Inferring Human Activity Preferences

by Modeling Human Decision Segments

Master's Thesis

David Došenović

Aalborg University
Electronics and IT

Electronics and IT
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Inferring Human Activity Preferences by Modeling Human Decision Segments

Theme:
Scientific Theme

Project Period:
Spring Semester 2019

Project Group:
dt108f19

Participant(s):
David Došenović

Supervisor(s):
Manfred Jaeger

Copies: 1

Page Numbers: 58

Date of Completion:
June 12, 2019

Abstract:

A research on modeling the human activity preference was conducted in this thesis. The dataset being used in this research is a historical Foursquare dataset containing check-ins made throughout the period of ten and a half months in Tokyo, Japan. Thesis claims that the decision on our activities are influenced by so called decision segments. In order to model the human activity preference, multilayer perceptron machine learning model is used. Two modeling approaches are tested trying to capture decision segments by using different sets of features. The models are evaluated and results of the research are reported.

Elektronik og IT
Aalborg Universitet
http://www.aau.dk

AALBORG UNIVERSITET
STUDENTERRAPPORT

Titel:
Inferring Human Activity Preferences by Modeling Human Decision Segments

Tema:
Semestertema

Projektperiode:
Forårssemester 2019

Projektgruppe:
dt108f19

Deltager(e):
David Došenović

Vejleder(e):
Manfred Jaeger

Oplagstal: 1

Sidetal: 58

Afleveringsdato:
12. juni 2019

Abstract:

A research on modeling the human activity preference was conducted in this thesis. The dataset being used in this research is a historical Foursquare dataset containing check-ins made throughout the period of ten and a half months in Tokyo, Japan. Thesis claims that the decision on our activities are influenced by so called decision segments. In order to model the human activity preference, multilayer perceptron machine learning model is used. Two modeling approaches are tested trying to capture decision segments by using different sets of features. The models are evaluated and results of the research are reported.

Contents

Preface

Aalborg University, June 12, 2019

David Došenović
<ddosen17@student.aau.dk>

Chapter 1

Summary

Research on modeling the human activity preference was conducted in this thesis. In order to accomplish that, a research on related work described in CHAPTER 3 has been conducted. CHAPTER 4 then explains the dataset and its properties that have been exploited in order to model human activity preferences. The dataset being used in this research is a Foursquare dataset containing 573703 check-ins made throughout the period of ten and a half months in Tokyo, Japan. CHAPTER 5 lists the machine learning approaches and ideas used in modeling *decision segments* which were seen as parts of a human decision that, when combined, lead to the final decision. The description of conducted research is contained in the CHAPTER 6. This chapter explains the process of modeling the *decision segments* and, based on them, creating multiple decision models with respect to insights acquired from data analysis and stated technical approaches. In this chapter, the main concern is how to create the most accurate decision model and which *decision segments* should it consider while making a decision. First, two baseline decision models are created, after which a decision model using a multilayer perceptron has been developed. Two approaches were followed while developing this decision model, and different *decision segments* were considered in both. *Decision segments* have been represented in these models as sets of features.

After describing the process of developing, the thesis continues with CHAPTER 7 containing all tests and evaluation experiments conducted in this work, along with the environment and settings which were used throughout them. Results are reviewed and conclusions are made based on them. The last CHAPTER 8 provides a conclusion of this research along with the future work while stressing the challenges encountered through this research.

Chapter 2

Introduction

2.1 Motivation

Nowadays, most of the people living in urban environments are bounded to those environments by their residence and workplace, school or similar, depending on their age and stage of life. Considering this, along with the schedule of those responsibilities, both result in people not being able to leave far away from their residential area as otherwise, they would not be physically able to return to their responsibility the next day. Therefore, living in a way in which responsibilities or some other reasons tie people to their residence area, makes people live a life of daily repetition.
Living a life of daily repetition can suggest that people are making very similar decisions about their everyday actions.
People who are mostly doing the same activities on a daily basis will most likely rarely make exceptions in their schedules. Based on these regularly done activities, a number of conclusions could be made about people. One could make conclusions about a person's preferences based on the activities he/she had done throughout a longer period of time. This can be a powerful premise, which can be used to gather insight into people's behavior.
Besides this premise, a great deal of insight could be extracted from the patterns in which people are doing activities, frequencies of those activities, even the sequence in which those activities are done. All these insights, put in a temporal and spatial context can be a very potent way to discover people's habits and preferences. These discoveries can then be used, for example, by marketers to offer products that would be of users' interest at the time they are receiving the offer. This way marketers are creating revenues for their businesses, and users are provided with a better service.

This thesis attempts to make research on and conclude to what extent future

human actions can be predicted while having a history of their previous behavior. In order to accomplish this, the modeling of a human decision process will be conducted. The modeling of a human decision process will be done by considering different decision segments which are believed to influence people's final decision. After the models are done, they will be tested to show to what extent it is possible to accurately predict a person's decision. Models will be evaluated accordingly and reviewed.

Proving the hypothesis that human behavior can be predicted using data of their past behavior correctly within a Location Based Social Network check-in history environment, would have a powerful impact on marketers, businesses in general and LBSNs.
Furthermore, proving this research successfully would create space for additional interesting research on human behavior, and the development of recommendation systems or similar concepts on top of this research.

2.2 Problem Statement

Being provided with the Foursquare dataset which contains ten and a half months of user check-ins in Tokyo, this thesis will focus on modeling the users' activity preference. This will be done by exploiting users' check-in history. The user preference model will be composed of several different segments which are all assumed to influence the user's decision.
Multiple approaches will be used in order to model those segments, and discussion will be made on how each of the approaches, and on what scale, influence the user's decision. In order to evaluate and compare the results, simple baseline algorithms along with the more advanced multilayer perceptron will be used to model the user's activity preference.
Model quality will be evaluated by taking the check-ins from the test data set and comparing them to the prediction which will be inferred from the model and context of the check-in being predicted. If the user's predicted activity corresponds to the one which occurred in reality, the prediction will be considered successful. A check-in to a category of a venue user has visited will be considered as user activity.

Creating models that would predict users' activity interest given a certain context, might encourage the development of the recommendation system which would base its functionality on top of created models as understanding peoples' activities and their behavior would then enable a recommendation system to lead a person into doing activities that might be of interest to them given their current situation.

The main contributions of this Thesis are:

- study of different segments which influence people's decision on what is their aimed activity given certain context
- evaluation of considered decision segments and models
- proposal of the best performing solution that models human decision making

Chapter 3

Related Work

In order to get a better insight in the current state of the art and define a thesis aim more clearly, several papers on the topic of exploiting the data for different purposes, collected by location-based social networks, have been reviewed.

This chapter outlines goals, approaches, conducted experiments, and evaluation methods of the existing research related to the thesis topic.

Motivated by extracting the semantic meaning from GPS trajectories, rather than using raw GPS data, Li et al. in [12] introduce a novel approach in detecting users' interest and intention is given raw GPS data. Authors are proposing a Hierarchical-graph-based similarity measure (HGSM) framework, which bases its functionality on top of two assumptions from which information can be extracted. The authors claim that there is a huge importance in a sequence in which a user is visiting the location throughout the day. Furthermore, different scales of geographic spaces are hierarchically related and differ between themselves in importance. Authors are dealing with raw GPS data from which so-called Stay Points are being extracted. Fortunately for the problem of this thesis, there is no need for point of interest extraction, as the Foursquare dataset contains user check-ins. This paper gave an insight that sequence patterns in which people are visiting places might exist and could be relevant.

Yang et al. in [1] are dealing with a similar dataset as in [12], but for a different purpose - very similar to the one of this thesis. Given a set of users' historical behaviors (check-ins), their objective is to infer users' interest in a certain activity for a given time, around the current geo-location. One of the contributions of the paper is the Context-aware fusion framework. The framework combines prior separated spatial and temporal characteristics of user activity preference in the Location Based Social Network. In the process of spatial activity preference modeling

of the user, authors are introducing a concept of *Frequented region*, which represents a location favored by a user. Another concept introduced in the process is *Personal functional region*, a *Frequented region* where user's activities have higher *Preference bias* - preference towards a region. Concepts introduced in spatial activity preference modeling focus on the semantic meaning of the location, which might give an insight into why are users visiting those places. In the process of temporal activity preference modeling, authors leverage tensor factorization techniques which might be one of the approaches used in this thesis. The main outtake of this work is the highlighted importance of spatial segment which biases the people's decision.

Based on the Foursquare dataset containing a history of users' check-ins, Lee and Chung in [9] are tackling the problem of user similarity detection using location semantics. Names of the places where users are checking-in are being used in order to infer a user's intention. Contribution of [9] is a method to calculate a user similarity. The usage of hierarchical graph structure formed from locations and their categories is being proposed in the paper, which greatly inspired the utilization of the same approach in this thesis. Evaluation being reported shows that the proposed method performs much better in terms of precision, recall, and f-measure than often used Jaccard index.

Yin et. al. in [7] broaden the recommendation by proposing a location-content-aware recommendation system that offers a particular user not only a set of venues but also events. The system gives the recommendation based on users' personal interest and local preference. Using the local preference, this recommendation system can facilitate people's travel not only in an area familiar to them but also in a city that is new to them. Evaluations conducted in the paper show that the LCARS recommendation system, the one authors have developed, is performing excellent, both in terms of effectiveness and efficiency, especially when users travel to new cities. The reason for such a good performance in new cities is that the algorithm used takes into consideration the preferences of local people who usually have thorough knowledge about their neighborhood.

Another work using Foursquare data and stressing the importance of calculating LBSNs users similarity is presented in [15]. The main contribution of this paper is the demonstration of how the different data dimensions, captured on location-based social networks, can be combined to represent useful views of user profiles and to compute the similarity between users. Authors approach the similarity calculation problem by modeling different levels of user profiles extracted from the heterogeneous user feedback, namely check-ins and *tips*. This paper is mostly related to the topic of the thesis because it introduces *Spatial* and Enriched Spatial user profile models which are created using only check-in data. Besides this, this

work provided insight in how the LBSN data can be used in order to calculate user similarity which is one of the points of this thesis.

Bao, Zheng, and Mokbel in [8] are building a Location-based and preference-aware recommendation system that bases its functionality in social knowledge learning and personal preference discovery, among the others. Authors are working with LBSN check-in data gathered from multiple cities, which created a need for discovering local experts for each of those cities. They divide those experts even further to find local category experts. In order to find these local category experts, the HITS algorithm is applied. Furthermore, the authors discover a user's personal preference using the TF-IDF measure. Both algorithms were implemented and have been considered to be used in this thesis.

Another interesting approach in giving recommendations which are, besides user check-ins, additionally backed up using user *tips* left on places he has visited is introduced in [2]. Furthermore, besides this method, the authors noticed that both user social similarity, and inter-venue similarity influence user preference model and location recommendation performance. Matrix factorization approaches are used in this paper, which have been considered to be used as one of the approaches in this thesis. Even though matrix factorization has not been used, this work proposes the usage of additional sources along with check-in data which might be a topic of the future work after this research has been done.

Finally, the authors of [6] write about the location-based recommendation system application in a real-world scenario. The main topics of their paper are recommendation system architecture and real-case application of the system in the form of restaurant recommendation based on the Foursquare data. The paper is particularly interesting as it covers a concept that can be built on top of the concepts studied in this thesis.

Chapter 4

Data

The analysis of the given data was conducted in parallel with related work research, in order to get a better insight in data and an idea on how can it be used in modeling approaches that have been undertaken in later stages of the thesis.
First, the dataset and its format are presented and described in section 4.1. Section 4.2 contains a step-by-step description of data manipulations done from the point of downloading an original dataset, to the point of reaching the final data format which is then used in experiments. Finally, section 4.3 describes the analysis performed on the data and data visualization. Furthermore, conclusions made from data analysis are stated in section 4.4, as well as undertaken steps made with regards to them.

4.1 Original Data

The original data was downloaded in CSV format from Kaggle repository [5] "FourSquare - NYC and Tokyo Check-ins". The repository offers an option to download two datasets, first of them containing 573,703 users' check-ins in Tokyo, and second 227,428 check-ins in New York, both filtered form invalid data entries and within a period from 4th April 2012. until 16th February 2013. The decision was made to use Tokyo dataset given the greater amount of data. Initially, dataset contained 573703 check-ins made by 2293 users at 61858 different venues which are categorized within 384 categories.
Table 4.1 shows columns from the original dataset providing a short description of each.

Each of the dataset columns describe check-in that has been made by a user with a certain *userId*, on a venue defined by *venueId* which belongs to a category described by *venueCategoryId* and *venueCategory*. The venue's geographical coordinates are described by *latitude* and *longitude* float values. *TimeZoneOffset* and *utc-*

Column name	Description
userId	Anonymized user identification number
venueId	Venue identification string
venueCategoryId	Venue category identification string. Each venue corresponds to a category defined by Foursquare. [3]
venueCategory	Venue category name. Each venue corresponds to a category defined by Foursquare. [3]
latitude	The latitude of the pickup location
longitude	The longitude of the check-in location
timezoneOffset	Timezone offset in minutes (The offset in minutes between when this check-in occurred and the same time in UTC)
utcTimestamp	UTC time format

Table 4.1: Original data format

Timestamp are columns which together determine the timestamp at witch check-in has been made.

Foursquare venue categories are hierarchically organized in a tree structure having 10 nodes in the top level of the hierarchy. From these top-level nodes, finer categories are derived. Users can check-in in at any venue category from these levels even though the category might not be a leaf category. An example of the leaf category and description of the Foursquare's hierarchical venue category structure is shown in image 4.1. In this image "Planetarium" is a leaf category as no categories are derived from it. This means that this category does not have any categories bellow itself in a hierarchy.
It is very important to emphasize that the predictions which are going to be made take into consideration only the venue categories which were seen in the training dataset (training dataset is described later in this chapter).

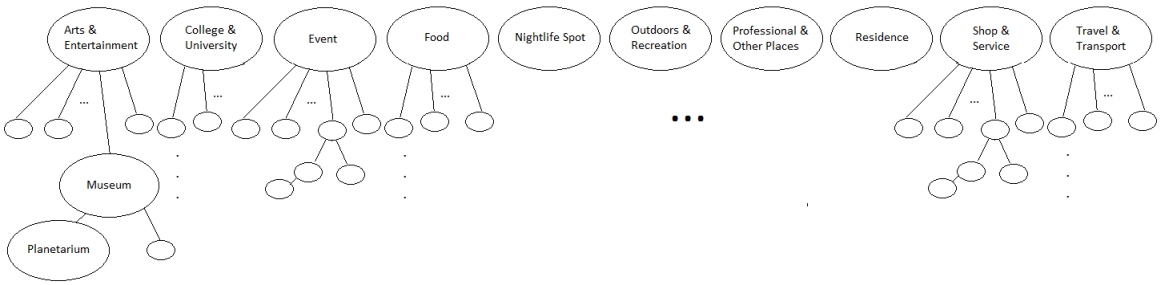


Figure 4.1: Foursquare hierarchy of venue categories

4.2 Preprocessing

Although original data is already in a format where it can be used, Machine Learning problems require great computational power and therefore, the efficiency of operations being made on data. Led by this thought, CSV file containing the dataset was imported in the PostgreSQL database management system in order to improve the efficiency of fetching the data which was the first step of data preprocessing.

After importing CSV file to the postgreSQL database manipulations on data have been done in order to improve storage and fetching efficiency and effectiveness. The first decision that led to a memory improvement was to combine timezoneOffset and utcTimestamp columns into one column. Prior to doing this, 27 entries were removed from the dataset as they were made in different time zones from the Japan standard timezone (JST) and were therefore considered irrelevant. After that, 9 hours were added to each original UTC timestamp in order to create a single, timestamp column. With these changes being made table *check_ins* has been created. This table, along with table *categories* where categories were extracted from Foursquare API, was a starting table in the database.

In order to improve the speed of data fetching, the second decision was to split the original check-in database table into four tables as shown on 4.2.

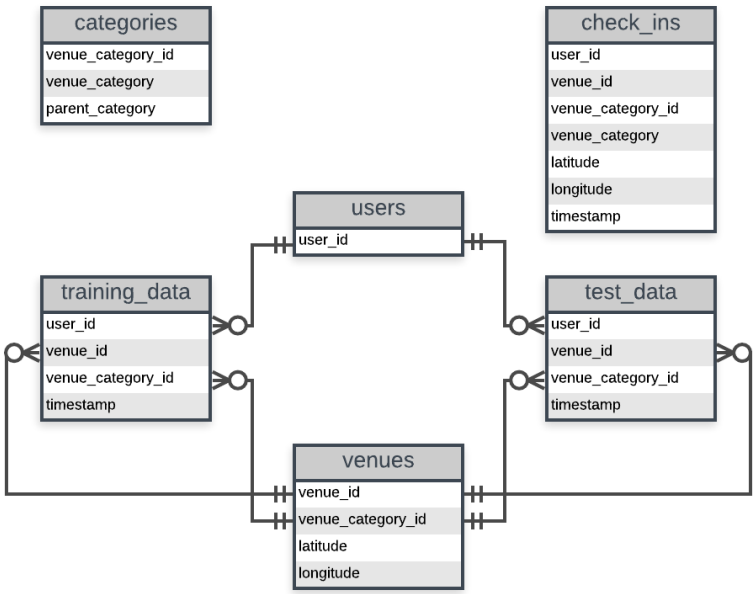


Figure 4.2: Table relations within PostgreSQL database

User table has only one column which represents an identification number of the user. *Venues* table represents the venue by its identification number, corresponding category identification number, longitude, and latitude. It is also important to mention that in this step *venue_category_name* column has been dropped. This is due to the fact that inconsistencies were noticed in mapping between *venue_category_id* and *venue_category_name* columns.
Relations described in [3] will be used to map identification number to the corresponding name.
Training_data and *test_data* origin is explained in the following text. For now, it is enough to say that they both have the same structure, having columns *user_id*, *venue_id*, *venue_category_id*, and *timestamp*. which represent user checking it at a venue of certain category at a certain time.

Check-ins as relationships between a user and a venue can be represented using a bipartite graph, where one domain represents venues and the other one users. The bipartite graph structure is shown in the following image.

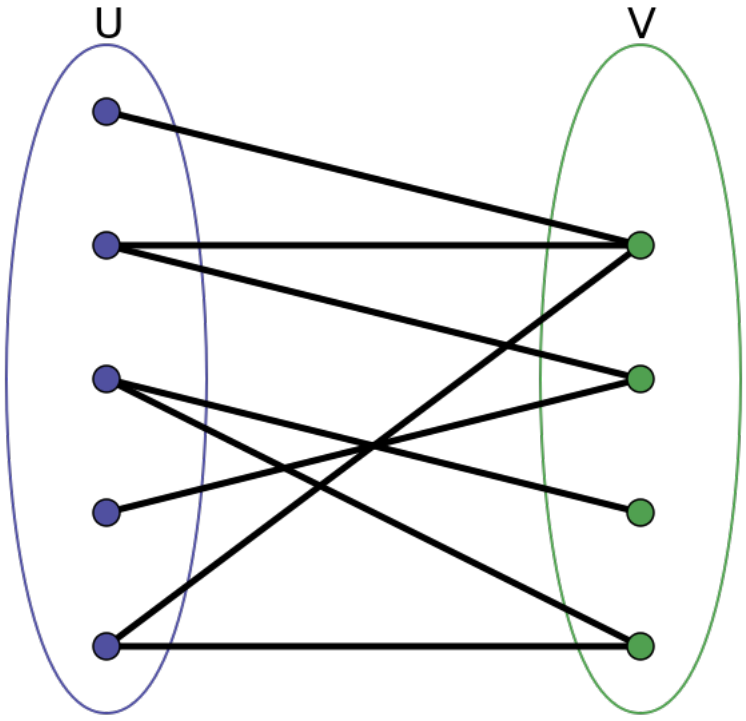
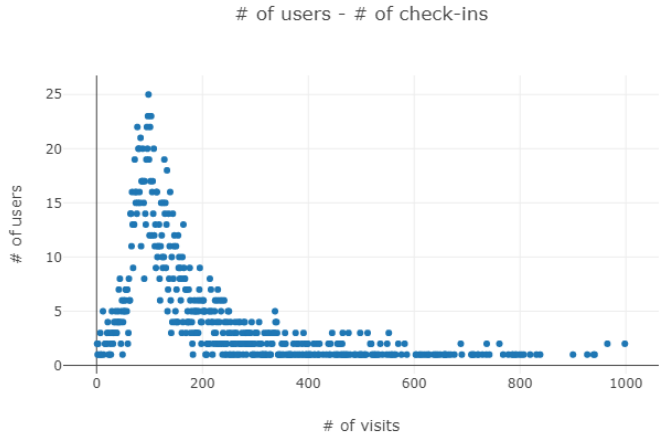
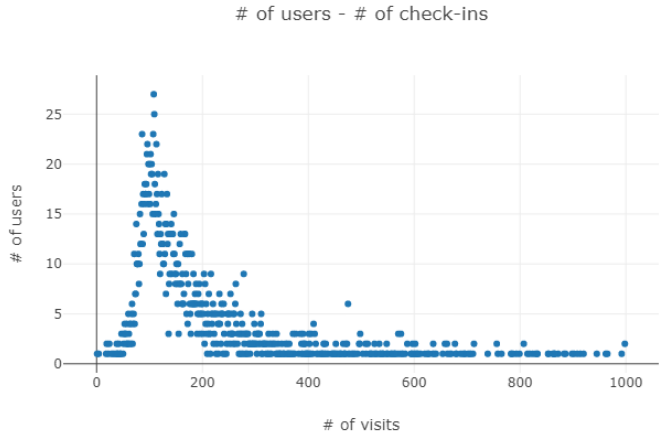


Figure 4.3: Bipartite graph

The next step of data analysis and preprocessing was to determine the dataset partitioning into the training and test dataset. Two options were taken into consideration, 70-30 and 80-20 split. 70-30 (shown on FIGURE 4.4a) split would result in splitting data into first dataset from 4th April 2012. until 15th November 2012 and second with the rest of the data, while 80-20 (shown on FIGURE 4.4b) splits data in a way where the first dataset contains check-ins from 4th April 2012. until 15th December 2012. and the second which contains the last two months of check-ins.



(a) 7 month check-in training dataset



(b) 8 month check-in training dataset

Figure 4.4: Plots show how many users have checked-themselves in x number of times

FIGURE 4.4b shows a training dataset statistic of how many users have checked-

themselves in a certain number of times. This plot highlights which users should be considered as relevant in a prediction problem as it shows the distribution of activity among the users in the network.
FIGURE 4.4b shows more active users as the considered period of check-ins is one month longer. One of the things it shows is that only two users have made less than 5 check-ins while that number is fairly larger on the other plot. User which has the minimal highest number of check-ins amongst the other users than these two has 18 check-ins.

The second split having more active users gave a reason to believe that models will be of higher accuracy if trained on the larger dataset. Also, if the first split had been used it would result in removing a great number of users due to their inactivity. Besides this, a lot of relevant projects to this one in the current state of the art are using 80-20 split of the dataset.
With this being said, the decision was made that 8-month training data will be used, and users who appear in training data and have at least 18 check-ins will be considered. Eighteen check-ins for an app like Foursquare could already provide some insight into users' preferences and it might be enough to predict those users' future activities. These actions are habitual and they are rarely changed therefore having a high chance to continue happening in the same way, unless a person goes through a bigger change in life.
At this point, it is important to note that from original 2293 users, 9 of them were discarded due to not appearing in training data, while 2 of them were discarded due to insufficient application usage up to the point where the dataset is split into training and test dataset.

Besides users, two venues have been removed after removing the category name column and mapping the original (venue_category_id, venue_category) pairs. This was done by pulling the JSON file, from the Foursquare API [4], which contains a hierarchy of venue categories with corresponding data. Two venue_category_ids were found to be invalid, while a lot of entries had an invalid and inconsistent id -> name mapping.
After venue filtering, 56139 venues are considered as possible check-in venues.

After filtering venues and users, one more thing was noticed and had to be filtered. It was noticed that some check-ins made by certain users have been logged multiple times at once (having the same timestamp).
After closer examination, it was discovered that some of those check-ins were done on purpose, having the same venue_id but different categories. For example, a venue with venue_id '4b9963aef964a520d07735e3' was categorized in three different categories, namely Japanese restaurant, Dinner, and Fast Food restaurant.

On the other hand, some check-ins were simply logged exactly the same, by the same user, multiple times and having the same timestamp. This was most likely due to the bug in the system, so those check-ins were made distinct (only one check-in was logged). The first mentioned anomaly might have an influence on the model accuracy results as the same inputs are mapped into different classes. However, there is just a small portion of these check-ins so they might not influence the prediction results significantly.

After filtering check-ins in this way, the final dataset contains 571181 check-ins, of which 467494 contained in the training data, and 103687 contained in the test data.

4.3 Data Visualization

This section focuses on data analysis and visualization, which will help to improve the understanding of the data and capture eventual patterns and anomalies that might influence location predictions.

The presented figures have been generated using Matplotlib and Plotly.

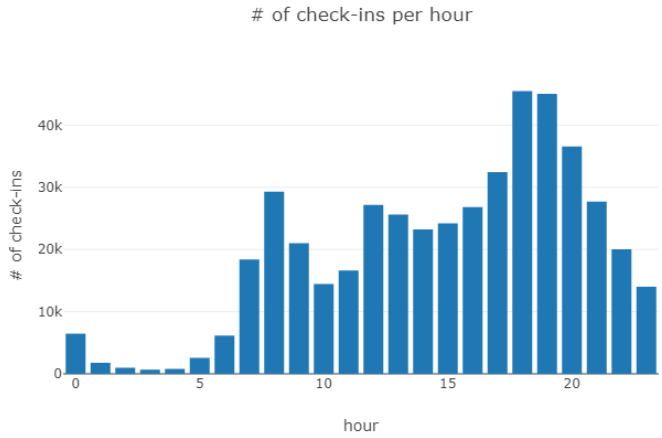
Data visualizations made during this stage of the project were made merely to get an understanding of which features might carry some data information. A list of anticipated decision segments that might be used as features in the learning process of the models is enlisted in the next chapter. This thesis focuses on three types of segments namely social, temporal and spatial.

Since there were no relations between users, along with this dataset, which usually exists in social networks, the social segment had to be inferred in the later stages of the project. The inferred social relationships were not visualized in this chapter as they were made in multidimensional vector space. Also, these are not representations of relationships between users, but rather representations of relationships between users' preferences.

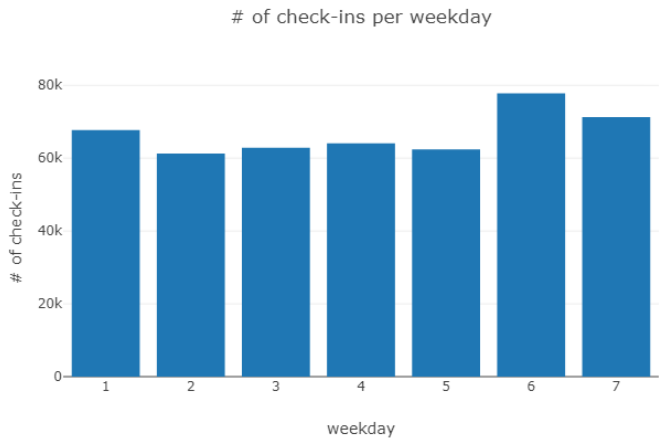
The temporal segment was always highlighted as important in most of the related work mentioned in CHAPTER 3, so it was natural not to leave it out of this research. The point of getting insights from the temporal segment was to look into possible check-in patterns throughout different periods of time. If any kind of check-in pattern is detected, it could then be used as a feature in decision models. Using those features would hopefully then result in capturing of the detected patterns along with capturing some more obvious patterns.

For example, people visit certain places at a certain time of the day. There is not much sense in visiting the school at 2 AM, nor visiting a cinema at 6 AM. Another example would be that people are more likely to engage in entertaining activities during the weekend as they would be occupied with their professional activities throughout weekdays. Besides this, some activities are seasonal like skiing or kite-surfing and so on. All of these patterns would need to be captured.
Following figures, aggregated from the entire training dataset, present some of the findings related to daily, weekly and monthly patterns concerning the number of check-ins:

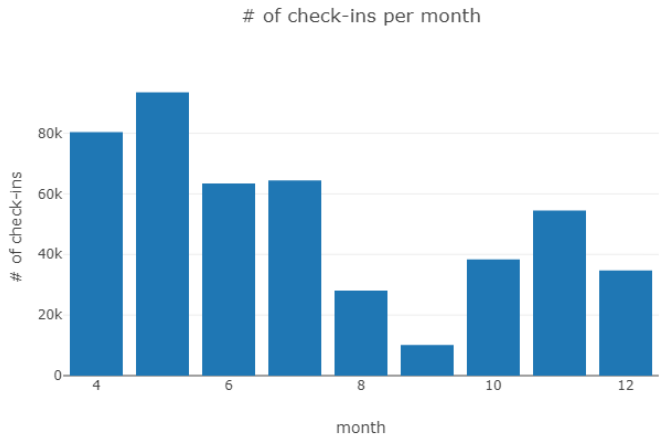
- Daily check-ins plot 4.5a
- Weekly check-ins plot 4.5b
- Monthly check-ins plot 4.5c



(a) Plot showing distribution of hourly accumulated check-ins



(b) Plot showing distribution of check-ins accumulated by the day of the week

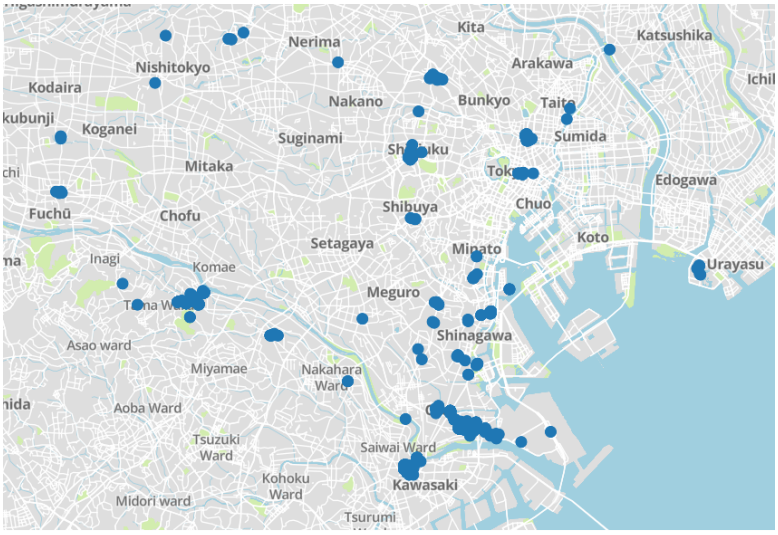


(c) Plot showing distribution of monthly accumulated check-ins

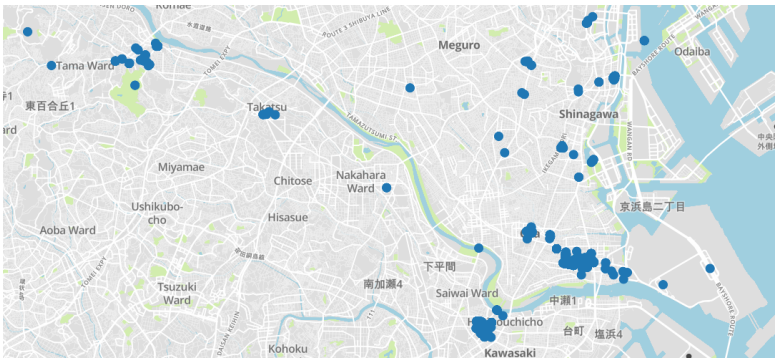
Figure 4.5: Check-in aggregations distribution based on different time units

The first image shows that there is a highly uneven activity distribution based on time of the day. The intuition behind this plot is that most users check themselves in between 7 AM and 11 PM as most people are awake at that time. Furthermore, the highest numbers of check-ins are being made between 5 PM and 9 PM which again shows the validity of the data as most people are done with their responsibilities for that day and can engage themselves in leisure activities. Based on this plot and mentioned explanations, time of the day will be considered as one of the features for decision models described in the next chapter.
Plot on FIGURE 4.5b shows an almost uniform distribution of the check-ins throughout the days of the week. Slightly increased activity can be seen during the weekend. In this chapter, there was an already mentioned example of the possibility that people will most likely do different activities during weekends when compared to weekdays and that the application usage activity might be higher. The most important information that this plot shows is a difference of around 15 thousand check-ins during Saturday compared to the other days. This information encourages the consideration of possibly using weekday as one of the features for the decision models.
The lasts of three plots represents the distribution of the check-ins aggregated based on the month in which check-in was made. The plot shows high differences in the number of check-ins between the months with most check-ins being made in May and the least in September. From this plot, there were no conclusions made even though there was an example given in this chapter which describes the importance of the month of the year for doing certain seasonal activities. No actions were taken based on this plot as the used dataset is using check-ins in a period which is less than a year. Therefore, capturing a seasonality pattern was considered unnecessary. This plot and example shows that some patterns, which were tried to be detected and were considered important based on the intuition, were discarded and not considered further in the modeling of the decision process.

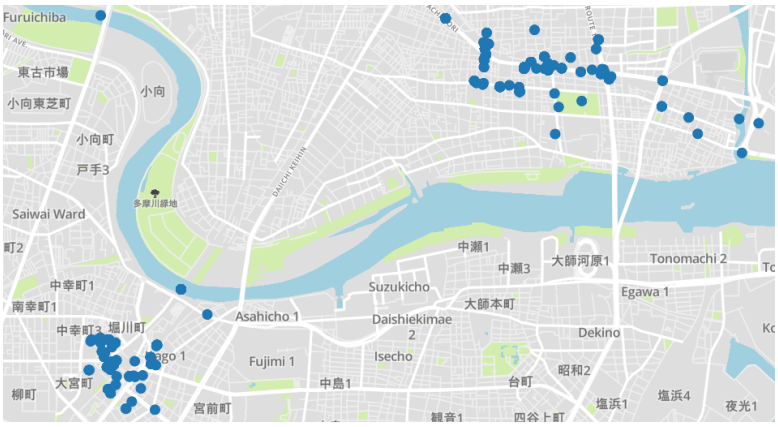
The spatial decision segment was mentioned as important in a large number of related works solving similar problems to this one. This research did not ignore this segment either, as people tend to be tied to certain areas within the city doing the same activities each time they go there. A good example of this is a person's workplace which located further from her residence area.
Usually, people tend to go to lunch during their workday, so that person can tag herself in a restaurant, cafe bar or fast food near her office. This can be seen as a series of similar activities done within a small spatial region. These activities should also be captured as they give some information about a person's preferences. Following images show check-ins of one of the users on different spatial scales:



(a) Whole town



(b) Several clusters of check-ins



(c) Two check-in clusters near Kawasaki and Ota

Figure 4.6: Plots showing check-ins from a training dataset of one of the users

The first image shows all of the check-ins made by a random user from the dataset. Image 4.6b then shows more zoomed invariant to one of the areas of the town. Both images clearly show that the user is not making check-ins on random areas in the town, but rather area clusters that have been visited multiple times as it can be seen from the images. These check-ins can be explained as serial activities which have been mentioned above in the text.
The last image shows two of those areas which represent clusters, namely Kawasaki and Ota. These plots show the distribution of check-ins within the clusters. An interesting detail on this image is that plotted check-ins are not in the water which might be an indicator of accurately read GPS data. Rather, there are several check-ins made on the bridges and on a pier.
These findings show that there might be patterns to extract from the spatial segment of the check-in data as users check themselves in only in certain areas of the town, area of their interest. Led by this reasoning latitude and longitude column in the dataset will be considered as features when testing the decision models.

It is important to note that spatial and temporal information will be available when the user has mobile location services turned on on her device. This way, a recommendation of an activity that is relevant to the activity detected by a user decision model can be recommended to the user. Led by this, it is quite intuitive to think of spatial and temporal information as crucial to this research.

The last visualization was made based on the assumption that some categories will be more popular than the others, in a sense that those categories are essential to the peoples' everyday life. Such categories are related to public transport, stores, and so on. This is clearly shown in figure 4.7 which represents a plot sorted by the number of users who visited a certain category.

It is important to note that these are the activities during which people will more likely be on their phones using the LBSN application. Besides that, it was assumed that the same, or similar categories, will be frequently visited by all of the users because of the same reason as mentioned above.
This is backed up by the figure 4.8, which represents the descend sorted plot, showing the number of users *favouring* a certain category. *Favouring* the category means that most of their check-ins are categorized as that category.

Both plots and conclusions above will be taken into consideration when modeling the decision process so that plotted trends do not create too much of a prediction bias due to the popularity of the presented categories.

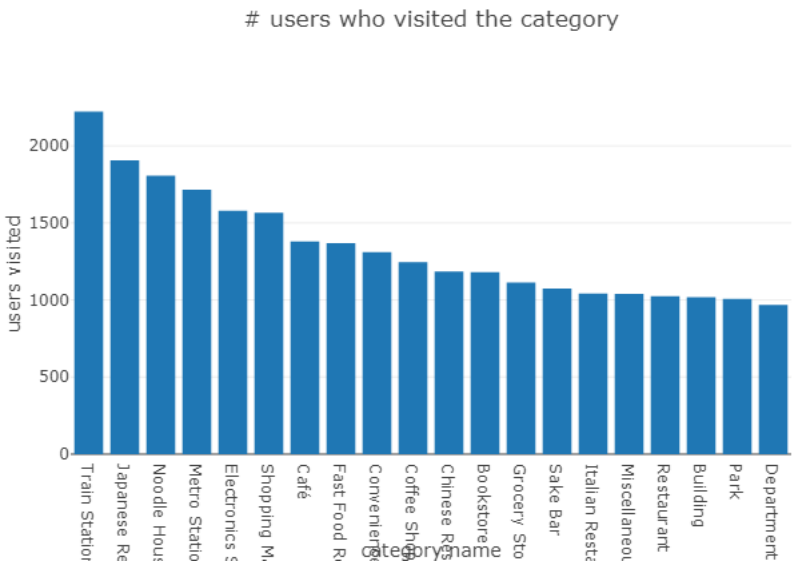


Figure 4.7: Plot showing the number of users who visited a certain category

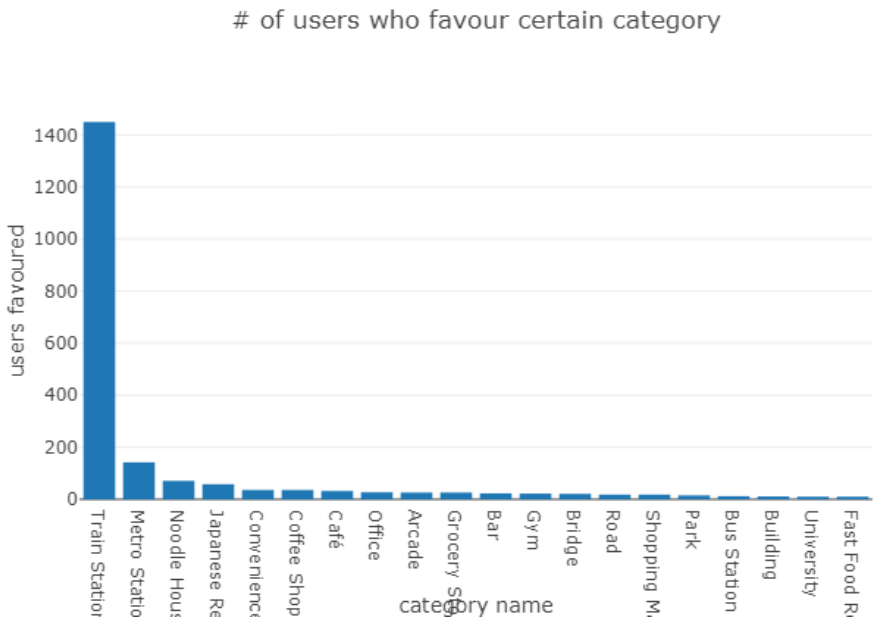


Figure 4.8: Plot showing number of users who favour a certain category

4.4 Summary

This chapter starts by explaining the dataset and its origin. It informs on data properties and format in which it is acquired. Then a preprocessing of the data is explained from the original CSV format to the final and more efficient postgreSQL format. After this, data has been partitioned for the purpose of the machine learning problem of this research and based on that adequately cleaned from the invalid entries.
The chapter then focuses on finding the essential information from a provided dataset which can help to make human decision modeling more accurate. Three decision segments were taken into consideration while two of them were analyzed. It has been shown that spatial and temporal segments carry essential information for the purpose of this research and therefore, these will be used in the modeling of a decision process.

Chapter 5

Modeling Approaches

Human decisions are influenced by a combination of multiple inputs. Those inputs are mostly imposed socially and by nature while taking into account the knowledge and personal experience of the person who decides. Some of the anticipated inputs are enlisted.

- Time of the day
- Weekday
- Type of the day (working day/weekend)
- Time of the year (holidays, winter/summer off-work period, school-break)
- Weather
- Special circumstances (heavy traffic, injury, ...)
- Person's interests and responsibilities
- Social influence
- Internet and social network activities
- Area around frequently visited venues
- Proximity of venues to the user's frequently visited venues
- Holidays and special days

This is of course not an exhaustive list of inputs which influence a person's decision, as more thorough psychological and similar researches are needed in order to discover these, but for the purpose of the thesis and with regards to available

data, these inputs are considered as relevant.

Time of the day is probably the most important segment of all mentioned above, as it greatly influences the person's decision on which action is going to be undertaken. The reason for that lies in the fact that most working people are at their job venue from 8 AM until 4 PM on working days, and students are at the schools/universities at a similar time. The rest of the day usually falls to doing other than mentioned, periodic activities, which usually repeat on a weekly basis. For students, some examples would be extracurricular activities such as sports, foreign languages and similar. Intuitively, these patterns are expected to be broken during the weekend when people are not working or going to the school, which makes them more flexible with deciding on which activity to do that day and where to do it. Time and type of the day are definitely two segments which will need to be captured in order to do a precise user preference modeling.

Time of the year/seasonality would be an important segment to consider if a longer period of check-ins was available. Since the span of the occurred check-ins in the available dataset is within ten and a half months, this segment is not considered. Weather is another segment which might greatly influence a person's decision, especially in the case of extreme weather conditions. However, modeling this segment would require additional historical weather data which would require more research and development time than what was predicted for the writing of this thesis. Therefore, this segment is not considered as relevant for the person's decision.

Similarly to the weather, special circumstances can also be taken into consideration as one of the segments which influences people's decisions. Special circumstances in this context would mean an injury of a person, heavy traffic, or some kind of an emergency situation. Data on these, as mentioned in some sources from SECTION 3, can be acquired through the people's feedback in form of the *tips*, and traffic reports. The available dataset does not contain user's *tips*, and the usage of traffic reports is not used for the same reason as the weather data.

Person's interest and responsibilities **might** be modeled by capturing frequencies of the check-ins, as well as the pattern of their recurrences. Word "might" appears in the previous sentence because people are not checking themselves in every time they do an activity, and their check-in history is expected to be biased to a certain extent compared to their activities which happened in reality. Still, interests and responsibilities are expected to be captured by a higher number of check-ins to the venues of similar category.

Social influence is a segment captured in several sources from 3 section. It is believed that this segment is of high importance especially when a user decides on visiting venues that the user has not visited before. The description of capturing this segment is described in this chapter.

Another segment which influences a people's decision can be extracted from their internet activity, especially social activities. Again, this segment, like some previously mentioned, requires additional sources of data. The fact which is important to mention here is that Foursquare offers the functionality of connecting user to user, which is a good source of the information that can be exploited to capture a social influence on a person's decision. Unfortunately, this information source was not provided along with the available dataset. Social relationships of the users from this dataset are inferred from the preference similarities between users instead of having pre-made relationships that go along with the dataset.

The spatial segment is the last of the enlisted ones. Spatial information relevant for this topic are areas which a person frequently visits, as it is more likely that a person will prefer visiting the venues which are in proximity of the area user usually visits. The logic behind this is that for example, people will most likely visit a restaurant or a cafe bar which is near their workplace. This is the most important of enlisted segments next to the temporal segment and is going to be used as a part of a decision process.

Sections 5.1 and 5.2 describe the concept of two baseline algorithms which will be used as a benchmark for comparison to the others.
Other sections in this chapter cover more complex approaches used to model user's decisions.

5.1 Favourite Choice

Favorite choice model is implemented in a way that it always predicts a category which was visited the most by a user.
For example, if a user has checked-in at the metro station 120 times, 63 times at convenience store, and twice in cinema, every future check-in is expected to be at the metro station.

This model will show to what extent are people repeatedly doing their favorite activity.

5.2 Time Aggregated Favourite Choice

Time aggregated favorite choice model is an upgrade to the favorite choice model. It is implemented in a way that check-in frequencies of each category are measured based on the hour of a day. This is then expanded in a way that if there were no check-ins logged at a certain hour, an hour before and after the one for which the category is being predicted are considered. If there is no log for any of these time periods, the category is predicted based on the Favourite choice model.

The second variant of this model is also implemented, with the difference that it takes both hour of the day and day of the week into consideration.

This model is basically a look-up table having the format as depicted in TABLE 5.1.

Hour	0	1	2	3
Category	Category1	Category2	Category3	Category4

Table 5.1: Time Aggregated Favourite Choice model of some user

In this example, for a user based on whose check-in history this table is made, Category2 will be predicted for every time it is 1 AM.
The idea behind the implementation of this model is to see whether the temporal segment is as important as it was assumed at the beginning of the thesis, and if yes to what extent. This model can answer that assumption as it is purely based on the temporal decision segment (hour of the day/hour of the day and weekday).

5.3 Multilayer Perceptron

One of the algorithms that are usually used in pattern recognition problems, and was proposed by various sources is the Multilayer perceptron classifier. This section describes a perceptron, which is a building unit of a multilayer perceptron. After that, the concept of a multilayer perceptron is introduced.

Perceptron

The perceptron is the simplest neural network as it is a single neuron model. It has only one layer and it contains one or more inputs, a neuron, and an output. Perceptrons are used in supervised learning for solving classification problems.

FIGURE 5.1 shows the structure of a perceptron. Inputs are multiplied with corresponding weights, and then the multiplications are summed. That sum is then

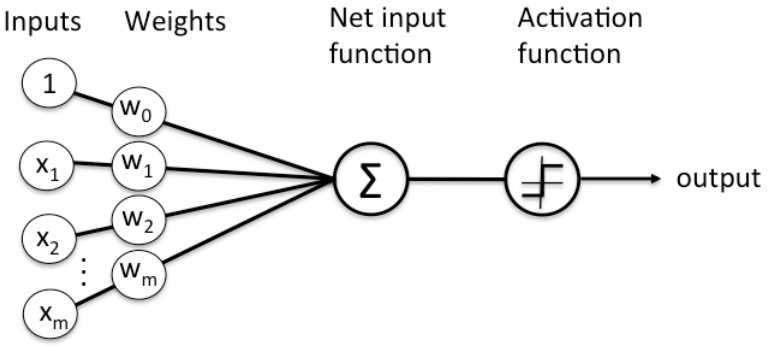


Figure 5.1: Perceptron [13]

propagated to the activation function, based on which output is calculated. There are different activation functions which are used in this concept. Often used activation functions are shown on FIGURE 5.2.

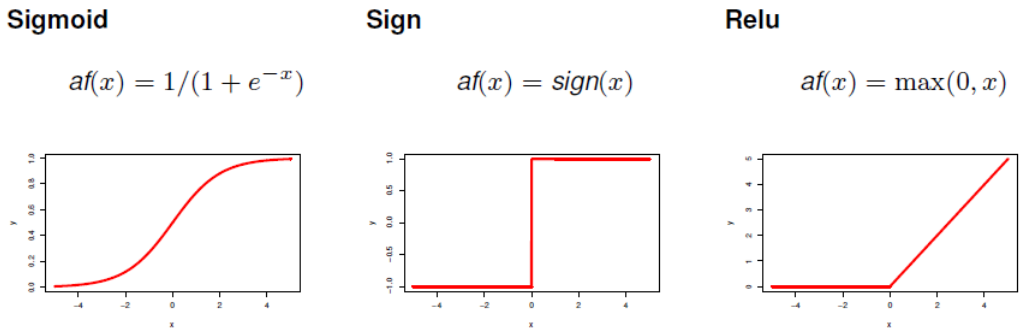


Figure 5.2: Activation functions [11]

Mutilayer Perceptron

A limitation of a single perceptron is that it is only able to make classifications which are linearly separable. In order to increase the expressiveness, the concept of multilayer perceptron is introduced. A multilayered perceptron (MLP) is an artificial neural network having at least three layers. These layers include the input layer, one or more hidden layers, and an output layer, all consisted of perceptrons and connections between them as shown on FIGURE 5.3.

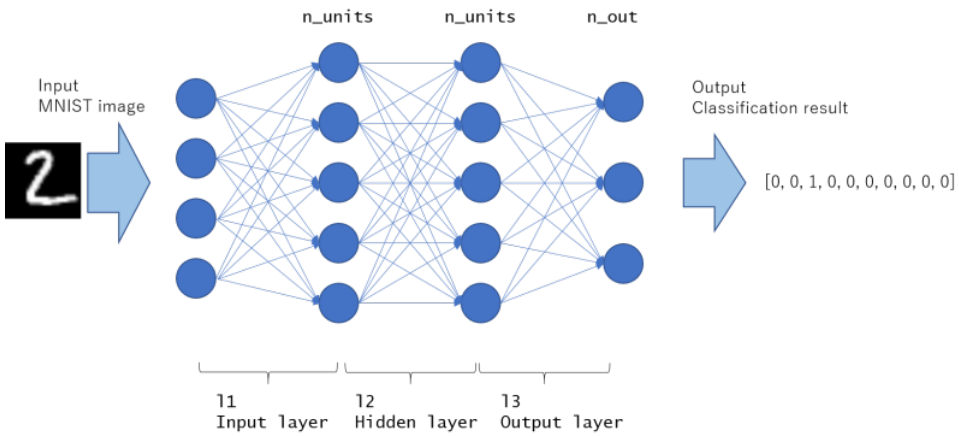


Figure 5.3: Multilayer perceptron [10]

The purpose of training the multilayer perceptron is to, by "feeding" it with training data, find the weights within its network such that they give the most precise output results when "fed" with training data. In other words, a certain combination of weights is going to be a product of the process of learning, which for the classification task of the data which has been input in the neural network will be considered as optimal. That means that not every entry from the training data will be classified in the right way given this combination of weights. But an optimal amount of the training data will be classified with the response to their true class. The aforementioned *optimal* amount is determined by an error function used to optimize the weights. Scikit-learn MLP classifier implementation, which is being used in this thesis, uses Cross-Entropy as an error function.

The process of learning is described as follows.
The input signal propagates through the network layer-by-layer generating an output value which is then compared to the expected value from the training data. This step of the learning process is called *forward propagation*. After that step, *backward propagation* updates and optimizes the values of the weights. It minimizes the error by finding the derivative of the error relative to each weight. This value is then subtracted from the weight value. These steps are done until the maximal iteration threshold is reached or convergence of error function is ensured.
When giving a prediction, output of the neural network, in the form of a vector which length corresponds to the number of classes which have been seen in the training dataset, is passed to the softmax function 5.1[16] where z_i represents the i-the element of the input to softmax, which corresponds to class i, and K is the number of classes. The result is a vector containing the probabilities that sample x belongs to each class. The output of the prediction (softmax function) is the class with the highest probability.

$$softmax(z)_i = \frac{exp(z)_i}{\sum_{l=1}^{k} exp(z_l)} \tag{5.1}$$

Using multilayer perceptron more accurate predictions are expected than predictions baseline models are giving. Due to the complexity of the multilayer perceptron, it is expected to extract activity patterns that can not be captured by a lookup table. In order to do that, multilayer perceptron will also take into consideration other segments than temporal, while still being implemented and tested for only using temporal segment features.

5.4 Term Frequency – Inverse Document Frequency Measure

Due to the variety of different users' preferences and variety in nature of places that users are visiting (as shown on FIGURE 4.7 and FIGURE 4.8), TF-IDF measure is selected to be one of the components to model a user preference.

Term frequency-inverse document frequency measure often referred to as TF-IDF, is often used in information retrieval and text mining. This approach is based on calculating TF-IDF weights in order to determine, in the context of text mining, how important a *term* is to a document in a collection or corpus. The importance increases logarithmically given a number of times a term appears in the document while taking into consideration the frequency of the word in the corpus. Measure which describes the aforementioned properties and that is used in this thesis has the following structure:

$$tf^* - idf_{t,d} = (1 + \log tf_{t,d}) \cdot \log N/df_t \tag{5.2}$$

with a constraint that if $tf_{t,d}$=0, expression $\log \mathrm{tf}_{t,d}$ equals zero. In this equation t and d represent term and document respectively. N represents number of documents in a corpus and $tf_{t,d}$ represents a number of terms t appears in a document d, while df_t represents document frequency (a number of documents in which term t appears).

The same importance measure is suitable for this thesis' problem. In this problem, a user's history of check-ins is considered as a document, and venue categories are regarded as terms in the document. These considerations are justified as an activity is important to the user if it has been frequently done. Furthermore, some categories are generally visited more by the users and do not give us any

specific preference meaning, such as train station, parks, roads and similar. While on the other hand, some categories carry more meaning in terms of user's preference, like school for example. If a user visits a school on a daily basis, we can assume that he is either a school employee or a student, while we can not make any particular conclusions about the user if he checks himself somewhere on the road.
Given these arguments, it is sensible to claim that inverse document frequency can describe the specificity relation of the category, while term frequency can describe a user's fondness to a certain category.
This then transforms term to a category and document to a user or equation 5.2 to the equation:

$$cf^* - iuf_{c,u} = (1 + \log cf_{c,u}) \cdot \log N/uf_c \tag{5.3}$$

User's preference weight ($u.w_c$) is calculated as shown in equation 5.4. The equation has two parts, which denote the TF value of venue category c in user u's location history, and the IDF value of the category, respectively. $|\{u.c_i : c_i = c\}|$ represents user's number of check-ins to venue category c, , and $|\{u_j : c \in u_j.C\}|$ regards number of users who have visited the category c out of total users U.

$$u.w_c = (1 + \log |\{u.c_i : c_i = c\}|) \cdot \log \frac{|U|}{|\{u_j : c \in u_j.C\}|} \tag{5.4}$$

TF-IDF scoring will be used in order to represent a user as a multidimensional vector. Representing users this way will allow user comparison through the vector cosine similarity. This comparison is of high importance as a social segment of the decision will be modeled by detecting similar users and taking their consideration when predicting someone's activity. TF-IDF measures can also be used as features based on which users can be classified using clustering algorithms.

5.4.1 Cosine Similarity

Cosine similarity is a measure that describes how similar two vectors are to each other. Users are represented as multidimensional vectors of activities whose dimension segments are TF-IDF scores. The goal is to calculate the angle between those TF-IDF vectors in order to measure the similarity between vectors/users. This can be done using the following formula for cosine similarity:

$$cos(a, b) = \frac{a \cdot b}{|a| \cdot |b|} = \frac{a}{|a|} \cdot \frac{b}{|b|} = \hat{a} \cdot \hat{b} \tag{5.5}$$

$$\hat{a} \cdot \hat{b} = \frac{\sum_{i=1}^{N} a_i \cdot b_i}{\sqrt{\sum_{i=1}^{N} a_i^2} \cdot \sqrt{\sum_{i=1}^{N} b_i^2}} = \sum_{i=1}^{N} \hat{a_i} \cdot \hat{b_i} = cos(\hat{a}, \hat{b}) \tag{5.6}$$

5.5 K-means

K-means is one of the most important cluster analysis algorithms. K-means clustering is a classification algorithm designed to group data into a predefined number (k) of clusters where each data must be a vector of numbers. The algorithm is described as follows:

Algorithm 1 K-means Algorithm

```
K <- Choose the number of clusters
D <- Obtain the data points
C <- Randomly generate K centroids
while convergence || n_iter > iteration_treshold do
    for data_point : Data do
        Find the nearest centroid c
        Add data_point to the cluster defined by centroid c
        for centroid : C do
            Move centroid to the mean of all associated data points
```

As the algorithm progresses, centroids are changing their location until the algorithm converges or hits the iteration threshold. The algorithm will always halt, but it will not guarantee the most optimal result possible.

K-means will be used as a part of the preprocessing stage in order to classify similar users in the same class. This will be done by representing users as multidimensional vectors whose structure is given by TF-IDF scores as described earlier in the chapter.

The motive for doing this clustering is to create new features for a neural network which will represent a user's affiliation to a certain cluster. This way, the neural network is expected to find a pattern where similar users are doing similar activities.

Chapter 6

Problem Approach

This chapter describes the undertaken approach to the solution of user activity preference modeling. First SECTION 6.1 addresses the goal of this thesis and the modeling plan. Next, SECTION 6.2 describes the environment setup and used technologies. Then, the following SECTION 6.3 describes the implementation of the used approaches and features and ends with the description of decision models. Lastly, this chapter ends with the summary 6.4 which shortly lists models and their properties.

6.1 Process description

The goal of the thesis is to capture people's activity interest as accurately as possible. The activity interests of people are represented by their check-in to a venue of a certain category. The checked-in category is a very good indicator of what is a person's interest at the moment, given the assumption that a person will not make check-in for no reason, but that each check-in made is a representation of person sharing their current experience with their network.

In order to capture people's interest, this thesis is focusing on what might be the influential factors of a person's decision and how to model those as accurately as possible. Then, in the introduction of the CHAPTER 5 the segments of a person's decision are enlisted. The list was created based on the context of this research while keeping in mind subjectivity of individual reasoning.

Firstly, the dataset from social network Foursquare is acquired and appropriately processed. That processed data was then studied in order to find valuable insights. Decisions on which segments will be considered relevant were made during the process of planning, based on the data insights acquired.

As a result of planning, it was decided that two simple baseline algorithms will be implemented. The second algorithm is an expansion of the first algorithm, and therefore it is expected to perform better.
Apart from simple algorithms, a more complex multilayer perceptron classifier will be implemented. This model is expected to perform better than baseline algorithms, due to its more complex structure and ability to extract patterns out of parameters which are considered to be important.
Decision segments in the focus of the thesis, from the enlisted ones at the beginning of CHAPTER 5, are:

- Time of the day
- Weekday
- Person's interests and responsibilities
- Social influence
- Area around frequently visited venues
- Proximity of venues to the user's frequently visited venues

Different approaches, described in the SECTION 6.3 will be used to model these segments creating multiple human activity preference models.

6.2 Environment setup

This section describes the processes and technologies used in order to tackle the problem of this thesis.
First, raw data had to be formatted and stored properly in order to be used in a more efficient way and filter the unnecessary information. After this, data was ready to be efficiently loaded and used in model training. In the end, trained models gave an output for the test data which was then compared to the true values. The comparison between prediction and real values is conducted and the results are evaluated. FIGURE 6.1 captures all of the above mentioned processes.

Given the Machine Learning nature of this problem, Python was decided to be used as a programming language due to its rich scikit-learn machine learning library and its ease of use in the development. Besides having good machine learning libraries, Python has also great visualization libraries which were used to represent data and the results.
PostgreSQL was used as a database management system due to the wide community of users and its documentation quality.

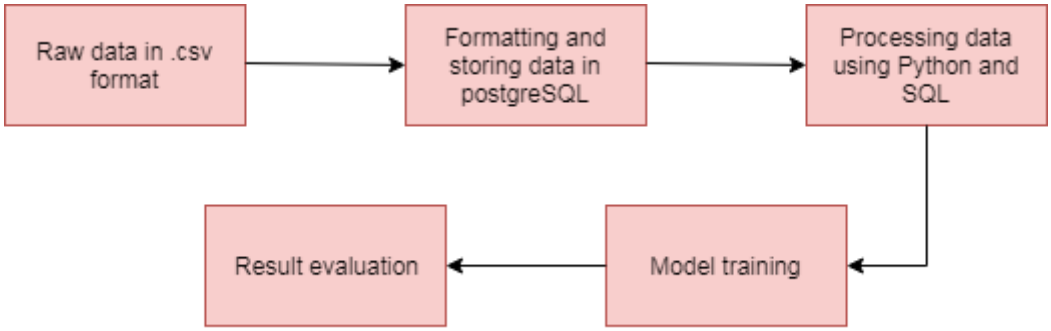


Figure 6.1: Dataflow

6.3 Models

This section describes detailed use of models described in CHAPTER 5. All models are enlisted and described, besides two baseline ones which are considered straightforward and do not require any additional description. However, at least a brief reflection on each of the models will be written at CHAPTER 7.

Multilayer Perceptron Classifier

The previously described multilayer perceptron is used in the context of this thesis as a Multilayer perceptron classifier (MLP) implemented using a scikit-learn Python package. Multiple approaches were undertaken when evaluating the quality of this classifier.

One approach to solve this problem is to create a separate MLP for every user individually. In the other approach, rather than having one MLP for every user, only one MLP is created to predict activities for every user.
Both of these initially distinct approaches were tested with multiple combinations of features and MLP settings.

Based on the findings from CHAPTER 4, day of the week and hour within a day are selected as features that will be used in the user activity preference detection using MLP classifier.
First, the day of the week feature was given a value from a domain DoW = {1..7}. At the same time hour of day feature elements were taking a value from a domain HoD = {0..23}.

The following combinations contain encoded features that were a part of the first combination described above. These encodings are considered because the type of their encoding carries a property of a feature that needs to be captured by

a neural network. The desired goal of feature encoding is to increase the accuracy of a model.
In the second combination, the day of the week feature was encoded in the same way, but an hour of a day feature was encoded as a combination of two new features. Encoding of the hour of the day feature in two dimensions was made in order to ensure the cyclicality using sine and cosine functions. Both sine and cosine function are needed in order to secure that time is not ambiguous. Using only one of the functions would result in mapping different time of the day to the same value due to the periodic nature of those functions.
Example of encoding for some times of the day is shown on the following table.

Time	**Sine encoding**	**Cosine encoding**
5 AM	0.96593	0.25882
7 PM	-0.96593	0.25882
10 AM	0.5	-0.86603
2 PM	-0.5	-0.86603
7:07 AM	0.95757	-0.28820

Table 6.1: Times of the day using sine and cosine encoding

In order to do these transformations, hour and minute were extracted from the timestamp. After that, a time of day is expressed in minutes and transformed into cosine and sine value. This process is mathematically formulated in the following equations where m stands for a time of the day expressed in minutes and M being a day length expressed in minutes.

$$sin\_tod = \sin(2 \cdot \pi \cdot \frac{m}{M}) \tag{6.1}$$

$$cos\_tod = \cos(2 \cdot \pi \cdot \frac{m}{M}) \tag{6.2}$$

By doing this, the cyclical nature of the time of the day feature has been ensured. For example, 0:01 AM and 11:59 PM will not be considered as 2 minutes apart unless this transformation is made.
After this transformation, both newly created features are mapped to the values from domain CS = [-1..1].

In the third combination, the time of the day feature is encoded in the same way as in the second combination, and the day of the week is encoded using one hot encoding. Instead of a day of the week feature, now there are 7 features each encoded with 1 or 0, depending on which day was check-in made on. For example,

if check-in was made on Wednesday, a vector of features would look like this: (0, 0, 1, 0, 0, 0, 0).

Up until here, all features that were used were temporal. As noted before, the spatial segment is very important in this problem, so latitude and longitude were taken into consideration as additional features which are expected to improve the neural network accuracy.
This approach creates a need for a small adjustment in the test dataset. The reason for this need is that it does not make a lot of sense to predict the activity of the user while feeding a neural network with the exact coordinates of a venue where the user will do a certain activity. This does not make any practical sense, and would most likely lead to neural network learning venues' coordinates and predicting the activities accordingly. In order to avoid this, a small bias is added to latitude and longitude features in the test dataset. This bias will transform the venue coordinates into coordinates somewhere within 700 meters of the place the user has visited. Creating this bias creates a real case situation, where based on the current time and a user's location (transformed latitude and longitude coordinates), model is trying to determine what is the user's next action.

Scikit-learn documentation notes that many models usually behave badly if a requirement of data standardization is not fulfilled. Therefore, longitude and latitude features were standardized by removing their means and scaling them to their unit variances.

With this step, temporal and spatial decision segments are captured by selecting corresponding features for each segment. These features are shown in a TABLE 6.2.

Feature	**Description**
sin_tod	Minute of the day, encoded using sine function
cos_tod	Minute of the day, encoded using cosine function
dow	Day of the week, encoded using one hot encoding. This is not actually feature for itself, but rather a set of seven features representing the days of the week. One of these features has value 1 and the rest 0, depending on what day of the week check-in was logged on.
lat	Standardized latitude of the venue's location.
lon	Standardized longitude of the venue's location.

Table 6.2: MLP features defined by Spatial and Temporal segments

After capturing the spatial and temporal segments, the social segment was the only left. The idea of capturing the social segment is that people with similar preferences might influence each other's actions. Usually, LBSN datasets have user relationships defined through which a social segment can be inferred. However, the acquired dataset did not have these, so the social segment needed to be inferred differently. In the solution of this problem, the decision was made that people will be grouped according to their interest in certain activities.
Having these constraints, opinions of the users similar to the user for whom a prediction is being made will be taken into consideration before the final user's prediction has been made. These opinions will be expressed in different ways for the two approaches followed in this research.

First, in the approach where just one neural network is used across the whole dataset, new features were set to be created. Newly created features were set to be a one hot encoded k number of clusters. This means that, for example, in the case in which the neural network was tested in a scenario where users were classified in fifty clusters, the neural network would have fifty additional features which would all equal to zero except the feature describing the user's cluster.
Clustering was done using the K-means clustering algorithm, testing a different number of clusters in order to find a value k for which neural network is performing the best. In this problem, the scikit-learn implementation of the k-means algorithm has been used.
The following table shows features for which when used this approach performs most accurately:

Feature	**Description**
sin_tod	Minute of the day, encoded using sine function
cos_tod	Minute of the day, encoded using cosine function
dow	Day of the week, encoded using one hot encoding. This is not actually feature for itself, but rather a set of seven features representing the days of the week. One of these features has value 1 and the rest 0, depending on what day of the week check-in was logged on.
lat	Standardized latitude of the venue's location.
lon	Standardized longitude of the venue's location.
user_cluster	User cluster encoded using one hot encoding. This is a set of k features created after k-means clustering of the users. Feature which has value 1 corresponds to the cluster user has been assigned to, while the other values are 0

Table 6.3: Features used in one MLP model defined by Spatial, Temporal and Social segments

The second approach is based on the top-n most similar users. In this approach, users are again represented as TF-IDF vectors and similarity between all of them is calculated using cosine similarity. Twenty most similar users are stored for each user after calculating cosine similarities. The idea then is to take into consideration the preferences of n similar users to the user for who a prediction is being made. Doing these predictions, the weights of the user's prediction and similar users' prediction will be tuned, as well as a number of similar users that will be taken into consideration.
The idea behind this implementation is that by combining the results of different neural networks while predicting the next activity, the combination of results might give a more accurate end result of the prediction. The formulation of this approach is shown in the following pseudo-code:

Algorithm 1 N-MLP model defined by Spatial, Temporal and Social segments

```
w_u, w_su = define_weights()
sims = fetch_similar_users()
for entry : test_data do
    f, u = extract_features_and_user(entry)
    pred = make_a_prediction(u, f)
    preds = initialize_vector()
    for sim : sims[u] do
        preds.append(make_a_prediction(u, f))
    scores = initialize_vector()
    scores[pred] += w_u
    for p : preds do
        scores[preds] += w_su
    final_pred = get_final_prediction(scores)
```

The last potential implemented improvement was one hot encoding of the previously visited **top level category** if that visit occurred within the last t minutes (this parameter was changed in order to see for what value neural network performs the best). Since there are only ten top-level categories as described in CHAPTER 4, this did not increase the complexity of the neural network by a great margin. This feature, combined with the spatial features, tries to capture two of the enlisted segments at the beginning of the chapter (Area around frequently visited venues Proximity of the venues to the user's frequently visited venues).
The gathered results of all approaches did not show any improvements in neural network prediction accuracy so these features were discarded.

The results of both neural network approaches using different sets of features

are shown in CHAPTER 7.

Initially, after deciding which features were the most suitable for the neural network to accomplish the highest accuracy, it was decided to start parameter tweaking of the neural network. This process is considered as a fine tuning of the neural network, and the goal of the process is to find the most accurate neural network setting given the selected features. However, due to the schedule limitations and resources, it was not properly conducted. Some settings other than default ones were tested but not reported in this document as they were not conducted entirely and did not report large differences in results when compared to the result of default settings.

6.4 Summary

This chapter describes a detailed step-by-step process of testing different decision models. Besides two simple baseline models which use the frequency of check-ins to categories, and the second one using the temporal property of the check-in, there are two approaches in modeling activity preference of LBSN users.
These approaches achieve their functionality using a multilayer perceptron algorithm in different ways. In the first approach, a multilayer perceptron is created for each user. This way users are modeled to make "their own decisions" as only their check-in history is being considered within the training of the network.
On the other hand, the second approach has only one multilayer perceptron created which is being trained on entire training data. This way multilayer perceptron has an insight into the check-in history of all users.
Within these approaches then several sets of features representing decision segments were tested in a way that they had different encoding. The features which led to the highest accuracy of the models have been shown and used as final.
The first described approach considering the social segment as a final addition to the model uses cosine similarity to find possible influencers of users. In order to capture the social segment for the second approach some additional preprocessing in form of K-means algorithm was required to create the last set of features that represents one hot encoded clusters of users.

Chapter 7

Experiment and Evaluation

This chapter describes experiments conducted on different modeling approaches and the evaluation of those models.

Chapter starts by describing experiment setup and then follows with the SECTION 7.2 reminding feature configurations used for training of the models. Evaluation of baselines is shown and commented on in SECTION 7.3. The chapter finishes with evaluation of models captured in SECTION 7.4.

7.1 Experiment Setup

In order to correctly and precisely evaluate implemented models, a common environment needed to be set up.

The evaluation conducted concerns about how accurate the implemented models perform when different decision segments are used in their prediction. The prediction task of these models is to predict activity, in the form of a venue category, which is going to be performed/visited next by a certain user. In terms of a machine learning task, this would mean that different approaches and features are used in order to predict a class label, or a target, which in this case refers to the venue category.
In order to predict users' future activities, a process of model training needs to be executed on a part of the dataset. As described in CHAPTER 4, the dataset is split in training and test datasets containing eight and a half, and 2 months respectively. Therefore, the training dataset contains 467494 entries and the test dataset contains 103687 entries in this prediction problem. While predicting, the only data that learning was made on is the training data. This is with the exception of testing the models when the previous check-in was used as a feature. However, this feature is not used in the final models and is mentioned here for clarity.

The described setup is common for all of the implemented models. The differences between implemented models are in their concept and decision segments which they are using. In order to evaluate the quality of the implemented models, F1 and accuracy measures have been used.

7.2 Feature selection

n order to make the most accurate model which will capture a user's activity preference, an important user decision segments need to be determined. These segments are captured through different features which have input to the neural network and tested. The extensive table of all the tested features follows:

Two approaches have been described in CHAPTER 5 describing the improvements of:

- One neural network
- N neural networks

The first approach is one neural network using the check-ins from the whole training dataset as training input. Here is the list of features for which one neural network is performing most accurately:

In the second approach neural network is created for each user and it differs from the first approach in using more than one neural network in predicting an activity for the user. Following table enlist the feature for which this approach performs most accurately:

It is important to note here that N-neural networks model was performing the best in accuracy when it used only spatial and temporal segment. So this table shows what each of those networks was used as a set of features.

7.3 Baseline setting

As already mentioned earlier, in order to make comparisons between decision models and prove certain assumptions, two baseline decision models have been created. The purpose of the first model, Favourite Choice described in 5.1, was to inspect whether people are doing favorite activities frequently.

Feature	Description
user_id	User identification number
hour	Hour of the day
weekday	A number from a domain [1..7] representing the day of the week
sin_tod	Minute of the day, encoded using sine function
cos_tod	Minute of the day, encoded using cosine function
dow	Day of the week, encoded using one hot encoding. This is not actually feature for itself, but rather a set of seven features representing the days of the week. One of these features has value 1 and the rest 0, depending on what day of the week check-in was logged on
latitude	Latitude of the venue's location
longitude	Longitude of the venue's location
lat	Standardized latitude of the venue's location
lon	Standardized longitude of the venue's location
previous_category	Encoded category number representing the venue category in which user checked-in prior check-in to the current one. If there were no check-ins within the hour of the current one, this value is set to zero
prev_top_level_cat	Previous check-in encoded using one hot encoding. This is a set of ten features describing ten top level categories in a venue category hierarchy. One of these features has value 1 and the rest 0, depending on what was the tree of a venue category where user made the previous check-in. The criteria of this feature being 1 is described in more detail in SECTION 6.3
user_cluster_n	Number of the cluster from domain [0..k-1] user has been assigned to
user_cluster	User cluster encoded using one hot encoding. This is a set of k features created after k-means clustering of the users. Feature which has value 1 corresponds to the cluster user has been assigned to, while the other values are 0

Table 7.1: Table containing all tested features

On the other hand, the aim of the second model was to show that along with activity frequencies, the temporal feature is also of great importance. This model was

Feature	Description
sin_tod	Minute of the day, encoded using sine function
cos_tod	Minute of the day, encoded using cosine function
dow	Day of the week, encoded using one hot encoding. This is not actually feature for itself, but rather a set of seven features representing the days of the week. One of these features has value 1 and the rest 0, depending on what day of the week check-in was logged on
lat	Standardized latitude of the venue's location
lon	Standardized longitude of the venue's location
user_cluster	User cluster encoded using one hot encoding. This is a set of k features created after k-means clustering of the users. Feature which has value 1 corresponds to the cluster user has been assigned to, while the other values are 0

Table 7.2: Table containing features of the best performing model

Feature	Description
sin_tod	Minute of the day, encoded using sine function
cos_tod	Minute of the day, encoded using cosine function
dow	Day of the week, encoded using one hot encoding. This is not actually feature for itself, but rather a set of seven features representing the days of the week. One of these features has value 1 and the rest 0, depending on what day of the week check-in was logged on
lat	Standardized latitude of the venue's location
lon	Standardized longitude of the venue's location

Table 7.3: Table containing features of the best performing model

expected to provide more accurate predictions than the first baseline model due to the temporal segment it is based on. Favorite activity in this context is the activity which has the most logs in the training dataset.
Different from what was expected, the first assumption was proved to be correct but the second one was not. Comparison of the two baseline models are compared on figure 7.1.

Based on the captured results it can be seen that check-in of favorite activities happens in around 39% of cases which is a lot more than expected. Therefore, the assumption that people are repeatedly doing the same activities is proved to be true.

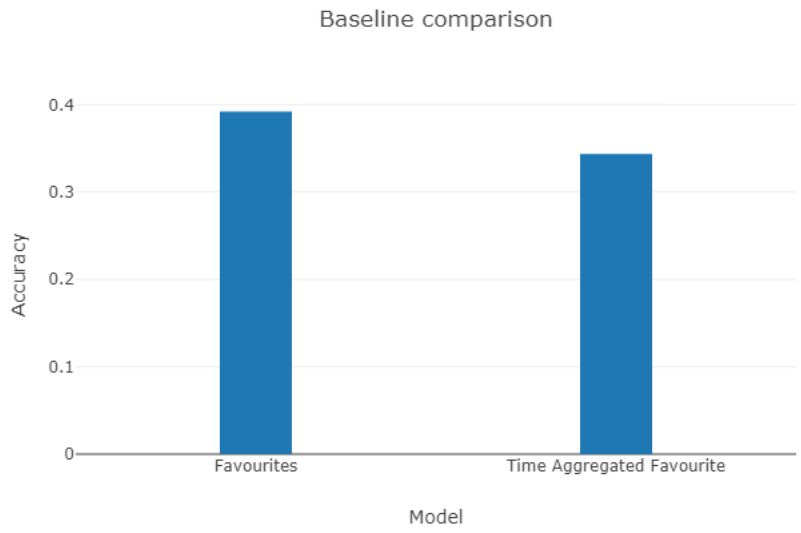


Figure 7.1: Accuracy comparison of baseline models

On the other hand, the assumption that people are doing the same activities given certain times of the day and day of the week turned out to be true in around 34% of check-ins from the test dataset. This accuracy was expected to be higher as the model was based on temporal segment along with the frequency of visits.

7.4 Model evaluation

MLP classifier was implemented using a scikit-learn Python package and evaluated using default parameters of the *MLPClassifier* method and feature combinations which were previously described in SECTION 7.2. The method parameters can be found in scikit-learn documentation at [14].

Both of the neural network approaches were tested with different combinations of temporal and spatial features. For example, the short table 7.4 shows two different feature settings and MLP accuracy for them in an approach where one MLP is used.

Adding longitude and latitude as features to the neural network on top of temporal segments resulted in the decreased accuracy in the prediction at first. Scikit-learn documentation notes that many models usually behave badly if a requirement of data standardization is not fulfilled. Therefore, after standardizing

Features	Accuracy
sin, cos, days	33.05%
user_id, sin, cos, days, lat, lon	35.8%
sin, cos, days, lat, lon	38.1%

Table 7.4: Table containing accuracies of the model using different feature settings

both features by removing their means and scaling them to their unit variances, the accuracy of the neural network improved by 5% in comparison to the neural network using only temporal features.

Results are shown on image 7.2 show the comparison of baseline models with the MLP models which capture temporal and spatial decision segment.

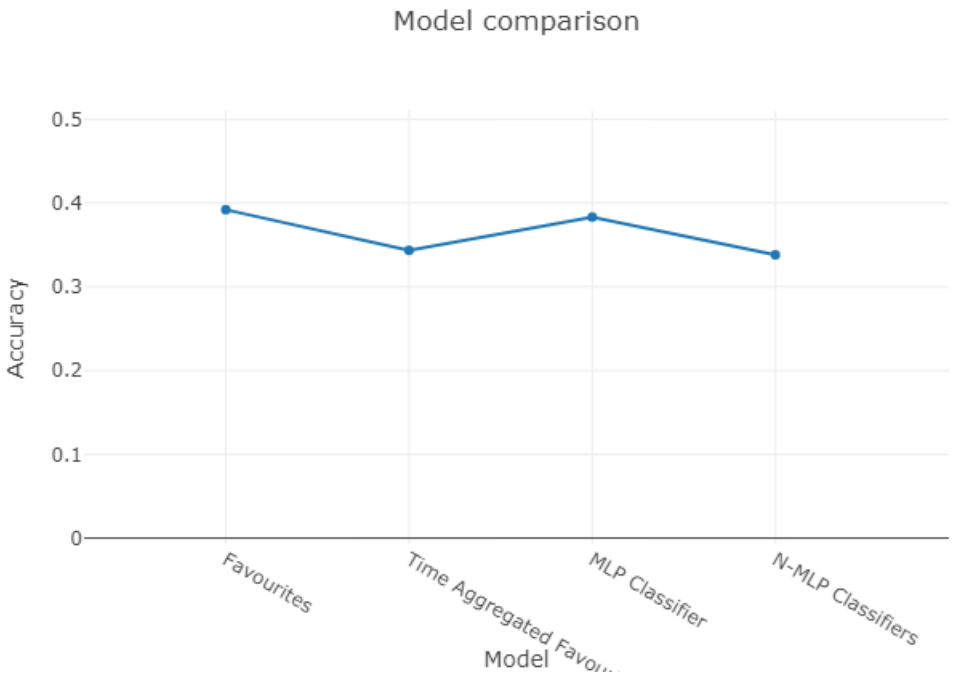


Figure 7.2: Accuracy comparison of models

It can be seen from the image that *Favourites* model is performing better than both of the MLP modeling approaches. One MLP classifier model is performing better than N-MLP classifiers and *Time Aggregated Favourites* model. With the current feature setup, *Time Aggregated Favourites* performs better than N-MLP classifiers.

On top of this approach, the previous check-in feature was used with the intention to improve the model accuracy. This turned out to be an approach that did not improve model accuracy by a large margin. So experiments were continued without using this feature. Tested results are shown in the following table for an approach when the neural network is used for each user separately:

Time window	**Mean**	**Standard Deviation**
30 minutes	33.78%	21.75%
1 hour	33.86%	21.79%
1.5 hours	33.94%	21.74%
2 hours	33.90%	21.87%
2.5 hours	33.85%	21.69%
3 hours	33.78%	21.77%

Table 7.5: Results of testing previous check-in feature

Lastly, experiments with the social decision segment were conducted by taking into consideration the activity preferences of similar users. The process of doing this was described earlier in CHAPTER 6 and is different for N-MLP and one MLP approach.
Evaluation of one MLP approach required tweaking one parameter which represents the number of clusters in which users are classified. This parameter also determines the number of features, as the cluster feature is one hot encoded. The following table shows experiment accuracy results of one neural network approach with respect to the number of clusters used in a test.

Number of clusters	**Accuracy**
5	37%
7	38.02%
8	38.28%
10	38.02%
20	38.18%
50	38.62%
100	39.17%
150	39.81%
200	39.99%
400	41%

Table 7.6: Model accuracy results with respect to number of clusters

It can be seen from this table that there is an increasing trend in accuracy as the number of clusters gets higher. Noticing this, the evaluation of neural network using one hot encoded *user_id* has been tried. Unfortunately, this turned out to be a very demanding task for a computer on which test has been made. Therefore, the evaluation has not been made.
In a real case scenario, however, the number of users gets enormously high making this kind of evaluation even more demanding and therefore even harder to conduct. With this being said, it is not a problem not having results of that evaluation.

Evaluation of the second approach based on all three decision segments came down to tuning three parameters. First of the three parameters sets a number n_{su} of similar users to the user for whose prediction is being made. This means that besides prediction which was made for the user, another n_{su} predictions are going to be taken into consideration for the final prediction that is going to be a weighted combination of these predictions. These weights, w_u and w_{su} are the remaining two parameters which have been tuned. They determine up to what extent is user's prediction relevant (w_u) compared to the predictions of similar users (w_{su}). The following settings have been tested and the results are as follows:

User weights	n_{su} **= 5**	n_{su} **= 10**	n_{su} **= 20**
$w_u = 0.75$, $w_{su} = 0.25$	33.36%	31.79%	30.56%
$w_u = 0.60$, $w_{su} = 0.40$	31.18%	30.34%	30.13%
$w_u = 0.50$, $w_{su} = 0.50$	28.15%	29.52%	29.95%
$w_u = 0.40$, $w_{su} = 0.60$	27.59%	29.11%	29.81%
$w_u = 0.25$, $w_{su} = 0.75$	27.56%	29.02%	29.79%

Table 7.7: Framework accuracy results using different parameters

From the accuracy results in this table, it can be seen that the model in the second approach performs the best when similar users are less considered, that is when the social segment is not used. In comparison to this approach not using social segment and having prediction accuracy of around 34%, this approach can not beat it in accuracy even with the best performing of all the tested settings.
The table also shows a trend in accuracy decrease as the weight of the user for who prediction is being made decreases. It was decided that the final model in this approach will not be using the social segment as a part of a decision-making process.

The final comparison of all implemented models based on MLP is shown on the FIGURE 7.3. This image sums up all the accuracy evaluations of MLP based models. Best performing model in terms of accuracy is one neural network using

all three decision segments as a part of the decision-making process.

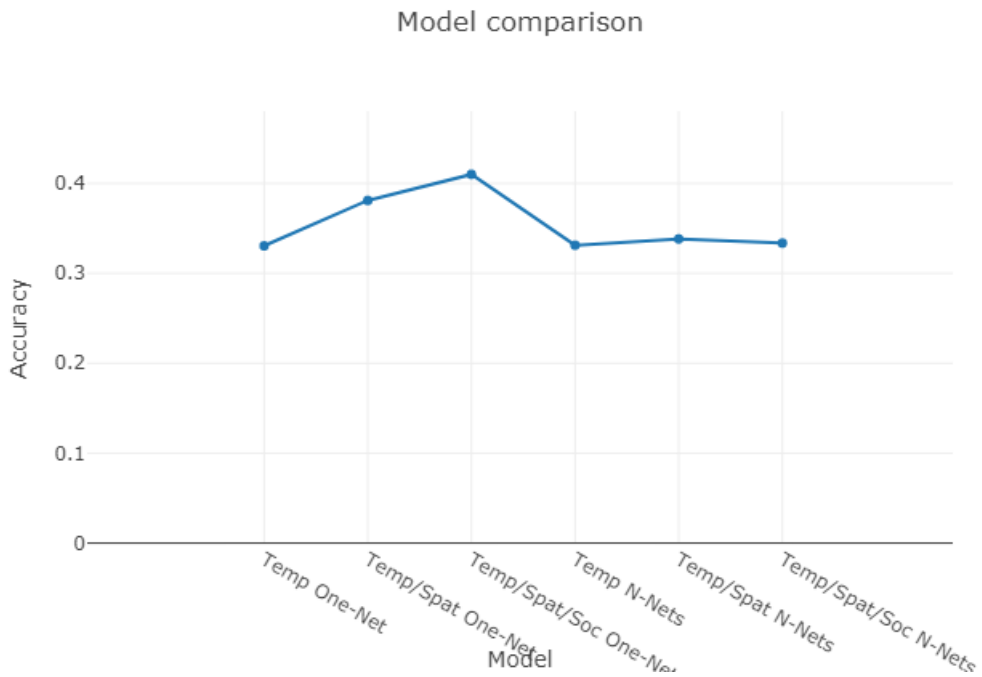


Figure 7.3: Accuracy comparison of MLP-based models

The result of 41% accuracy of this model is not better than the accuracy of a baseline *Favourite model* by a large margin. In order to get a better insight into the performance of the model additional evaluations have been made.

The first evaluation is the evaluation of the accuracy of predicting ten top-level categories in the category hierarchy. An argument for doing this is that the derived categories from the top level ones are very similar in their nature and they are all pointing to the similar activities, or rather a type of the activity which is being done.
After testing the neural network following this approach, its accuracy increased to 55.657%. This is 14 % better than the best performing model when all categories in the hierarchy as being considered as a target in classification.

Along with this evaluation, a confusion matrix was created for the ten top-level categories.
These results were reported in order to see which categories are neural network predicting, and to get a better understanding of this model.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	743	18	0	677	13	87	120	3	439	2021
1	1	539	0	186	4	38	22	26	99	619
2	0	0	0	0	0	0	0	0	0	0
3	228	117	0	5956	85	230	289	18	1873	6081
4	35	4	0	717	362	32	13	9	237	1224
5	53	10	0	516	31	1357	207	4	656	1814
6	61	4	0	924	20	178	1401	0	825	3165
7	12	6	0	50	5	19	12	134	56	248
8	191	55	0	2231	44	363	314	16	6219	8107
9	309	159	0	3818	107	499	781	56	3931	41564

Table 7.8: Confusion matrix for top level categories

Numbers from zero to nine represent categories 4.1 in the following manner:

- 0 - Arts and Entertainment
- 1 - College and University
- 2 - Event
- 3 - Food
- 4 - Nightlife Spot
- 5 - Outdoors and Recreation
- 6 - Professional and Other Places
- 7 - Residence
- 8 - Shop and Service
- 9 - Travel and Transport

Entry of the table indexed as $E_{i,j}$ represents number of observations known to be in group i but predicted to be in group j.
In other words, diagonal elements of the table represent the number of true predictions, and every other element in the table represents a number of false predictions. A very good indicator of model accuracy is if diagonal elements are higher in value then the rest of them in a row while others should be as low as possible, as that would mean that the model is predicting the correct category in most of the cases.

This trend can be seen in table 7.8, with the exception of columns 8 and 9 representing the most popular categories - "Shop and Service" and "Travel and Transport". Columns 8 and 9 have consistently high values for all the categories which are being predicted. This concern was addressed already in CHAPTER 4, when analysis of most popular categories was visualized.
One of the plots in that chapter was showing that for more than 75% of the users in the dataset "Travel and Transport" category is a favorite one (one that they have made most of their check-ins at). This fact as shown in the table hugely biases the results of the model. Categories "Shop and Service" and "Food" are generally the second most falsely predicted categories but to a lesser degree than "Travel and Transport" category.
If a person looks at this table and imagines it not having rows and columns number 3, 8 and 9, a table would contain a representation of a high-quality model.
These results indeed show some patterns in user activity with the exception of activities which include food consumption and shopping activities, along with transportation activities. One can argue that two firstly mentioned activities are quite impulsive and they can occur at any given time. A person can get hungry or thirsty at any time of the day, or left in need of consuming a tobacco product or similar. However, transportation activities should be quite periodic and easy to capture due to train schedules, job, and school schedules and similar. A person would not decide to impulsively travel for no reason. Therefore, the bias which check-ins to "Travel and Transport" category have created is left to be unresolved.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

As a part of research on human activity preferences, a human decision model has set to be created.
In order to achieve this goal, the work has been put in researching related work that is tackling similar issues in the location-based social network context. The Foursquare dataset has been acquired as a source of LBSN data, containing check-ins made throughout ten and a half months by the portion of Foursquare users in Tokyo, Japan. In order to get a better understanding and outline the thesis, data analysis has been conducted on previously preprocessed data. This was done successfully as insights on activities were extracted from the data along with occurring patterns in executing those activities. These insights were used in developing models described in chapters 5 and 6. The one multilayer perceptron approach using all three (temporal, spatial and social) decision segments turned out as the most accurate model in predicting the user activity. Different features and their encoding were used in testing both of the multilayer perceptron models. The features which essentially represent decision segments have been listed and explained in this work. After selecting the most accurate model and a feature setup, an additional experiment was conducted only considering top-level categories of the Foursquare venue category hierarchy. The experiment showed that three categories, but one of them in particular hugely biased the predictions of the model. The reason for the bias of these activities was concluded to be an impulsive nature of those activities. However, if a model is evaluated based on the rest of the categories, it would perform better by a large margin.

8.2 Encountered Challenges

As a part of the conclusion chapter, it was though as appropriate to include some challenges which were encountered throughout the work on this Thesis. The challenges were mostly concerning management and technical inexperience.
The biggest challenge was to be the only person on the team. Sometimes there was a need to discuss ideas with peers in order to clarify encountered obstacles or certain issues during the work. Consulting a colleague on these kinds of challenges would take far less time than overthinking about them just by one person.
The second challenge of this type was uncertainty about the thesis goal. The clear goal of the thesis was selected a month after starting the work. However, this only partly resulted in poor scheduling rather than it represented a big obstacle. This was overcome as there were some generic activities (data preprocessing and analysis, related work research) which have been done in parallel to setting a goal of the thesis.
Technical challenges that occurred were due to using a personal computer for the work conducted. This was done largely because of inexperience in participating in machine learning projects. Long computational times requiring high processing power were an obstacle to using a personal computer in everyday life.

8.3 Future Work

Possible future work includes additional research on human activities using other approaches than multilayer perceptron which might result in better pattern recognition. Another possibility would be to acquire a social network dataset with additional information inputs similar to the ones described in related work. The last possibility of an upgrade would be to create a model which would be suitable to provide users with recommendations on activities which might be of interest when they visit an unknown area.

Bibliography

[1] Vincent W. Zheng Zhiyong Yu Dingqi Yang Daqing Zhang. "Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs". In: (2014).

[2] Zhiyong Yu Zhu Wang Dingqi Yang Daqing Zhang. "A Sentiment-Enhanced Personalized Location Recommendation System". In: (2013).

[3] *Foresquare Venue Categories*. URL: https://developer.foursquare.com/docs/resources/categories.

[4] *Foresquare Venue Categories API*. URL: https://api.foursquare.com/v2/venues/categories.

[5] *FourSquare - NYC and Tokyo Check-ins*. URL: https://www.kaggle.com/chetanism/foursquare-nyc-and-tokyo-checkin-dataset/.

[6] Anant Gupta and Kuldeep Singh. "Location Based Personalized Restaurant Recommendation System for Mobile Environments". In: (2013).

[7] Bin Cui Zhiting Hu Ling Chen Hongzhi Yin Yizhou Sun. "LCARS: A Location-Content-Aware Recommender System". In: (2013).

[8] Mohamed F. Mokbel Jie Bao Yu Zheng. "Location-based and Preference-Aware Recommendation Using Sparse Geo-Social Networking Data". In: (2012).

[9] Ming-Joong Lee and Chin-Wan Chung. "A User Similarity Calculation Based on the Location for Social Network Services". In: (2011).

[10] *MNIST training with Multi Layer Perceptron*. URL: https://corochann.com/mnist-training-with-multi-layer-perceptron-1149.html.

[11] Thomas Dyhre Nielsen. "Machine Intelligence Lecture 8: Learning II". In: (2017).

[12] Xing Xie Yukun Chen Wenyu Liu Wei-Ying Ma Quannan Li Yu Zheng. "Mining User Similarity Based on Location History". In: (2008).

[13] *Single-Layer Neural Networks and Gradient Descent*. URL: http://sebastianraschka.com/Articles/2015_singlelayer_neurons.html.

[14] *sklearn.neural$_n$etwork.MLPClassifier*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html#sklearn.neural_network.MLPClassifier.

[15] Alia I. Abdelmoty Soha Mohamed. "Computing Similarity between Users on Location-Based Social Networks". In: *International Journal on Advances in Intelligent Systems, vol 9 no 3 4, year 2016* (2016).

[16] *Tips on Practical Use*. URL: https://scikit-learn.org/stable/modules/neural_networks_supervised.html#classification.