Next Place Prediction and Alert System

Forum Dattani[1], Parth Samant[2], Dhruv Parmar[3], Prof. Nileema Pathak[4]

[1]Student, Dept. of Information Technology, Atharva College of Engineering, Mumbai.
[2]Student, Dept. of Information Technology, Atharva College of Engineering, Mumbai.
[3]Student, Dept. of Information Technology, Atharva College of Engineering, Mumbai.
[4]Professor, Dept. of Information Technology, Atharva College of Engineering, Mumbai.

---***---

Abstract - *One of the most powerful features of modern smartphones is their ability to provide us with real-time, locationally aware information. For instance, if you enable the location services on your device, you can find out the approximate amount of time it would take for you to get to your desired destination using your smart phone after you offer the necessary data. The location services on mobile devices have provided us with a number of advantages and advancements but it has its own limitations. The user needs to input certain before they can get any relevant information on any given subject. Predictability is a tool provided to us with the advancement in technology and artificial intelligence. By processing the previously stored data, patterns can be observed in the behavior and a prediction model can be developed.*

Imagine the following situation. You hop in your car to head to the gym after work. You don't turn on your phone's navigation because you know the way, but there is an accident on your route and a lot of traffic. If your phone knows where you are going without you telling it, it could warn you before you start driving and help you reroute. If our smartphones could predict our destination by observing patterns in our previous behavior, they could give us more relevant information about the places we are heading to. The idea of someone having knowledge of your previous, current and future whereabouts may seem unnerving but if this data is processed on your individual devices without any secondary storage or data revelation, the entire concept becomes much more appealing. For our project we created a system that can predict a user's next location using their current and previous location with regards to time, learning only from that user's location history.

Key Words: Location Prediction, Preprogramed alerts, Random Forest, Location history dataset, Data processing, Alert System.

1. INTRODUCTION

As technology advances, cellular phones continue to become more and more vital for users. Smartphones are the most widely used devices, given the services and features they offer to the users. Expanding the horizon of services offered by smartphones has become a prime focus for numerous organizations, individuals and industries. Personalization of these services and information with respect to each individual user has become the leading medium for keeping users engaged with the applications.

In this project, we are developing a system that can predict the prospective locations a user might visit at a particular time of the day or a particular day of the week in near future, based on the information obtained by processing the user's location history with regards to time of visit, time spent at a location, etc. The question this system is trying to answer is: If the user were to leave from a given location at a particular time, what is the most likely destination they would go to? This question can mathematically be deduced as follows:

P [Destination | Current (Time, Date, Location) && History (Time. Date, Location)]

Though the question may seem pretty basic when considering the answers pertaining to oneself, answering this question for multiple user, where each of them have their own lifestyle, schedules, habits and interests, answering this question for each of them using a centralized algorithmic system becomes much more complex and interesting. A number of challenges arise when a location history is the basis of analysis, since this dataset is often noisy and the datatypes may even be varied. In spite of these challenges, the results being generated are quite encouraging and usable for real-world services and applications.

Imagine the following situation. You are about to leave your home to head to college. It is submission day and it is necessary to carry a form which is important for the submissions. What if your phone knows you are going to college, and that's why, it gives you an alert to remind you to carry it with you? Also, consider, it is a holiday and you don't have to go. In that case, the alert could be irrelevant and therefore the system wouldn't give you the prompt.

2. LITERATURE REVIEW

In paper [1] "Predicting New and Unusual Mobility Patterns", authors Victor Liang, Vincent T.Y. Ng focused on extracting data from GPX file for further classification crucial for making predictions. Data for this project was collected using the GPS location tracking app, Moves. The project, uses his personal location data collected over the last two months and a friend's location data collected over a summer. This location data was obtained in the form of GPX files. GPX files are designed to store information about a user's movements throughout the day, so it tracks the latitudes and longitudes of locations visited, in addition to the routes taken between those locations. With a set of discrete locations on hand it is viable to build my training examples. Training examples are built using pairs of consecutive location points from the location history. These pairs of points (A, B) are instances from the dataset where the user travelled from A to B. The input vector consists of features of the date, time, and location of the first point in the pair. The target variable is the location of the second point.

• A binarized version of the input location. For example, if there are 5 possible location classes and the current data point is from location 3, then we have the features [0, 0, 1, 0, 0].

• A binarized vector of the day of the week. This is intended to catch the weekly repeating patterns in schedules.

• Monday is represented as [1, 0, 0, 0, 0, 0, 0].!

• A binarized vector representing the current time broken into 6-hour bins. This means that times are grouped into the following bins: 12am to 6am, 6am to 12pm, 12pm to 6pm, 6pm to 12am. For example, 5am would be represented as [1, 0, 0, 0].

• A binary feature indicating if the time is AM or PM.

• A binary feature indicating if the day is a weekend.

In paper [2] “Class-boundary alignment for imbalanced dataset learning” authors Wu, G. and Chang E. made the analytical research of striking trajectory capturing of geolocation and deviance from the usual observed trajectory. Overall logistic regression performed the best on both datasets. This is most likely because logistic regression is pretty resilient to noise, and there is a lot of noise in both datasets. Logistic regression is not trying to build hard boundaries between classes, like SVM, but instead uses a one vs all classifier system where it just assigns a score to each possible classification. It is interesting that logistic regression, one of the simplest classifiers, can outperform others on small, noisy datasets like these. SVM on the other hand struggled to deal with the fewer number of points. In fact, on dataset 2, the SVM resorted to classifying every test instance as “home.” There are a number of properties of location data that make it tricky to work with, and the results reflect. First, these datasets are small. For reference, my personal location history collected over two months was reduced to just 112 data points after filtering out rare locations. This is not a lot of data, especially when dealing with more complex schedules that contain a lot of noise. In order to learn more complex schedules, we need more features, but adding features increases the number of parameters that need to be learned which requires more data. As a result, my system uses relatively few features to try to extract the core parts of the user’s schedule.

In paper [3] “Location Prediction on Trajectory Data: A Review” authors Ruizhi Wu, Guangchun Luo, Junming Shao, Ling Tian, and Chengzong Peng introduced some of the common trajectory data preprocessing methods like noise filtering, stay point detection, and trajectory compression. The performance of this system depends heavily on the dataset given to it. Specifically, the performance is going to be much better if we use the data of someone with a consistent schedule rather than that of someone with a more irregular schedule. To make sure testing both ends of the spectrum using two datasets, irregular location history from this school year and my friend’s more regular location history collected while working over the summer. The first step in evaluating these models is to actually select the data for training and testing. Used two different methods to train and test the models. The first method is k-fold cross validation. It splits the data into k folds, trains on k-1 of them, and evaluates on the remaining one. I average the results for k iterations. This method uses less data for evaluation and saves more for training which is useful given that have pretty tight data constraints. The second method is intended to simulate online learning. In this method, I move through the dataset in chronological order and train on all data up to the current datapoint n-1 and test on the datapoint n. The results of each evaluation are averaged. This method is intended to simulate a real environment where this system would be employed. In the real world, the model will take in more data over time so it is important to evaluate how it will perform as the amount of available data increases. Here they used a few different metrics to evaluate the predictions of the models on the test set. Classification accuracy is the most obvious metric and it simply gives the fraction of the test set for which the model predicted the correct output. This can be useful, but is not always the most indicative of true performance.

In paper [4] “A probabilistic approach to mining mobile phone data sequences” authors Farrahi,Katayoun and Daniel Gatica-Perez considered how to balance the expected benefit of asking a driver to confirm their destination (making the suggested places more relevant) against the cost of interruption. This approach is useful in avoiding the "talking paperclip" syndrome where anticipatory applications interrupt users too much. It provided a Bayesian probabilistic model of individual location behavior that is a hybrid of Linear Discriminant Allocation (LDA) and the eigen behaviors approach of Eagle and Pentland. The second method is intended to simulate online learning. In this method, move through the dataset in chronological order and train on all data up to the current datapoint n-1 and test on the datapoint n. The results of each evaluation is averaged. This method is intended to simulate a real environment where this system would be employed. In the real world, the model will take in more data over time so it is important to evaluate how it will perform as the amount of available data increases. It used a few different metrics to evaluate the predictions of the models on the test set. Classification accuracy is the most obvious metric and it simply gives the fraction of the test set for which the model predicted the correct output. This can be useful, but is not always the most indicative of true performance this is especially true in data with an unbalanced output class distribution.

3. NEXT PLACE PREDICTION SYSTEM

Next place prediction is an alert system that provides location specific alerts preprogramed by a user, for locations that a user is most likely going to go to, based on where they have been and when. It can be implemented where in the future location of routine user is to be predicted in order to be given alerts and updates depending upon his expected location when they’re leaving from their home location. The proposed system can be use by various large scale, data-oriented corps where in user insights and services play an important role, without disregarding the privacy of users. Also, this system can be integrated with various software-based robots, personal assistants, or other artificial intelligence systems to provide recommendations with regards to where the user might go. The system is useful to reduce forgetfulness in users when leaving to go somewhere by reminding them about relevant information with respect to the destination before they leave.

The system uses historic location data as input in the prediction model to obtain insights, because of which, the system never stops learning and improvising. As the size of the dataset being fed to the algorithm grows, the precision of predictions considerably surges and the variability in output thus decreases. The algorithm used here to obtain predictions from the location history is Random Forest.

The random forest model is fed with the user's time and day of the week as independent input features. The location of the mobile user is determined using the Latitudes and Longitudes of the JSON location object of the user, while the time and day of the week can be extracted from the timestamp element. The location element serves as the dependent variable of the prediction model. Once the location has been predicted, the system checks for any reminder set pertaining to that location in the database. If set, the system displays the reminder as a prompt message.

The fundamental concept behind the random forest algorithm is very basic yet strong. The algorithm relies on the wisdom of the crowds. In data science speak, the reason that the random forest model works so well is:

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models. [5]

Random forest algorithm becomes more and more effective as the size of the dataset grows without affecting the processing capabilities, or basically slowing down the system. Because of its accuracy and computational upper hand, this algorithm was used to implement the next place prediction and alert system.

3.1. BLOCK DIAGRAM

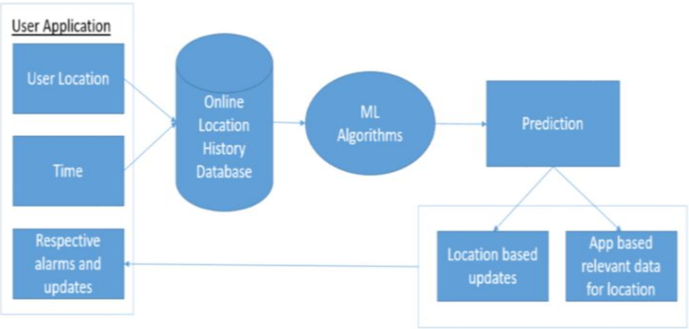

User Application
User Location
Time
Respective alarms and updates
Online Location History Database
ML Algorithms
Prediction
Location based updates
App based relevant data for location

Fig -1: Next Place Prediction System Block Diagram

4. RESULT AND DISCUSSION

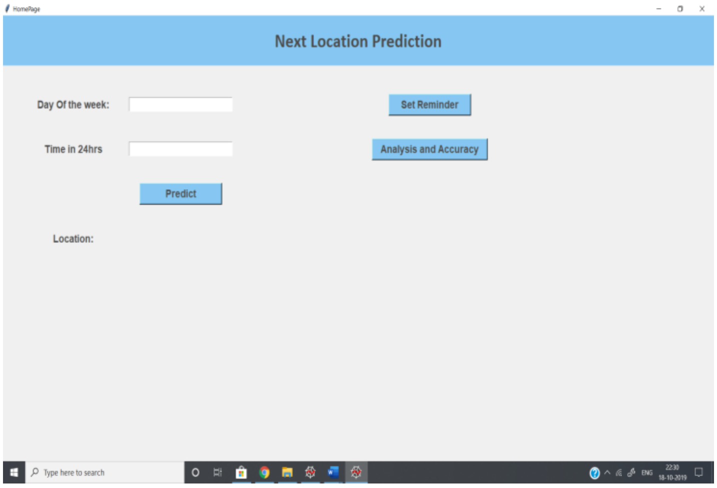


Fig -2: Home Page

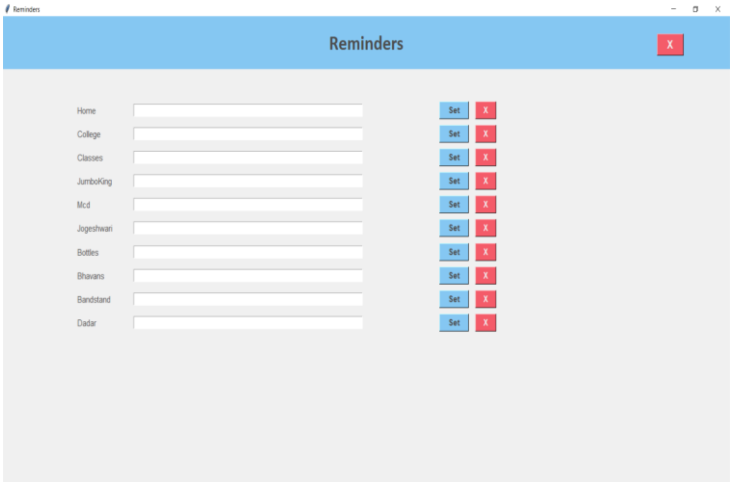
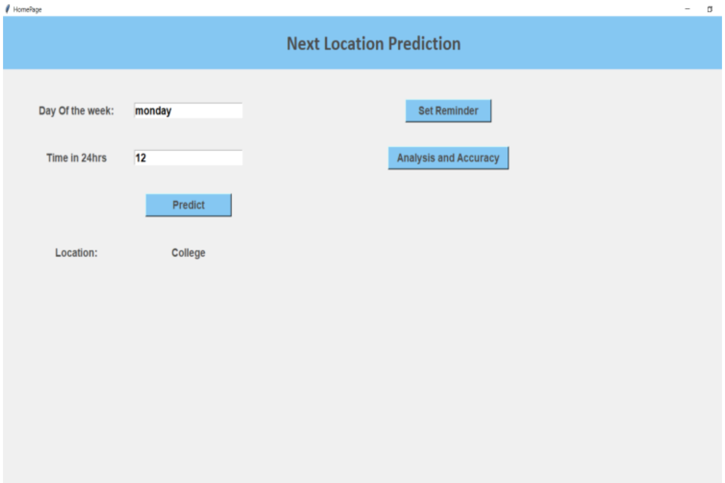


Fig -3: Setting Location Based Alert



Day Of the week: monday

Time in 24hrs 12

Location: College

Fig -4: Location Prediction

The user can preprogram location specific alerts based on the current records of locations visited. A user can input future date and time to predict possibilities of where they might be by analyzing the available data. The user can delete or update the reminders and can also check the accuracy rate of the predictions being made by the system.

5. CONCLUSIONS

Location Specific predictions and alerts help obtain alerts related to a place, before the user even leaves their house to go there. A system for accurately predicting the most likely future location of a person that does not have a completely monotonous lifestyle was developed. This paper elaborates how this system achieved the goal of providing users with information, alerts and reminders they may find useful w.r.t the location the system predicts the user might be visiting. By using the random forest algorithm, maximum accuracy and precision was obtained in the computation of data. This system provides a personalized, advanced and efficient experience to the user without invading on their privacy or without disclosing any location data the user provides as input to the system.

ACKNOWLEDGEMENT

We would like to express immense gratefulness to our guide Prof. Nileema Pathak for the motivation and guidance throughout. We would also like to thank the faculty of Information Technology Department who greatly assisted our research and for their valuable help

REFERENCES

[1] V Liang, PhD Thesis, Department of Computing, The Hong Kong Polytechnic University, in 2018 IEEE 11th international symposium on biomedical imaging "Predicting New and Unusual Mobility Patterns"

[2] Wu G. and Chang E. "Class-boundary alignment for imbalanced dataset learning"

[3] Ruizhi Wu, Guangchun Luo, Junming Shao, Ling Tian, and Chengzong Peng "Location Prediction on Trajectory Data: A Review"

[4] Farrahi, Katayoun and Daniel Gatica-Perez "A probabilistic approach to mining mobile phone data sequences"

[5] Tom Yui, Data Scientist at Solovis, Doing my Best to Explain Data Science and Finance in Plain English "Understanding Random Forest-How the Algorithm Works and Why it Is So Effective"