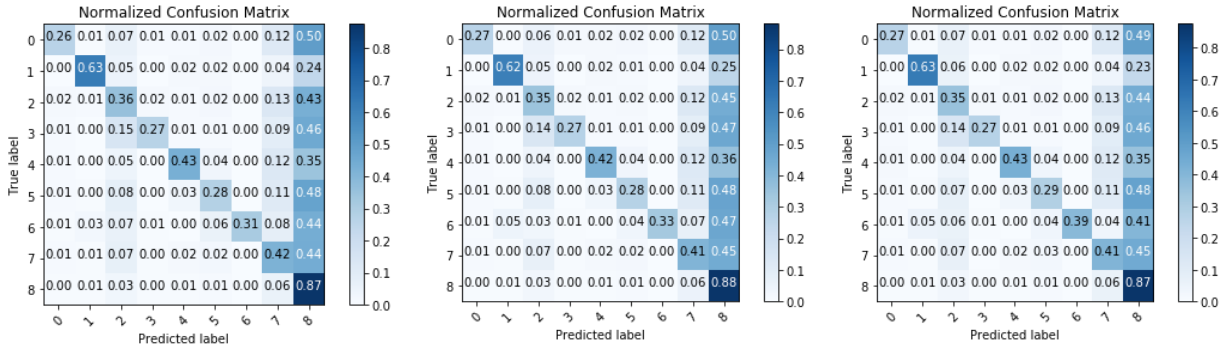

Figure 14: Confusion matrices of ensemble models: (*left*) random forests, (*middle*) extreme gradient boosting, and (*right*) extremely randomized trees.

4.5 Cascaded Multiclass Classification

The repeated observation of numerous false positives for Travel & Transport among the above classifiers reflects the difficulty of performing multiclass classification on imbalanced data sets. A common solution is to perform binary classification of the majority class against all other classes and then perform multiclass classification on the remaining classes with the majority class discarded. For this cascaded approach, the metric of interest will be F_1 score, as we are interested in evaluating a classifier's robustness in distinguishing between majority and minority classes.

Taking the cascaded k-nearest neighbors classifier as an example, training and testing of the data set is performed as follows. For the first cascade, all check-ins belonging to Travel & Transport will be encoded as 0 or 1 with 1 denoting Travel & Transport and 0 denoting all other categories. The classifier is trained and tested to generate predictions distinguishing check-ins for Travel & Transport against other check-ins. To prepare for the second cascade, all check-ins for Travel & Transport in the train set are discarded, and all check-ins in the test set predicted to be Travel & Transport are discarded. The observations corresponding to the remaining venue categories are fed into the second cascade where multiclass classification will be performed as usual.

For each of the classifiers tested, all of them outperformed their non-cascaded counterparts in terms of recall and F_1 score. On average, cascading improved the F_1 score of each classifier by

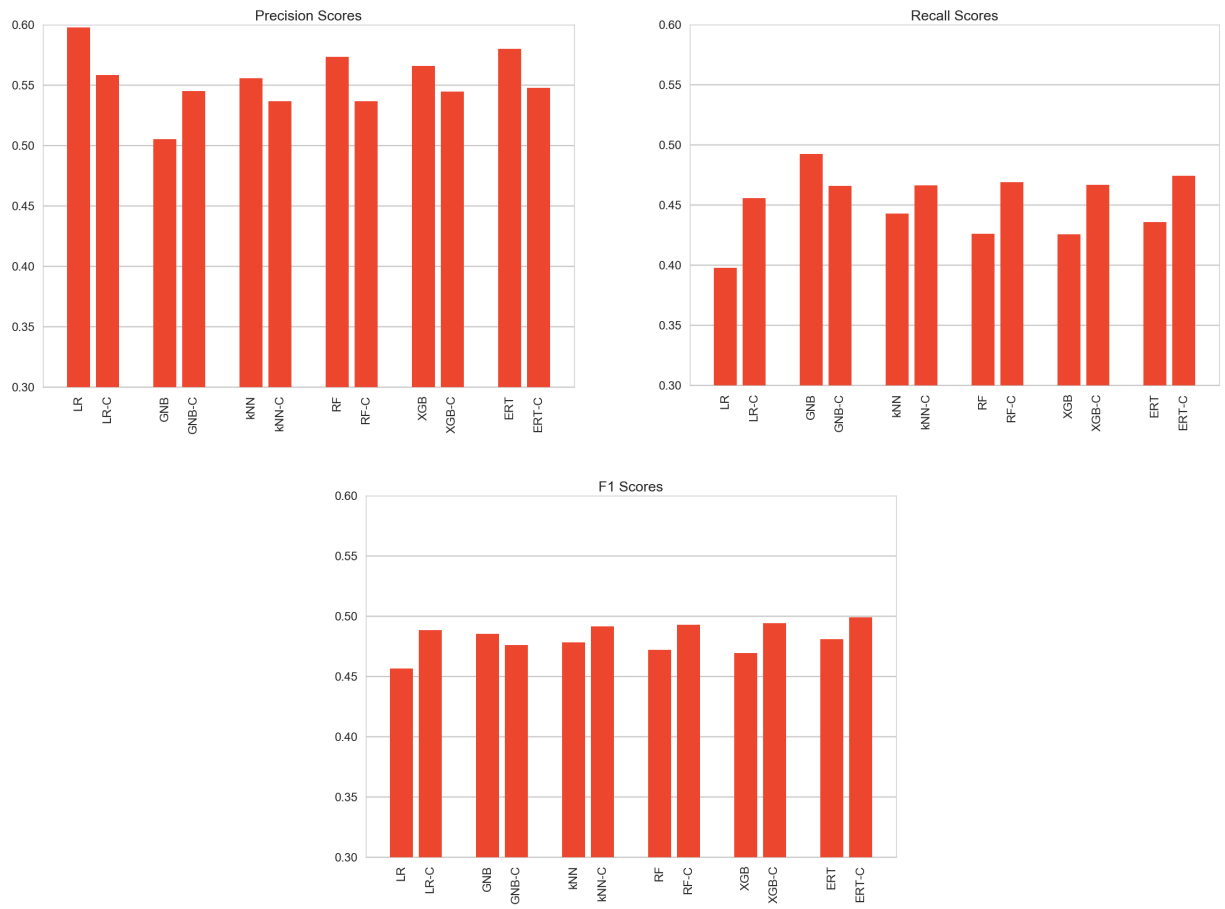

Figure 15: Comparison of precision, recall, and F_1 scores for all classifiers with and without cascades.

2%. Indeed, separately classifying the majority class helped each classifier perform better for the entire test set.

5 Discussion

It is not an overstatement that predicting venue category will always remain a challenging task. It is interesting to note that all metrics—accuracy, precision, recall, and F_1 score—seemed to plateau towards a certain score regardless of the classifier used. This suggests that the features were engineered in such a way that predictions were accurate for most check-ins, but for the rest of the check-ins, the features could not provide enough clues for the classifiers to make reasonable predictions.

Provided that improvements in the engineering of the features would possibly improve the performance of the classifiers, it is worth asking how well the classifiers currently perform in comparison to the literature. Yang *et al*. developed a spatial temporal activity preference (STAP) model to predict venue category for the same data set from Foursquare[3]. Rather than classifying check-ins according to the nine root venue categories, their model performs more granular predictions, choosing from among a set of 247 venue categories. Their best performing model

achieved an accuracy of ~0.53 with average F_1 score of ~0.34. It is understandable that their scores would be lower than those found above since they were tasked with the far more difficult challenge of classifying check-ins into 247 different venue categories. Nonetheless, the scores of Yang's study also suggest that the precision, recall, and F_1 scores of the above classifiers fared decently well and are a good first step towards accurately predicting check-ins of Foursquare users in Tokyo.

Recalling that features were originally engineered to account for a user's past activity for the same time frame and in the vicinity of his current position, we are led to question whether past check-in history provides sufficient information to predict future check-ins. A close examination of misclassified check-ins reveals that more information, and therefore more features, would be helpful. It turns out that the majority of misclassified check-ins occur when the user checks into a venue that is highly unlikely given their past check-in history. Although this certainly makes accurate prediction of check-ins more difficult, the model must be robust enough to accommodate such check-in behavior because the Foursquare app actually encourages users to explore new places that they have yet to visit, which means that sole reliance on past history is not enough to yield accurate predictions. Therefore, the classifier must be designed to consider other determinants that lead users to check into venues they visit infrequently or even venues that are entirely brand new to them. One possible solution is to include features that account for the typical sequence of venues visited in a certain area, since, for instance, users may always check into a bar after a late-night dinner at a restaurant during the weekday evenings. Another solution is simply to collect more data for each user. It is possible that there are not enough check-ins in a user's history to fully represent what the user typically checks into at a certain time and place. In some cases, the user had no prior history at all. It was thought that for these situations, the user's check-in could be predicted with the help of history from all other users for the same time window and GPS location; however, no improvement in precision, recall, or F_1 score was observed by using this approach.

All things considered, the cascaded classifiers present a promising solution for dealing with imbalanced data sets, improving F_1 score for every classifier compared to the non-cascaded case.

6 Conclusion and Recommendations

In conclusion, the Foursquare data set of user check-ins in Tokyo was employed for supervised learning. Given a user's current time and GPS location, the task was to predict what kind of venue the user would check into. Integer labels from 0 to 8 were used to denote the venue category visited. Features were designed according to the number of times a user checked into a particular venue category in the past, within a 2 hour window of the current time, within a 1-km radius of the current location, and for the current day of the week, either weekday or weekend. The last 100,000 check-ins of the data set were employed for multiclass classification. With a 75:25 train-test split, there were 75,000 check-ins in the train set and 25,000 check-ins in the test set. The initial results showed a typical accuracy of 0.60 with poor recall scores largely caused by the imbalanced data set. A cascaded classifier approach was implemented to first perform binary classification for the dominant venue category, Travel & Transport, and then follow up with

multiclass classification for the remainder of the check-ins. An average improvement of 2% in F_1 score was observed for all classifiers.
While the initial results appear promising, there are several potential avenues for improvement:

- **More features**. It is clear that the current set of features does not allow the classifiers enough traction in the data to make accurate predictions of non-obvious, borderline cases. Therefore, the current set of features can be engineered further, and more features may be introduced. More than simply looking at past check-in activity, more information may be gathered based on how each user orders his check-ins. If a user had just checked into College & University, would it be safe to assume that the next check-in would not be the same? If a user checks into Food late on a Saturday night, is there a good chance he would check into Nightlife Spot next? The fact that users typically arrange check-ins in a rational, ordered way presents an opportunity for continued feature engineering.

- **More data**. Gathering more check-ins would prevent the situation when a user has no prior history for a particular time and location. Currently, about 25% of the 100,000 check-ins used for supervised learning suffer from this issue; for these check-ins, predicting venue category is extremely difficult because the classifier has no prior information to use in order to make a prediction. With more data, these situations will hopefully be minimized.

- **Introduce new data sources**. Are there other data sources that can help boost the accuracy of our classifiers? Would accounting for current weather conditions help us predict check-ins more accurately?

7 References

[1] Weber, Harrison. “Foursquare by the Numbers: 60M Registered Users, 50M MAUs, and 75M Tips to Date.” VentureBeat, VentureBeat, 19 Aug. 2015, venturebeat.com/2015/08/18/foursquare-by-the-numbers-60m-registered-users-50m-maus-and-75m-tips-to-date/.

[2] Kim, Sam. “How Foursquare and Other Apps Guess What You Want to Eat.” Eater, Eater, 24 Apr. 2015, https://www.eater.com/2015/4/24/8486279/restaurant-recommendation-apps-foursquare-algorithms

[3] D. Yang, D. Zhang, V. W. Zheng and Z. Yu, "Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 1, pp. 129-142, Jan. 2015.