A Location Management System for Destination Prediction from Smartphone Sensors

Sun-You Kim, Sung-Bae Cho
Dept. of Computer Science
Yonsei University
Seoul, Korea
{sykim@sclab.yonsei.ac.kr, sbcho@yonsei.ac.kr}

***Abstract*— Several applications based on smartphones have been developed for user's requirements. Among them, the location based services (LBS) are demanding to many people. As a result, location management systems become more important to manage the locations and acquire new information in places because location information can be obtained in diverse sensors. This paper proposes a location management system which manages the information of location and predicts a future location and moving path from the current sensor values. The proposed system consists of five modules. Three modules perform to manage the information of location and user's context, and two modules predict future information about the locations. In order to show the feasibility of the proposed system, we conducted the evaluation on each module with a real dataset collected from mobile devices.**

Keywords-Location management; location-based service; destination prediction; hidden Markov model

I. INTRODUCTION

As many people use smartphones, such as Android phones, the smartphone market grows rapidly [1]. Smartphone is easy to develop the applications using device sensors because various mobile OSs provide open platform. It enables developers to access easily user's locations and sensor data.

In many cases, the applications of smartphones such as Location-Based Service (LBS) use location information in mobile device [2][3][4]. With the increase of LBS, many researchers investigate on the locations using smartphone. As a result, it is necessary to manage the data of locations and forecast user's future locations based on the sensor information.

In a mobile phone, the system which manages location information and predicts future places should include the following functions. Each function is classified into two categories. First category is management of location information. It manages user's Point Of Interests (POI) and finds new locations which are meaningful for user. Second category is a service using location information. It provides new information such as user's current location, moving time and destination. This system, which includes two categories, is composed of the following functions.

1) *Data collection:* It collects sensor data and user's information. Using data collection, the system generates new information.

2) *Location extraction:* It extacts location, which is a frequently visited place or meaningful place for user. By extracting meaningful location, location management system can recommend to register symbolic location for user.

3) *Location recognition:* It means to classify where user's current location is. The system is able to offer a suitable service by finding user's location exactly.

4) *Prediction of departure time:* It forecasts when user departs. The new information required for future can be allowed to the user at the appropriate time by using it.

5) *Prediction of destination:* It predicts the user's place and path of desitination not yet reached. Destination prediction should use all information of location manager system because it requires many data.

In location management system for destination prediction, user's mobile phone gathers data using sensors. And it extracts the POI, which is meaningful for user, using collected data. Extracted locations are managed by this system. Also, it classifies user's current location and predicts user's departure time in present location. Based on this information, this system predicts user's future location.

Some researchers have studied on the prediction of destination using various smartphone sensors. Do *et al.* proposed a location prediction method using linear regression, logistic regression and random forest [5]. It uses the information of mobile device, such as GPS, Bluetooth, call log, application history and proximity. Lu *et al.* developed a forecasting method based on Support Vector Machine (SVM) using GPS, acceleration, Bluetooth, Wi-Fi and call log in mobile device [6]. Kim *et al.* used the Bayesian network for destination prediction [7]. Its input values are location information, visiting time, staying time and user's gender.

Gambs *et al.* proposed a method for prediction of destination using mobility Markov chains with Point Of Interest (POI) sequence [8]. Liao *et al.* developed a destination forecasting system based on hierarchical dynamic Bayesian network with GPS [9]. Simmons *et al.* proposed a destination prediction framework to use hidden Markov model using GPS sensor and map database in mobile environment [10]. However, previous studies did not build a system to manage the whole information related with locations. This paper proposes a system to manage several sensor data related with location and predict the future location.

The rest of this paper is organized as follows. Section II describes each module that constructs the proposed system, which aims at the management and service. Section III addresses the result of experiments on each component.

Section IV summarizes the location management system for the destination prediction and draws a conclusion.

II. THE PROPOSED SYSTEM

The proposed system consists of two functions, which are the management of location and the service with location. The location management is conducted by data collection and location extraction. The location service is provided by transportation recognition, location recognition, moving time prediction and destination prediction.

To manage locations in this system, first, this system collects sensor data and user's input information in data collection module. Then, it identifies user's meaningful locations using location extraction module. Extracted locations are managed in symbolic locations.

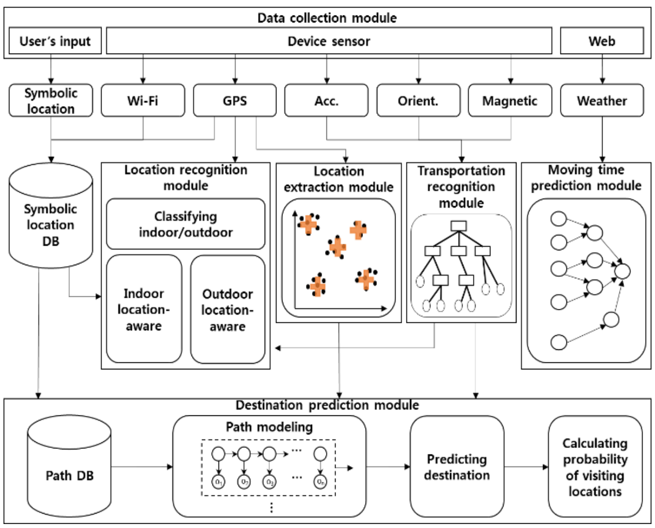


Figure 1. System overview

This system provides three service functions. It offers user's present location using location recognition module, and predicts departure time in future with moving time prediction module. Finally, it gives information about user's future location and moving trajectory using destination prediction module. Figure 1 illustrates an overview of the system.

A. Sensor data collection

We collected raw data such as acceleration, orientation, magnetic field, GPS information and Wi-Fi information by using Android phone API. GPS information includes latitude, longitude, accuracy, number of satellites, and SNR (Signal to Noise Ratio). Wi-Fi information contains SSID (Service Set Identifier), mac address, and RSSI (Received Signal Strength Indication). Most of data are collected twice per one second. GPS information is collected whenever the state of GPS sensor is changed. When user registers a location, Wi-Fi information is obtained.

Also, we collected weather information using Yahoo weather API. The weather data express temperature and weather state of 47 types. We transform the weather state into 7 types. The data specification and frequencies are summarized in Table I.

TABLE I. SENSOR DATA FOR DESTINATION PREDICTION

Sensor Type	Frequency	Description
Acceleration	Two times for one minute	3 axis acceleration (-2g~2g)
Orientation	Two times for one minute	Orientation, pitch, roll
Magnetic field	Two times for one minute	3 axis magnetic field (uT)
GPS	When GPS state is changed	Latitude, longitude, accuracy, SNR, number of satellites
Wi-Fi	When user registers a location	SSID, mac address, RSSI
Weather	Once for five minutes	Temperature (℃), Weather state (7 types)
Time	Two times for one minute	Current time

Acceleration, orientation and magnetic field can be used to check user's transportation mode. GPS is the information necessary to perform the location extraction and the location recognition. Wi-Fi is used to recognize indoor location, and weather is input of prediction of moving time.

This system stores the names of symbolic location, which are entered by user. Each location name is connected with Wi-Fi information and GPS information such as latitude and longitude.

B. Location extraction

For inducing to register user's meaningful locations, this system extracts candidate locations, which can be meaningful. Because it is impossible to use all GPS data, which are very big size in mobile device, the locations extracted are transformed into symbolic locations.

Previous studies about location extraction used *k*-means clustering, which is density-based algorithm [11]. However, *k*-means clustering should determine the number of '*k*'. In the location extraction, '*k*' means the number of locations. We do not know the number of locations extracted in advance. User's meaningful locations follow a Gaussian distribution [12][13]. However, because the criteria of density in *k*-means clustering are ambiguous, *k*-means clustering is not suitable in the location extraction problem.

Instead of it, we use G-means clustering method [14]. This is a clustering method to test each cluster in Gaussian distribution through statistical verification and repeat the *k*-means clustering until all clusters follow the Gaussian distribution. The statistical verification is performed by Anderson-Daring test, which is represented by the following equation:

$$A^2(Z) = \frac{-1}{n}\sum_{i=1}^{n}(2i-1)[\log(z_i)+\log(1-z_{n+l-i})] - n \quad (1)$$

Here, x_i is transformed into a value of average 0 and variance 1. When $x_{(i)}$ is the i-th value, we define $Z_i = F(x_{(i)})$. In this equation, F is $N(0,1)$ Cumulative Distribution Function (CDF). Using the latitudes and longitudes, which are obtained from GPS sensor, this system performs G-means clustering.

After the clustering, user's key locations are extracted. The extracted locations contain the information of latitude and longitude. Figure 2 is an example of the result of location extraction.

Extracted locations is used by two objects. Extracted locaitons of stop state are user's meaningful locations, and all the extracted locations are used for constructing a path. A path has many points of location. Therefore, it is nessesary to reduce locations, which express a path. We use the extracted locations to make a path.

Figure 2. An example of location extraction

C. *Transportation type recognition*

The transportation recognition needs to judge whether to perform the location recognition. It is necessary as an input for the destination prediction. To classify the moving state, we transformed sensor data such as acceleration, orientation, and magnetic field using decision tree algorithm. In some cases, decision tree shows better performance than alternative algorithms to process time series data such as acceleration [15].

For using sensor data in mobile device for decision tree, we extract some features such as the difference between previous and current sensor values, average sensor value for a specific period, and standard deviation of the sensor value for a specific period. Following (2)~(4) are the equations for pre-processing in decision tree.

$$sum_X = \sum_{i=1}^{N} \sqrt{(x_i - x_{i-1})^2} \tag{2}$$

$$mean_X = (\sum_{i=1}^{N} \sqrt{(x_i - x_{i-1})^2})/N \tag{3}$$

$$std_X = \sqrt{\sum \sqrt{((x_i - x_{i-1})^2 - mean_x)^2}} \tag{4}$$

In the equations, X means specific sensor, and x_i represents the i-th sensor value. Equation (2) is the summation of the difference between previous value x_{i-1} and current value x_i from a sensor X. Equation (3) represents the mean of the difference value which is calculated by equation (2). Equation (4) denotes the standard deviation of the difference between previous and current sensor data. Acceleration, orientation and magnetic field exist in 3-axis. This method uses average and standard deviation as the feature of decision tree. Each sensor is calculated by three averages and three standard deviations [16].

Using these features, decision tree is generated, which classifies the transportation types about input values, which are acceleration, orientation and magnetic field. Transportation types are staying, walking, running, and in vehicle.

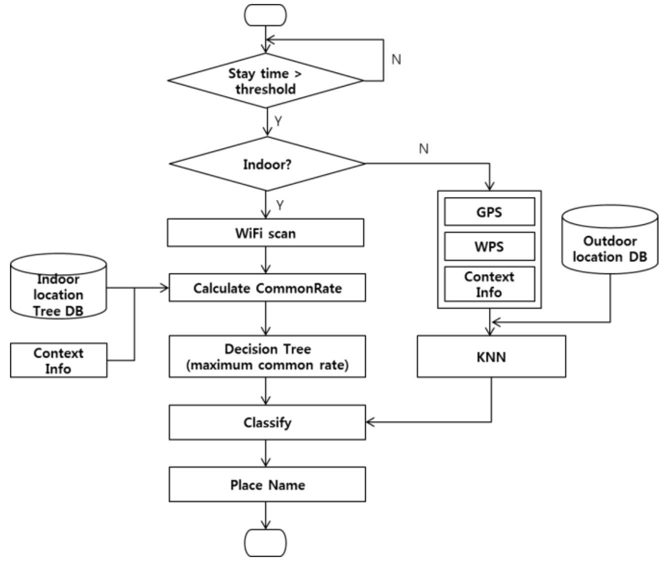


Figure 3. Process of location recognition

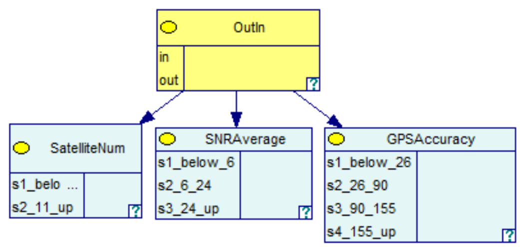


Figure 4. Model of na ïve Bayes classifier for distinction of indoor / outdoor

D. *Location recognition*

In the location management system, it is an important problem to identify user's location. In order to know user's place, this system includes the location recognition module. For recognizing an outdoor place, it is easy to identify user's current location by using GPS value of latitude and longitude. However, in an indoor location, the signals of GPS satellite cannot pass wall of the buildings. Therefore, we should use a different method for an indoor location. Recently, in the field of the location recognition, many researchers use Wi-Fi AP (Access Point) [17][18]. Thanks to the ubiquitous Internet, it does not need additional configuration, so that we use Wi-Fi AP information to identify user's current location in indoor. The process of the location recognition is shown in Figure 3.

For accurate recognition of user's current location, this system distinguishes between indoor and outdoor. Next, if the result is outdoor, it performs outdoor location recognition using the *k*-nearest neighbor. Otherwise, it executes the algorithm for indoor location recognition using decision tree.

1) Classification of indoor/outdoor: To discriminate indoor and outdoor, we use the na ïve Bayes classifier, which is a fast and simple inference method. Naïve Bayes is a probabilistic model, which is based on Bayes rule under the strong conditional independence assumption. In the proposed method, the input of naïve Bayes includes GPS information such as number of statellite, SNR, and GPS accuracy. The number of statellite converts to 2 discrete values, which are seperated by a threshold of 11. SNR is made by 3 values, and GPS accuracy is transformed into 4 values. The preprocessing values are entered by inputs of naïve Bayes model. The naïve Bayes model used is shown in Figure 4.

2) Outdoor location recognition: k-NN method is used to identify outdoor place. k-NN classifies the data by performing majority vote with the k neighbors closest to the input data. In the stored symbolic locations, after selecting k GPS points closest to the current position, it selects a location, which is the largest number in k-point locations.

3) Indoor location recognition: For indoor place, recogintion method cannot use the GPS sensor because of the unavailable GPS signals. So, we adopt the Wi-Fi finger print method based on the decision tree. Decision tree is generated by Wi-Fi information such as RSSI and MAC address, which is from the previously stored symbolic location. It makes a result, which is symbolic location to use decision tree with the new input of Wi-Fi information.

E. Prediction of moving time

If the system predicts the departure time, it can offer information, which is necessary for user in advance. To predict user's departure time based on context has the disadvantage, which is high error rate because it should determine a specific time in 24 hours. So, our system calculates user's staying time and predicts departure time after inferring how long the user stays at the current place.

The system uses Bayesian network for the prediction of user's moving time. Bayesian network is a stochastic model, which has a Directed Acyclic Graph (DAG) structure and Conditional Probability Tables (CPTs). Bayesian network is used to handle the uncertainty with probability. It supports the efficient probability calculation based on conditional independence assumption. The Bayesian network modeled in this system is as shown in Figure 5.

This network is a structure, which has root nodes, intermediate nodes, and observation nodes. Observation nodes are for input of system time, current location and weather information. Intermediate nodes calculate the probability values based on evidence values of observation node. The root nodes are computed by intermediate nodes. Each root node means at least 1 hour, 1~2 hours, 2~4 hours, and 4~8 hours, which are the result values in this module.

F. Destination prediction

The destination forecasting informs the location that user has reached at the last time and calculates the probability of visiting location, which is intermediate location of the path. Destination prediction uses Hidden Markov Model (HMM) [19]. HMM is a statistical model characterized by a Markov process with unknown parameters, modeling observations to

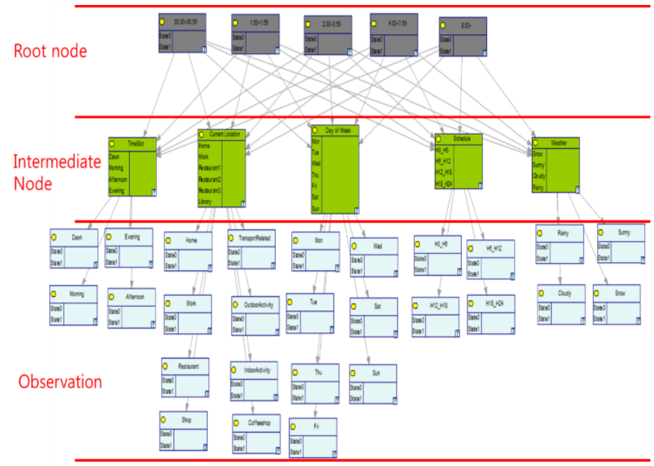


Figure 5. Model of Bayesian network to predict moving time

determine these hidden parameters. HMM is a widely used technique that stochastically models sequence data of the time series. It is mainly composed of the state transition probabilities, which select the observation value at each state. HMM composed by state transition probability A, probability distribution of observed value B, and probability distribution of initial state Π. One HMM, λ, is represented as (5).

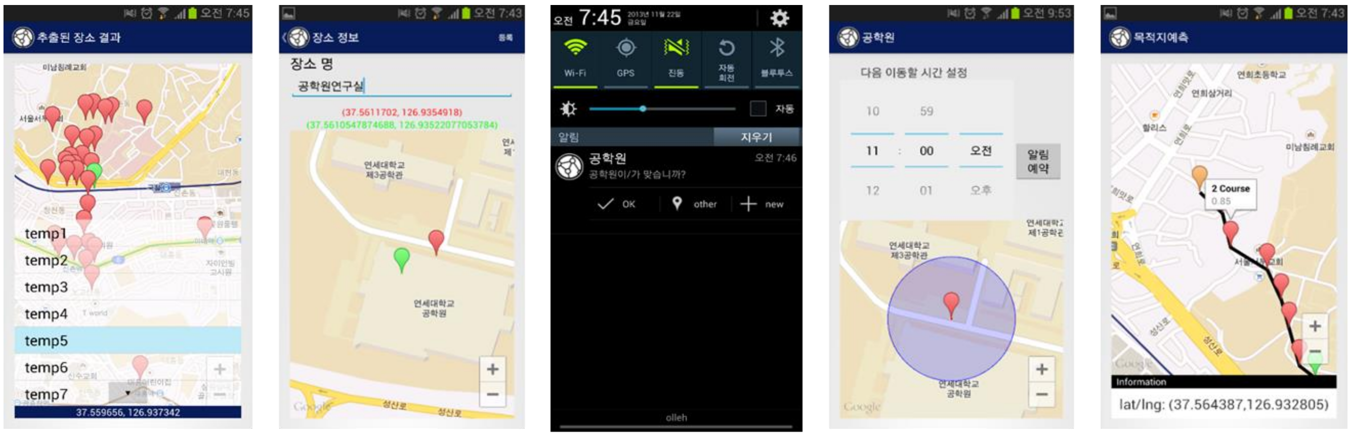
$$\lambda = \{A, B, \Pi\} \tag{5}$$

In the proposed system, destination prediction uses extracted locations, transportation type and time. This module consists of the three parts.

1) Building path model: The HMM of path has information about the start and end points of a path. HMM is built by number of pairs of source and destination locations. Path information is made up by sequence, which contains the extracted location, the transportation type and the time that is quantized. The HMM included the path information is learned by Baum-Welch algorithm [20], which is a learning method typical to represent probabilistic information of multiple sequences.

2) Predicting destination: About the new input, which is information of departure or sub-path, this method evaluates all HMMs for finding the path of the highest similarity. Evaluation is conducted by the forward algorithm, which is basic method to check the similarity between a sequence and an HMM.

3) Calculating visiting probability: Based on destination. which is determined by optimal path, the probabilities of visiting destination and intermediate locations are calculated. First, we find out an optimal path sequence, which is the same as a departure and a destination of optimal HMM and includes current path, from the path repository. By determining a sequence of future movements of the location, it finds out the optimal state sequence from the HMM and calculates the probability of visiting locations based on the optimal state sequence. The calculation of the optimal state



(a) (b) (c) (d) (e)

Figure 6. Interface of the proposed system

sequence is conducted by Viterbi algorithm [21]. It can determine the most probable sequence of states in optimal state sequence.

III. EXPERIMENTAL RESULTS

In order to evaluate the proposed system, we applied it to a SAMSUNG Galaxy S4 Android phone and conducted experiments. Data set is gathered for four months from ten people. The specification of data set is shown in Table II.

We implement this system using Android API. In order to speed up the operation, all core modules are implemented by using Android NDK API. The NDK is a toolset that allows to implement parts of App using native-code languages such as C and C++ [22].

TABLE II. DESCRIPTION OF DATASET

	#Location	#Path	Size of storage
User 1	16	193	2.44GB
User 2	20	268	2.62GB
User 3	32	149	1.46GB
User 4	50	288	3.41GB
User 5	42	309	3.66GB
User 6	32	233	1.34GB
User 7	28	236	1.48GB
User 8	24	294	3.21GB
User 9	36	237	2.37GB
User 10	14	189	2.08GB

The interfaces for each component are illustrated in Figure 6. Screen of (a) is an interface, which shows the result of the location extraction. The interface for entering the symbolic location in the system is shown in (b). When user stays for a certain period of time, this system notifies a place where it is now for user, such as (c). After the location-awareness, in current location, the screen predicted when the user might depart in (d). When it comes to the predicted starting point for the user, predicting a destination and a path to reach the destination are shown as in (e).

A. Location recognition

In this system, location recognition module uses the methods of classifying indoor or outdoor, outdoor location recognition and indoor location recognition. To evaluate this module, the performance of each method is measured.

Figure 7 illustrates the performance of discriminating indoor or outdoor for the ten users. This experiment is evaluated by using naïve Bayes classifier with 10-fold cross-validation. In the experiment, accuracy shows 96.6% in average.

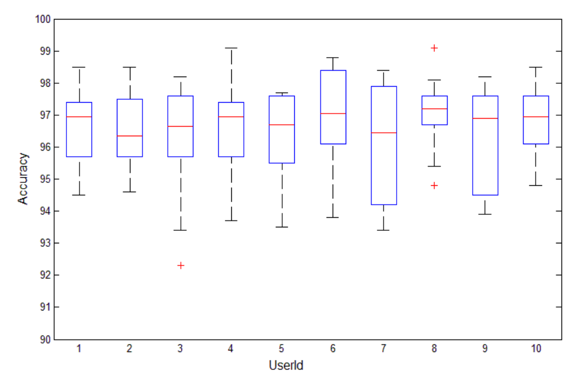


Figure 7. Performance evaluation of classifying indoor/outdoor

Figures 8 and 9 show the accuracy of outdoor and indoor location recognitions, respectively. These experiments are conducted by 10-fold cross-validation. Outdoor location recognition result of the *k*-NN method shows average accuracy of 98.96%. However, in case of indoor location recognition, we obtain 95.36% of average accuracy, which is relatively low.

B. Prediction of departure time

To identify the prediction of departure time can offer new input at appropriate time for user. In this system, we evaluate

 ISBN: 978-1-61208-353-7

the accuracy for the ten users. The average accuracy results in 80.6% as shown in Figure 10. This system has high usability by customizing user's own moving time.

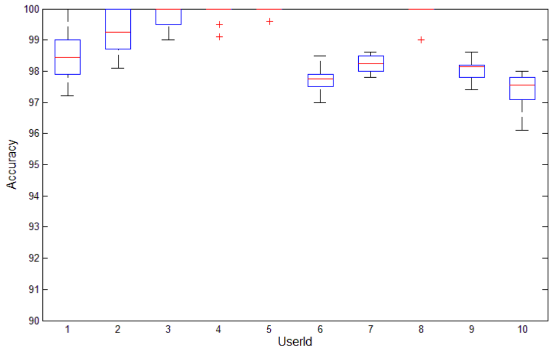


Figure 8. Performance evaluation of outdoor location recognition

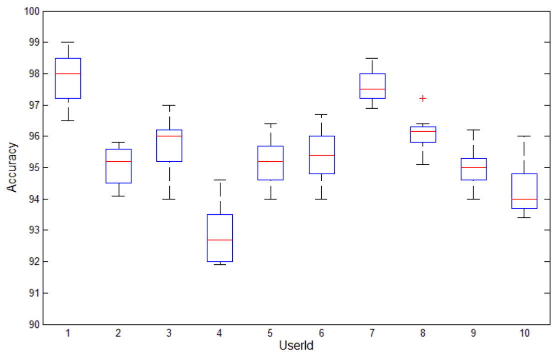


Figure 9. Performance evaluation of indoor location recognition

C. *Prediction of desitinaion*

In order to evaluate the accuracy of the proposed detination prediction method, we measured accuracy according to the path of progress. Prediction result of the advancement of the path is illustrated in Figure 11. It is a result that is preformed by 10-fold cross-validation. We trained the path models using 90% paths and measured accuraccy using 10% paths. Looking at the prediction accuracy in accordance with the progress of the path, as the path is largely moves, it can be seen that the prediction accuracy becomes higher because the information of the location movement is increased. For 0% progression of the path, which is capable of predicting only location information from the starting place, HMM showed accuracy of 57.96% in average only with the information of departure.

IV. Concluding Remarks

In this paper, we have proposed a system to manage location information and predict user's destination with outputs of modules using smartphone sensors such as GPS, Wi-Fi, acceleration, orientation, and magnetic field. The proposed system consists of location extraction, transportation type recognition, location recognition, prediction of moving time and prediction of destination. Destination prediction preforms calculating the similarity between new path and probabilistic model which contains information of the path with different modules, and finds a user's destination. Experimental results with real data collected from ten people show the usefulness of the proposed system.

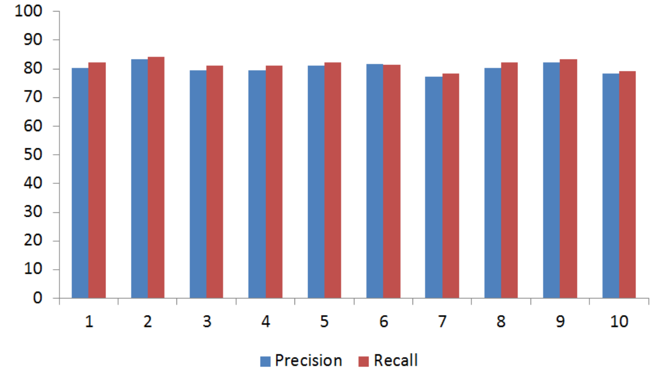


Figure 10. Performance evaluation of departure time prediction

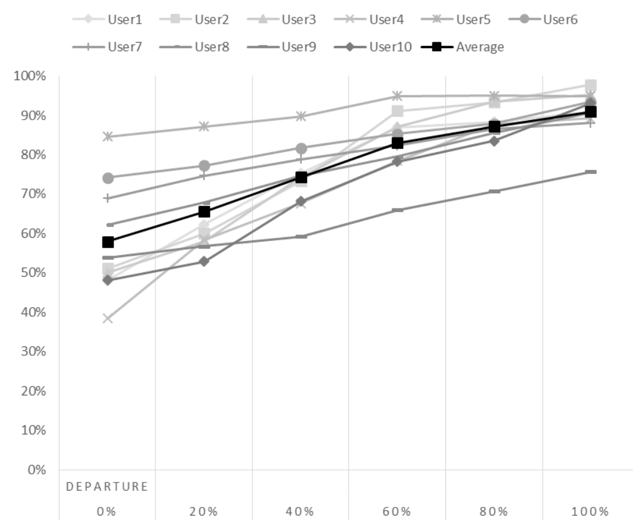


Figure 11. Performance evaluation of destination prediction

Acknowledgements

This work was supported by Samsung Electronics, Inc.

References

[1] M. Butler, "Android: Changing the Mobile Landscape," IEEE Pervasive Computing, vol. 10, no. 1, 2011, pp. 4-7.

[2] Y. Chon and H. Cha, "LifeMap: A Smartphone based context provider for Location-based Services," IEEE Pervasive Computing, vol. 10, no. 2, 2011, pp. 58-67.

[3] S. Bell, W. R. Jung, and V. Krishnakumar, "WiFi-based enhanced positioning systems: Accuracy through mapping, calibration, and classification," Proc. of the 2nd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, 2010, pp. 3-9.

[4] N. Brouwers and M. Woehrle, "Dwelling in the canyons: Dwelling detection in urban environments using GPS, Wi-Fi, and jeolocation," Pervasive and Mobile Computing, 2012, pp. 1-16.

[5] T. M. T. Do and D. Gatica-Perez, "Contextual conditional models for smartphone-based human mobility prediction,"

Proceedings of the 2012 ACM Conference on Ubiquitous Computing, 2012, pp. 163-172.

[6] Z. Lu, Y. Zhu, V. W. Zheng, and Q. Yang, , “Next Place Prediction by Learning with Multiple Models,” In Proceedings of the Mobile Data Challenge Workshop, 2012.

[7] B. Kim, J. Y. Ha, S. Lee, S. Kang, and Y. Lee, “AdNext: A Visit-Pattern-Aware Mobile Advertising System for Urban Commercial Complexes,” In Proceedings of the 12th Workshop on Mobile Computing Systems and Applications, 2011, pp. 7-12.

[8] S. Gambs, M. O. Killijian, and M. N. del Prado Cortez, “Next place prediction using mobility Markov chains,” In Proceedings of the 1st Workshop on Measurement, Privacy, and Mobility, 2012, pp. 1-6.

[9] L. Liao, D. Fox, and H. Kautz, “Learning and Inferring Transportation Routines,” In Proceedings of the National Conference on Artificial Intelligence, pp. 348-353, 2004.

[10] R. Simmons, B. Browning, Y. Zhang, and V. Sadekar, “Learning to Predict Driver Route and Destination Intent,” In Proceedings of the IEEE Intelligent Transportation Systems Conference, 2006, pp. 127-132.

[11] A. J. Dou *et al*, "Data clustering on a network of mobile smartphones," IEEE/IPSJ Symposium on Applications and the Internet , 2011, pp. 118-127.

[12] M. Kim, D. Kotz, and S. Kim, "Extracting a mobility model from real user traces," in Proc. IEEE Infocom, 2006, pp. 1-13,.

[13] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A. Campbell, "Nextplace: A spatio-temporal prediction framework for pervasive systems," Pervasive Computing, 2011, pp. 152-169.

[14] G. Hamerly and C. Elkan, “Learning the k in k means,” Advances in Neural Information Processing Systems, vol. 16, 2004, pp. 281.

[15] J.-K. Min and S.-B. Cho, “Mobile Human Network Management and Recommendation by Probabilistic Social Mining,” IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 41, no. 3, 2011, pp. 761-771.

[16] Y.-S. Lee and S.-B. Cho, "An Efficient Energy Management System for Android Phone Using Bayesian Networks," Distributed Computing Systems Workshops (ICDCS), 2012, pp. 102-107.

[17] I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone, "GPS/HPS-and Wi-Fi Fingerprint-Based Location Recognition for Check-In Applications Over Smartphones in Cloud-Based LBSs," IEEE Transactions on Multimedia, vol. 15, no. 4, 2013, pp. 858-869.

[18] D. H. Kim, J. Hightower, R. Govindan, and D. Estrin, “Discovering semantically meaningful places from pervasive RF-Beacons, ” Proc. of Int. Conf. on Ubiquitous computing, 2009, pp. 21-30.

[19] Y. J. Kim and S.-B. Cho, “A HMM-based location prediction framework with location recognizer combining k-nearest neighbor and multiple decision trees,” 8th International Conference on Hybrid Artificial Intelligent Systems, 2013, pp. 618-628.

[20] L. E. Baum, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," Ann. Math. Statist, vol. 41, 1970, pp.164 –171.

[21] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proc. IEEE, vol. 77, 1989, pp. 257–286.

[22] http://developer.android.com/tools/sdk/ndk/index.html: July, 2014.