

國立臺灣大學電機資訊學院資訊工程學系

碩士論文

Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis



指導教授：許永真 博士

Advisor: Jane Yung-jen Hsu, Ph.D.

中華民國 98 年 6 月

June 2009



國立臺灣大學碩士學位論文
口試委員會審定書

由個人移動軌跡紀錄分析交通模式與活動據點

Mining Transportation Modes and Significant Places from
Individual GPS Trajectories

本論文係張翰文君（學號 R96922005）在國立臺灣大學資訊工程學系完成之碩士學位論文，於民國 98 年 6 月 12 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

許永真

（指導教授）

陳鈞甲

朱浩等

陳穎平

王偉智

呂育道

系主任



Acknowledgments

完成一篇碩士論文，除了需要自己的熱忱與努力一路支持，也仰賴這段期間來自各方的鼓勵與協助。

感謝我的指導教授許永真老師，在我遇到研究瓶頸時指點迷津，提供後續研究方向的靈感與各種資源；在我鑽牛角尖而情緒不穩定時給予鼓勵與支持，增加我的自信心，提醒我專注於眼前重要的事。感謝友邁科技卓政宏董事長在我大四專題大眾運輸路線規劃系統時提供協助，開啓我對適地性服務（Location-based Services）的基本認識。感謝與資策會合作的計程車載客熱點分析研究案，讓我學會許多空間資訊分析的方法，並且應用於我的碩士論文中。感謝中研院許鈞南教授、台大朱浩華教授、王傑智教授，以及交大陳穎平教授擔任我的口試委員，於口試過程中給我許多寶貴的建議。

感謝婉容學姊長久以來的幫忙，不管是在學術研究、實驗室事務、或是一般行政業務上都給我許多協助。感謝于晉與元翔兩位實驗室電腦管理者，在我遇到電腦操作問題時幫忙我解決，使我可以快速地排除困難，回到工作中。感謝益庭學長、祖佑學長與家峻學長，教我使用CRF++與LibSVM兩套工具，使我可以快速上手；感謝冠鑒常常幫我訂正英文，還分享了許多寫網頁程式的小技巧。感謝研究小組的夥伴們，包括啓嘉學長、于晉、筱薇、元翔、世強、谷原與俊豪，在小組分享時給予很多意見與建議，尤其是大力相挺幫我看論文的

世強，謝謝你。也感謝所有幫我收集與標記軌跡資料的同學們，包括映嫻、薇蓉、皓遠、于晉、筱薇、世強、谷原、孟傑、俊豪、守壹與彥伶（本排序與論文內使用者順序無關，無需擔心洩露身分）。

感謝從大一開始就一直很幫我的忠毅，謝謝你總是很樂意陪我解決各種我不知道如何解決的問題，不管是課業上或是生活上的；也謝謝你的包容，雖然我有時候會情緒不好對你不禮貌，你也沒有對我生氣過，真是對不起。感謝碩二這一年來，常常陪我從系館走回宿舍，還要一路聽我聒噪的培堯學長，謝謝你，讓我有人可以分享任何突如其來的想法與心情，減少很多一個人胡思亂想的機會。謝謝這兩年一起生活的實驗室夥伴們，包括文傑學長、能豪學長、伍妮學姊、鶴齡學長、嘉涓學姊、怡靜學姊、琮傑、庭媽、中川、育誠...等等，大家平常一起聚餐、聊天、分享生活經驗，為研究生活增添了許多樂趣。謝謝我的好朋友們，包括家旗、婉婷、謹譽、厚達、佳穎、鴻昇、俊甫、宗哲、昱豪學長...等等，在我情緒低落時給我打氣加油，幫我增加信心與動力，度過消沉的日子。

謝謝我的好室友們，包括育昇、上傑、朝祺、育潞、品睿、宇傑；也謝謝我最愛的家人與親友們。謝謝你們的關心與耳提面命，每周提醒我要早睡早起，要盡早把該做的事完成。即使你們仍有些不放心，但你們鼓勵並要求我自己做決定，並且盡可能支持我的決定，謝謝。

謝謝身邊所有的人，與你們每個人的互動，都影響了今天的我。

Abstract

The whereabouts of a person not only indicates her schedule, but also reflects her lifestyle. The transportation taken and the places visited indicate the habit and preference of the user. With the growing popularity of commercial GPS loggers and GPS-enabled mobile phones, the positions of a person could be obtained and logged, and further analyzed to infer the transportation taken and places visited. Moreover, some places are more significant than others in one's daily life. These significant places shapes the life of the person.

In this thesis, we created a prototype of a trajectory management service to annotate and visualize the trajectories. We adopted machine learning techniques to segment the trajectories and extract their features, and used supervised learning approach to train probabilistic models. We modeled the transportation mode learning problem as a sequence labeling problem using linear-chain conditional random fields (CRF). We compared the CRF model with support vector machines (SVM), and our results show that CRF outperforms SVM, when temporal relationship is considered.

In addition, we adopted OPTICS clustering to find the places visited by the user. Results show that, among ten measures we used, visit frequency and stay duration predict the most significant places more accurately.

Keywords: Location-based Service, Trajectory Analysis, Spatial Data Mining, Conditional Random Field, Clustering



摘要

個人行蹤除了受到行程安排的影響外，也顯示了一個人某個面向的生活型態，例如常用的交通模式與常去的地點可以顯示一個人的習慣與偏好。而有些地點在一個人的生活模式中扮演較重要的角色；藉由較重要的活動據點可以了解一個人的生活模式。受惠於近來附有全球定位系統功能手機的普及，個人行蹤軌跡的取得與紀錄更加容易。從軌跡紀錄中可以進一步分析其交通行為與停留地點，並找出重要的活動據點。

在本論文中，我們設計了一個軌跡管理服務網頁，提供使用者將紀錄下來的軌跡上傳，並於網頁上標記、管理、與檢視。將軌跡分段並擷取特徵值後，我們應用條件隨機場模型（Conditional Random Fields）於交通模式辨認，並與支援向量機模型（Support Vector Machine）比較其預測準確度。實驗結果顯示，條件隨機場模型因為考慮了時間關係，辨識準確度較支援向量機模型高。

此外，我們使用OPTICS 群聚演算法將使用者曾停留的位置群聚為活動據點，並比較十種可作為預測據點重要程度的特徵值，如造訪次數、停留時間等。在十種特徵值中，以造訪次數多寡與停留時間長短之排序最能將使用者認知中相對重要的據點優先辨識出來。

關鍵詞：適地性服務、軌跡分析、空間資料探勘、條件隨機場、資料叢集



Contents

Acknowledgments	iii
Abstract	v
List of Figures	xii
List of Tables	xiii
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Objective	3
1.3 Thesis Outline	3
Chapter 2 Background	5
2.1 Related Work	5
2.1.1 Location-Based Services	5
2.1.2 Transportation Mode Learning	9
2.1.3 Significant Location Mining	9

2.2	Related Technology	11
2.2.1	Support Vector Machine	11
2.2.2	Linear Conditional Random Fields	12
2.2.3	OPTICS	14
Chapter 3 GPS Trajectory Analysis		17
3.1	Problem Definition	17
3.1.1	Location-Transportation Sequence	18
3.1.2	Significant Place Set	19
3.2	Notations	19
3.3	Proposed Solution	22
Chapter 4 Location-transportation Sequence		25
4.1	Preprocessing	26
4.1.1	Trajectory Segmentation	26
4.1.2	Feature Extraction	29
4.2	Classification and Sequence Labeling	32
4.2.1	SVM	33
4.2.2	LCRF	33
4.3	Performance Measures	36
Chapter 5 Significant Place Set		39
5.1	Location Clustering	39

5.2	Significance Estimation	40
5.2.1	Feature Selection	40
5.3	Performance Measures	43
5.3.1	Precision and Recall	44
5.3.2	Significance Ordering	44
5.3.3	NDCG	46
Chapter 6 Experiment and Evaluation		47
6.1	The Dataset	47
6.1.1	Data Collection	47
6.1.2	Data Annotation	48
6.2	Transportation Mode Learning	51
6.2.1	Experiment Steps	51
6.2.2	Example Result	52
6.3	Significant Location Mining	57
6.3.1	Experiment Steps	57
6.3.2	Example Result	59
Chapter 7 Conclusion		63
Bibliography		65

List of Figures

1.1	A trajectory example with annotated transportation modes.	2
2.1	An example of separating hyperplane learned by SVM.	12
2.2	An example of linear-chain CRF structure.	13
3.1	The proposed solution flow.	23
4.1	Trajectory segmentation process.	27
4.2	The CRF model used in this thesis.	34
5.1	Example result of significant place set mining.	43
5.2	Example of NDCG calculation.	46
6.1	GPS loggers used in the experiments.	48
6.2	A screen shot of the off-line annotation web page.	50
6.3	A screen shot of the significant place annotation web page.	50
6.4	Process of dividing trajectories into 5 folds.	52
6.5	The ground truth and inferred significant places of user11.	60
6.6	Average NDCG values of ten measures.	61

List of Tables

4.1	Features extracted from trajectory segments.	30
5.1	Features describing the visits of a place p	41
6.1	Statistics of dataset.	49
6.2	<i>Accuracy per segment</i> of 5-fold cross validation using CRF.	54
6.3	<i>Accuracy per segment</i> of 5-fold cross validation using SVM.	54
6.4	<i>Accuracy per log</i> of 5-fold cross validation using CRF.	54
6.5	<i>Accuracy per segment</i> of cross subject validation using CRF.	55
6.6	<i>Accuracy per segment</i> of cross subject validation using SVM.	55
6.7	<i>Accuracy per log</i> of cross subject validation using CRF.	55
6.8	Confusion matrix of 5-fold cross-validation using uniform length segmentation with 100 meters and CRF model on trajectories of user11.	56
6.9	Confusion matrix of 5-fold cross-validation using uniform duration segmentation with 60 seconds and CRF model on trajectories of user11.	56
6.10	Precision and recall of inferred significant places using the parameters ($Eps = 1000$ meters, $MinPts = 2$, $\xi = 0.3$, $\zeta = 100$ meters, $\alpha = 0$)	61

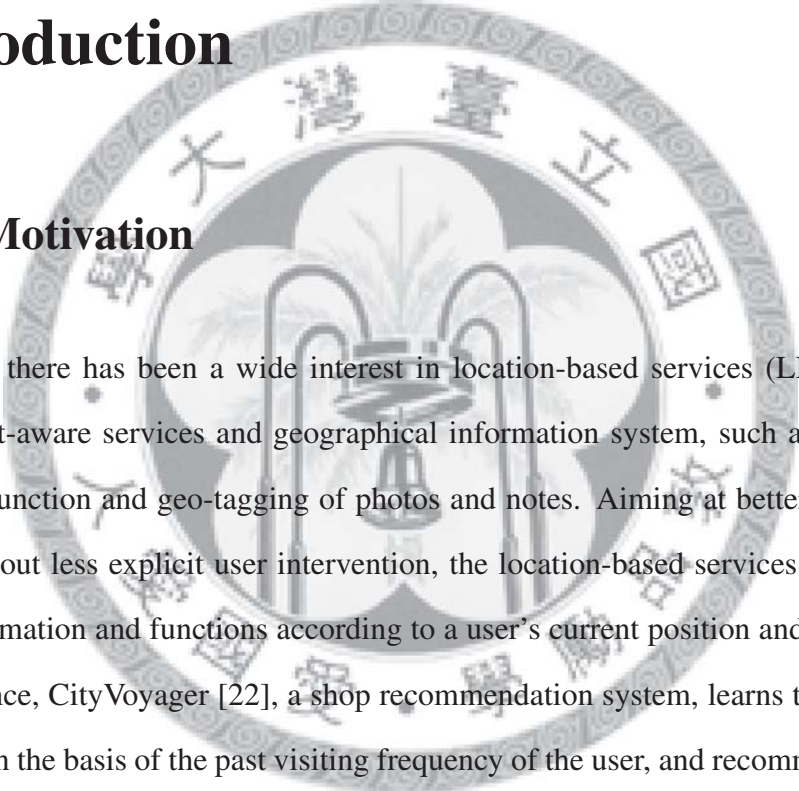
6.11 Strict Ordering Accuracy (OA_{str}) of the inferred significant places using exactly one of the ten features. 61



Chapter 1

Introduction

1.1 Motivation



Recently, there has been a wide interest in location-based services (LBS), a subset of context-aware services and geographical information system, such as the “search nearby” function and geo-tagging of photos and notes. Aiming at better user experience without less explicit user intervention, the location-based services provide suitable information and functions according to a user’s current position and visit history. For instance, CityVoyager [22], a shop recommendation system, learns the shop preferences on the basis of the past visiting frequency of the user, and recommends similar shops close to the user’s current position. This adaptability enables the systems to provide personalized services and react to changes of a user’s location context.

One central component in location-based services is the location profile of the user. A location profile includes a set of identified locations, user’s visit behaviors among these locations, and some measures as a summary of the user’s mobility. It is com-

monly believed that everyone's location profile is unique and the visiting behaviors may follow some patterns. The location profile should be derived from the visit history of a user. In particular, the trajectories, sequences of logged user positions associated with the timestamps, are used as the data source. Figure 1.1 is one trajectory example. In this example, the user started the day by taking bus to one bus station and waited for the bus to National Taiwan University. After arriving at the university, the user walked to the parking lot and ride the bike to the building, the destination of the trajectory. To automatically extract the information like the description above from trajectories requires the identification of places and transportation modes. With the description, the services could obtain more information about the users and provide suitable services. Hence, trajectory analysis has become one of the most popular topics in the field.



Figure 1.1: A trajectory example with annotated transportation modes.

1.2 Objective

Most of the current analysis focus on absolute locations and transition intervals; however, it is important to consider the semantics of the locations and the visiting contexts. For example, people may go to Starbucks at different branch stores. The branch stores are often viewed as different locations in spatial domain, but they should be viewed as similar locations since they share the same semantic: coffee shops. With the semantic information, it is possible to generalize the trajectory patterns from “visiting branch A” to “visiting coffee shop.” On the other hand, people may go to the same place but at different time or under different weather conditions. The contexts affects the behavior of users and should be captured in the trajectory for analysis. In addition, visiting frequency should not be the only measure to profile the user and the location. Measures such as moving speeds, staying durations, and variety of whereabouts provide more information to profile a person and understand the behavior pattern of the person.

The main purpose of this research is to propose a more general and powerful trajectory model and the semantically meaningful and statistically significant measures of the trajectories so that the other location-based services may better understand the users via the summary of the trajectories.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. Chapter 2 summarizes the background and related work about location-based services and trajectory analysis researches. In Chapter 3, the problem formulation is presented and the proposed solution is briefly

described. Following the definition, Chapter 4 and Chapter 5 details the approaches to model the trajectories and profile users with the significant places. Next, the experiments, including the dataset and implementation, and results are explained in Chapter 6. Finally, concluding remarks and future directions are stated in Chapter 7.



Chapter 2

Background

2.1 Related Work

2.1.1 Location-Based Services

Location-based services are becoming more and more popular, and the research field of location-based services and user profiling attracts people's attention [8]. Common applications include pre-destination [2, 12, 34], life-logging and whereabouts generation [20, 3, 30], spatial pattern mining [25, 31, 27] trajectory-based user-user similarity [16] and other assist utilities. Transportation mode learning [17, 29] and significance place mining [33, 28, 19] are the bases of these applications to provide personalized services. Knowing the places and transitions among them, service providers could help users summarize their trips, prepare related information to their next destination, and suggest similar users to form communities and social networks. Before extracting information from location data, location models and trajectory models are necessary

to represent the trajectories users taken. The following paragraphs summarize related researches toward modeling the location and trajectory and learning the transportation modes and significant places.

Location Models

In the field of geographic information systems, data are separated into two categories: the geographic feature and the attributes. The geographic feature of a location includes the reference coordinate system and its geometry. Each location may be associated with several attributes, including the syntactic labels like names and addresses and numerical measurements like area and height.

Open Geospatial Consortium, Inc. (OGC)¹ defines the OpenGIS® Simple Features Interface Standard (SFS)² to represent geographical features in relational databases such as MySQL³ and PostgreSQL⁴. The two most common open source databases both support storing, retrieving, and manipulating the geographic features.

In order to operate the locations in the information systems, Ye et al. [26] proposed a general spatial model for representing the locations. This model allows both syntactic and semantic labels on a location, and provides both absolute and relative references for geographic positions. In this model, containment relationships are organized in a lattice model and connection relationships a graph model.

¹<http://www.opengeospatial.org/>

²<http://www.opengeospatial.org/standards/sfa>

³<http://dev.mysql.com/doc/refman/5.0/en/spatial-extensions.html>

⁴<http://postgis.refractor.net/>

Trajectory Models

Built on location models, trajectories connect locations with additional temporal dimension. Giannotti et al. [10] proposed the spatio-temporal annotation sequence to represent the visiting sequences and relative transition time between consecutive stops. In this representation, the sequential order and the duration of transition provide more features to estimate the similarity of two trajectories and detect the moving patterns. However, this representation ignores the absolute time context and lacks the expressions about the moves between stops. Brosset et al. [4] proposed using an ordered sequence of route segments as location-action-location triples to represent the entire trajectory. A sample triple is “from the main gate, walk east, and arrive at the library.” Each action is further characterized by its cardinal orientation (walk *east*), relative direction (turn *left*), and elevation direction (go *downstairs*). Kulyukin et al. [14] focused on verbal descriptions of routes, and segmented route descriptions into environment features, delimiters, verb of movements, and state-of-being statements. Verbs of movements are like actions and transportation mode, while the other three components are mainly for characterizing the reference locations. These models provide us the direction to formalize trajectories, but the approaches to automatically derive these descriptions are not well-established.

Representation Specifications

While there are still researches investigating more general location models, some specifications already exist to meet the requirement for different applications. Different representations are used by different sensor devices and applications for different pur-

poses. For exchanging GPS logs among different hardwares and softwares, some popular specifications are commonly used as the interfaces, including NMEA 0183, GML, KML, and GPX.

NMEA (NMEA 0183)⁵ is an interface standard defined by U.S.-based National Marine Electronics Association. This standard defines the communication protocol and sentence formats between marine electronic devices; GPS receiver is one of the devices using this standard. This specification is commonly used for real-time data transferring among Bluetooth GPS receivers and other portable devices, such as mobile phones, but for most off-line services, like trajectory pattern mining, NMEA 0183 is not the most preferred one.

GPX (GPS Exchange Format)⁶ is a light-weight data format for exchanging GPS data among GPS devices, applications, and web services. Initially released in 2002 on the basis of XML standard with capability of extensions, GPX has been adopted by several programs and services, including most of Garmin GPS devices. The main components of a GPX file are metadata, waypoints, routes, and tracks.

GML (Geography Markup Language)⁷ is an XML tag-based Open Geospatial Consortium (OGC) standard for expressing geographical features. GML serves as a modeling language for geographic systems, and it can describe the geographic data in the form of points, polylines and polygons.

KML (Keyhole Markup Language)⁸ serves as an XML tag-based standard for express-

⁵http://www.nmea.org/content/nmea_standards/nmea_083_v_301.asp

⁶<http://www.topografix.com/gpx.asp>

⁷<http://www.opengeospatial.org/standards/gml>

⁸<http://www.opengeospatial.org/standards/kml>

ing geographic annotation and visualization. KML follows the GML specification to describe the geographic shape; in addition, it specifies the metadata annotation and visualization properties for web services like Google Maps. KML is now a standard of Open Geospatial Consortium (OGC).

2.1.2 Transportation Mode Learning

Among the research fields about location-based services, transportation mode learning is becoming an interesting topic. Liao et al. [17] used hierarchical conditional random fields, a discriminative probabilistic model in machine learning, to label the most likely activities and places associated to the GPS trace. Besides statistical features of the velocity, they considered contextual features such as the proximity to known landmarks and the temporal information at the time (*morning, afternoon, etc.*) Zheng et al. [29] found three other features which may increase the inference performance: pause rate, velocity change rate, and heading change rate. They observed that in different transportation mode, the frequencies of being in slow speed, of changing the velocity, and of changing the heading direction are different. For example, when a person walks along a street, more pauses may be detected and the heading change rate may be larger than driving or biking.

2.1.3 Significant Location Mining

To tell apart the stops from moving trajectories collected by the GPS signal, a number of researchers have developed different approaches. Marmasse and Schmandt [18]

observed that most buildings are GPS opaque, thus the loss of the GPS signal for a period of time can be used to infer indoor locations. Ashbrook and Starner [2] further adapted the K-means clustering method to group individually detected nearby locations into one representative location, compensating for the redundancy caused by the imprecision of the GPS signal when the user enters the same building several times. To identify the outdoor locations where a user stays for a long period of time, Zhou et al. [33] applied a density-based clustering method, DB-SCAN, on the trajectory to find the locations. Compared with to K-means, DB-SCAN discovers clusters of arbitrary shapes, and is less sensitive to noises and outliers. Zhang et al. [28] later indicated that these off-line methods rely on the data collected in advance to extract locations. In [28], Zhang et al. modeled the locations as a mixture of Gaussians, and proposed an on-line learning method to dynamically update the parameters of the mixed Gaussian distributions. This formulation reduces the response time to extract the latest whereabouts of an user, but at the same time loses the precise location of the significant places. In [17], Liao et al. not only learns the transportation mode but some visiting behaviours to detect locations such as home, offices, and parking lots.

However, not all places play the same role in one's daily life. Some are more important and attract the user to visit more often. As a result, some researchers proposed different measures to estimate the relative importance of places. Zhou et al. [32] partitioned the locations into four categories: important and frequent, important but infrequent, unimportant but frequent, and unimportant and infrequent. They used the total visit count, number of unique days visiting, total record count (including pass-by records), and number of unique days having records as features and apply K-NN (K-

nearest neighbors) to classify the locations. In [9], Froehlich et al. hypothesized that the visiting frequency and travel effort of a person to one location reflects the importance of the location and the level of interest the person to the location. From the idea of PageRank in estimating the importance of an URL, Sabbata et al. [21] proposed the SpaceRank, a similar mechanism to PageRank, to model the possibilities of visiting one place on the basis of the adjacency of locations and past movements of the users. They consider the geographical properties of locations and history records of users to form the SpaceRank matrix and simulate random walk on the Markov chain with the SpaceRank matrix. The dominant eigenvector decides the relative importance of the places.

2.2 Related Technology

2.2.1 Support Vector Machine

Support Vector Machines (SVMs) are a set of supervised learning methods commonly used for classification problems. Viewing data of two classes as two sets of vectors in an n -dimensional space, an SVM constructs a *separating hyperplane* in the space, which maximizes the margin between the two sets of vectors. With additional kernel functions, an SVM may transform the data points in the n -dimensional space to a higher dimensional space. The transformation may be non-linear; thus though the classifier is a hyperplane in the high-dimensional feature space it may be non-linear in the original input space. Radial basis function is a commonly used kernel for SVM model. Figure 2.1(a) is an example with data points of 2 classes in a 2-dimensional space, and

Figure 2.1(b) shows a separating hyperplane which may be used to distinguish the 2 classes.

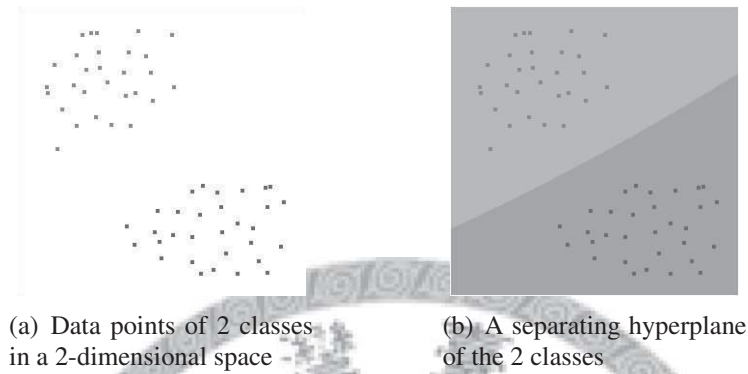


Figure 2.1: An example of separating hyperplane learned by SVM. This hyperplane can distinguish two classes in the n -dimensional space.

2.2.2 Linear Conditional Random Fields

Conditional Random Fields (CRFs) [15] provide a probabilistic framework for labeling structured data. In particular, linear CRFs have been extensively applied to sequence labeling problems in many fields. Unlike generative models which capture the joint probability of labels and observations, CRFs model the conditional probability distribution over labels given one particular set of observations. With the conditional nature, CRFs result in the relaxation of the independence assumptions required by HMMs. Additionally, CRFs avoid the label bias problem exhibited by conditional Markov models based on directed graphical models.

To define the CRF model, let $G = (V, E)$ be an undirected graph structure consisting of two vertex sets $X \subseteq V$ and $Y \subseteq V$, where X represents the vertex set of

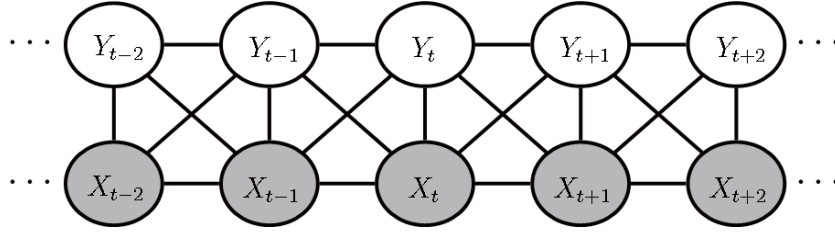


Figure 2.2: An example of linear-chain CRF structure. In this structure, the hidden nodes Y are connected as a chain and each Y_t is connected to the previous, current, and next observations.

observation nodes and Y represents the vertex set of hidden nodes. Figure 2.2.2 is an example of CRF graph structure, in which the the hidden nodes Y are connected as a chain and each Y_t is connected to the previous, current, and next observations. This kind of structure is commonly seen in sequence labeling problems. Let C be the set of the fully connected subgraphs, called *cliques*, in a CRF, where each clique c is composed of vertices $X_c \subseteq X$ and $Y_c \subseteq Y$. Then, a CRF factorizes the conditional probability distribution into a product of non-negative *clique potential functions* $\Phi_c(X_c, Y_c)$. Clique potential functions map the variable configuration to a real number to capture the compatibility among the variables; the higher the potential function value, the more likely the configuration of the hidden nodes and the observations. Using potential functions, the conditional probability distribution over the hidden nodes given the observations is written as

$$P(Y|X = x) = \frac{1}{Z(x)} \prod_{c \in C} \Phi_c(x_c, Y_c) \quad (2.1)$$

where x denotes the observation values assigned to X and $Z(x) = \sum_y \prod_{c \in C} \Phi_c(x_c, y_c)$ is the normalization term.

Without loss of generality, the *potential functions* $\Phi_c(X_c, Y_c)$ can be described by

log-linear combinations of *feature functions* $\mathbf{f}_c(X_c, Y_c)$. That is,

$$\Phi_c(x_c, Y_c) = \exp \{ \mathbf{w}_c^T \mathbf{f}_c(x_c, Y_c) \} \quad (2.2)$$

where \mathbf{w}_c^T is the transpose of a weight vector. Combining Equation 2.1 and 2.2, the conditional probability distribution could be expressed as the following form.

$$P(Y|X = x) = \frac{1}{Z(x)} \exp \sum_{c \in C} \mathbf{w}_c^T \mathbf{f}_c(x_c, Y_c) \quad (2.3)$$

Suppose the feature functions are pre-determined. The learning phase of CRF model is to estimate the weight vector \mathbf{w} which maximize the conditional probability $P(y|x)$ given the training data with labels y and observations x , while the predicting phase of CRF is to find the probability distribution of $P(Y|x, w)$.

2.2.3 OPTICS

OPTICS (Ordering Points To Identify the Clustering Structure) [1] is a density-based clustering algorithm. Modified from DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [6], OPTICS not only finds the dense parts the data point distributed but also provides a mechanism to derive the hierarchy of clusters at different level of granularity. In DBSCAN, a spatial distance threshold ϵ is used to define the proximity of two points. If the number of proximity of a specific point exceeds a predefined parameter $MinPts$, the point is regarded as in the core of one cluster, and its proximity belongs to the same cluster it is in. Adapting from DBSCAN, OPTICS defines the *core-distance* and *reachability distance* for each data point. The *core-distance* of a data point p is the distance to its $(MinPts - 1)$ -th neighbor. In the

other word, the *core-distance* records the minimum distance threshold under which the data point is a core object. If the distance to its $(MinPts - 1)$ -th neighbor is larger than the given threshold ϵ , the *core-distance* is set to *UNDEFINED*. The *reachability-distance* of a data point p with respect to another data point o is the smallest distance threshold such that p is directly reachable from o if o is a core object. During the clustering process, only the smallest *reachability-distance* is recorded for generating the order and determining the boundaries of clusters.

With the two definition, OPTICS generates the order of points by incrementally picking a data point, calculating the *core-distance*, updating the *reachability-distance* of unprocessed data objects, and picking the next data point with smallest *reachability-distance* to be processed. The algorithm is shown in Algorithm 1 and 2.

After generating the order, by plotting the *reachability-distance* along the order of data points to form the *reachability-plot*, the hierarchical cluster structure can be obtained easily. Since points belonging to a cluster have a low reachability distance to their nearest neighbor, the clusters show up as valleys in the reachability plot. However, setting different thresholds may extracted clusters with different boundaries. An automatical way to determine the boundary of clusters is to detect the difference on *reachability-distance* between consecutive data points. If the difference is greater than $\xi\%$, the points may be the local boundary to cut the clusters.

Algorithm 1 OPTICS(*objects*, ϵ , *MinPts*)

inputs: *objects*, the set of data points
 ϵ , the maximum distance threshold
MinPts, the minimum size to be called dense

returns: *order*, the ordered objects

- 1: *order* \leftarrow $\{\}$
- 2: **for** $i = 0$ to $\text{sizeof}(\text{objects})$ **do**
- 3: *object* \leftarrow *objects*(i)
- 4: **if** *object* has not been processed **then**
- 5: *object.reachability-distance* \leftarrow UNDEFINED
- 6: ExpandOrder(*objects*, *object*, ϵ , *MinPts*, *order*)
- 7: **end if**
- 8: **end for**
- 9: **return** *order*

Algorithm 2 ExpandOrder(*objects*, *object*, ϵ , *MinPts*, *order*)

inputs: *objects*, the set of data points
object, the data point to be expanded
 ϵ , the maximum distance threshold
MinPts, the minimum size to be called dense
order, the set to records the order

- 1: *object.processed* = *true*
- 2: *neighbors* \leftarrow neighbors(*objects*, *object*, ϵ)
- 3: SetCoreDistance(*object*, *neighbors*, ϵ , *MinPts*)
- 4: *order.append*(*object*)
- 5: **if** *object.core-distance* \neq UNDEFINED **then**
- 6: UpdateReachabilityDistance(*object*, *neighbors*)
- 7: **while** exists unprocessed data points with real-value *reachability-distance* **do**
- 8: *next* \leftarrow the unprocessed data point with smallest *reachability-distance*
- 9: ExpandOrder(*objects*, *next*, ϵ , *MinPts*, *order*)
- 10: **end while**
- 11: **end if**

Chapter 3

GPS Trajectory Analysis

3.1 Problem Definition

With advanced location tracking technologies using global satellites and local wireless access points, the accuracy and precision of positioning are getting better and better. In addition, with the growing popularity of GPS-enabled mobile devices for general people, users can locate themselves on the map and enjoy location-based services such as real-time navigation and local POI search. Moreover, users can log their positions all the time for latter reference, such as travel route sharing and photo geo-tagging. Thus, personal daily trajectories are more available nowadays. However, these trajectories are mostly only used for visualizing the whereabouts on map services, and seldom location-based services consider the whole trajectory to provide services. In order to utilize the whole trajectory, a model to describe the trajectory is necessary. As a result, it brings out the two questions:

1. What information could be retrieved from the trajectories?

2. How to use the information to describe a person's trajectories?

Two main elements in the trajectories are the visiting locations and the transportation modes taken. With these information, the significant locations in one person's daily life and how the person move among them could be inferred. As a result, we proposed two models to describe a user's trajectories: **Location-Transportation Sequence** and **Significant Place Set**.

3.1.1 Location-Transportation Sequence

To utilize the trajectory data, an appropriate model is necessary for easy understanding and manipulation. Observing how people communicate and understand one's moving trajectory helps us define what are the important elements in one trajectory. In our experiences of having conversations with others about travel, there are two popular questions people care about: where the person has been and how the person travels. Similar concept has been proposed by Brosset et al. in [4] using location-action-location triples to represent a trajectory segment. They have shown that this model can fit several sentences people used in describing their trajectories. Hence, we propose a location-transportation sequence to describe where the person has been and how the person travels, and the technical problem is to generate the location-transportation sequence from the position logs.

3.1.2 Significant Place Set

After abstracting the trajectories into location-transportation sequences, a set of visited place could be obtained by union all the stops in the trajectories. Among these stops, some plays more important roles in the trajectory, such as home and office, while some are less meaningful, such as crossroads. For each place, some measures could be used as features, like the visit frequency and the length of staying duration, to estimate the significance of each place to the user. Generally speaking, significant places are those where user visits often and regularly and stays for a long period of time. With the estimated significances of each place, the importance order of places to the user could be retrieved and the most important places the user visits in daily lives could be used to describe the user.

3.2 Notations

Trajectory data are temporally-ordered logs of user position, which may be retrieved from tracking systems such as GPS. Positions from outdoor tracking devices are represented in the latitudes and longitudes.

Definition 1 Position Log

A position log $g = \langle t, \phi, \lambda \rangle$ is a record at time $t \in \mathbf{T}$ at latitude $\phi \in [-90, 90]$ and longitude $\lambda \in [-180, 180]$.

Definition 2 Trajectory

A trajectory $Traj(t_b, t_e)$ between time t_b and t_e is a temporally-ordered sequence of

position logs during the given duration.

$$\text{Traj}(t_b, t_e) = (g_1, g_2, \dots, g_n) \quad (3.1)$$

where $t_b \leq g_i.t < g_j.t \leq t_e$, for all $1 \leq i < j \leq n$.

A person may move on foot or with other transportations, such as bikes, cars, metros, and other vehicles. And at any particular time, a person can only move with one transportation mode. A transit happens when a person changes from one transportation mode to another.

Definition 3 Transportation Mode Function

Let the set of transportation modes be denoted as TM . The transportation mode function $f_{tm} : \mathbf{T} \rightarrow \text{TM}$ maps any particular time to one transportation mode.

Definition 4 Transit Event

A transit event (c) changes from transportation mode $tm_{c,b}$ to $tm_{c,e}$ during time $[t_{c,b}, t_{c,e}]$ if $f_{tm}(t_{c,b}) = tm_{c,b}$, $f_{tm}(t_{c,e}) = tm_{c,e}$, $tm_{c,b} \neq tm_{c,e}$ and there is no other logs except the boundary logs; that is, $|\text{Traj}(t_{c,b}, t_{c,e})| = 2$.

A place is a location or region where users may stay for a period of time for some purposes such as working, eating, having fun, resting, and other means. Homes, offices, restaurants, tourist spots are all kinds of places. The collection of all places are denoted as \mathbf{P} , and the instances could be obtained from existing commercial datasets or from user generated contents. At a particular time, a person can visit at most one place; that is, the person may not visiting any place if the user is just passing by the location without intention to make a visit.

Definition 5 Visiting Status

The visiting status of a user at a particular time is indicated by the place the user being visiting or a “not visiting” status if the user is just passing by the location without intention to make a visit. The mapping function is denoted as $f_{vs} : \mathbf{T} \rightarrow \mathbf{P} \cup \{\text{null}\}$

Definition 6 Stop Event

A stop event (s) is a trajectory within a time interval $[t_{s,b}, t_{s,e}]$ which satisfies the condition that there exist a place $p \in \mathbf{P}$ such that the visiting status during the whole time interval is p ; that is, $f_{vs}(t') = p$ for all $t_{s,b} \leq t' \leq t_{s,e}$. A transit event c is a special kind of stop event.

As a result, a trajectory could be simplified as a sequence of stop events and the transportation mode taken between two events. In this way, the trajectory is more understandable than the sequence of log points.

Definition 7 Location-Transportation Sequence

A Location-Transportation Sequence (LTS) of trajectory in $[t_b, t_e]$ is a concatenation of stop events and the transportation modes.

$$LTS(t_b, t_e) = (s_1, tm_1, s_2, tm_2, \dots, s_{m-1}, tm_{m-1}, s_m) \quad (3.2)$$

where $t_b \leq t_{s_i,b} < t_{s_i,e} < t_{s_j,b} < t_{s_j,e} \leq t_e$, for all $1 \leq i < j \leq m$, $f_{tm}(t') = tm_i$ if $t_{s_i,e} < t' < t_{s_{i+1},b}$ and $f_{vs}(t') = \text{null}$ if $t_{s_i,e} < t' < t_{s_{i+1},b}$.

Definition 8 Significant Place Set

A Significant Place Set of a person is a fuzzy set $SPS = (\mathbf{P}, Sig)$ where \mathbf{P} is the place

set and $Sig : \mathbf{P} \rightarrow [0, 1]$ is the significance function. An α -Significant Place Set is the α -cut of the fuzzy set which contains the places with significance value larger than α .

As a result, the task of significant place mining is to find the significance function of the person.

3.3 Proposed Solution

With the definitions above, the core problem to find the location-transportation sequence of a trajectory is to find the transportation mode function and visiting status function from the raw GPS logs. In other words, given the trajectory $Traj(t_b, t_e) = (g_1, g_2, \dots, g_n)$, the task is to find the corresponding functions f_{tm} and f_{vs} of transportation mode and visiting status.

Since one person can only have at most one record at a given time, it is impossible to directly learn f_{tm} and f_{vs} on time domain; instead, we extracted features from the observations $O : \mathbf{T} \rightarrow \mathbb{R}^l$ where l is the number of features, and learn the $f_{tm}^* : \mathbb{R}^l \rightarrow \mathbf{TM}$ and $f_{vs}^* : \mathbb{R}^l \rightarrow \mathbf{P} \cup \{null\}$.

Moreover, the assumption is made that when a person intend to visit one place, the person will approach the place on foot. With this assumption, we could combine f_{tm}^* and f_{vs}^* into one $f^* : \mathbb{R}^l \rightarrow \mathbf{TM} \cup \mathbf{P}$ as the following definition.

$$f^*(O_t) = \begin{cases} f_{vs}^*(O_t), & \text{if } f_{vs}^*(O_t) \neq null \\ f_{tm}^*(O_t), & \text{otherwise} \end{cases} \quad (3.3)$$

Then, we only have to learn one function instead of two.

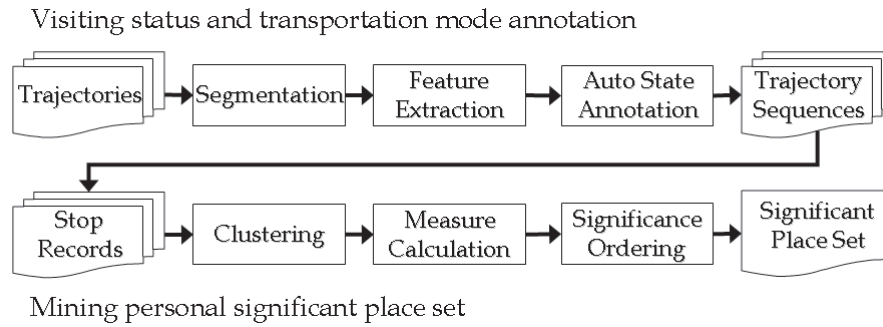


Figure 3.1: The proposed solution flow.

The learning process consists of segmentation, feature selection, and model learning, while the testing process uses the learned model to label the trajectory. As to segmentation, the trajectory can be chunked into segments in respect to unit time or unit distance. Features related to speed, heading direction, and other contextual information will be extracted for each segment. Support Vector Machine (SVM) and Linear Conditional Random Field (LCRF) models [15] could be used to train the corresponding classifier or labeler. Details of the transportation learning process are described in Chapter 4.

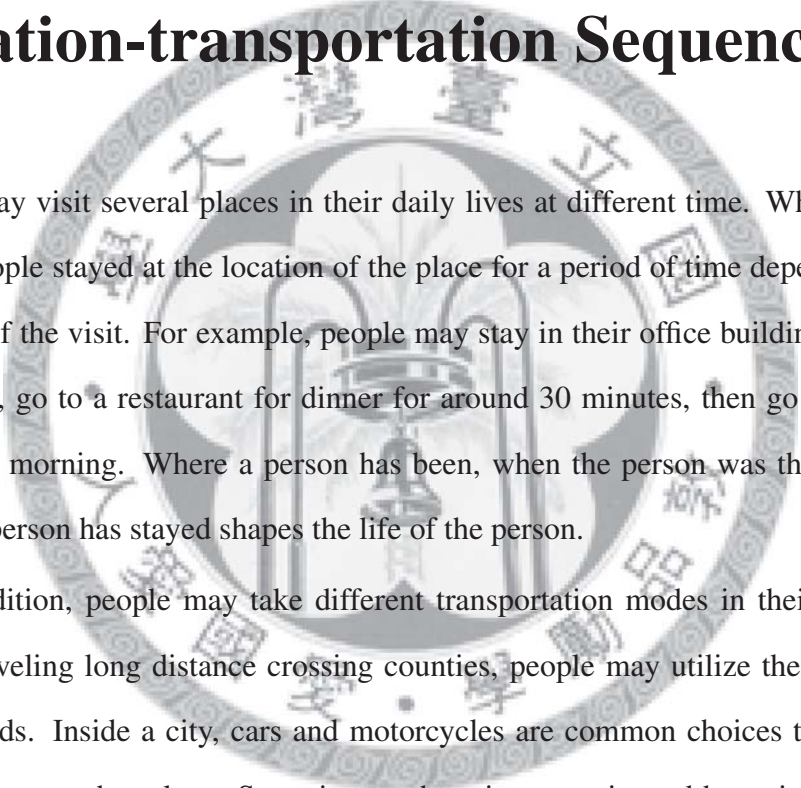
For the significant place set problem, we assume zero-significance to places the user has never been to. Then the domain of significant function shrinks to only the visited places of the user. If the place set P is pre-defined from existing sources, and user generated contents are not dynamically inserted into the set, the stop locations are associate to one place by choosing the nearest neighbor; if user generated contents are considered, clustering algorithm is applied on the stop locations to group nearby locations, and check if it is an existing place or a new place. After associating the stop events to places, measures such as the visit frequency, total stayed time, average stayed

time were calculated and the significances are derived from these measures. Details of the significance estimation are described in Chapter 5.



Chapter 4

Location-transportation Sequence



People may visit several places in their daily lives at different time. When visiting a place, people stayed at the location of the place for a period of time depending on the purpose of the visit. For example, people may stay in their office buildings for whole afternoon, go to a restaurant for dinner for around 30 minutes, then go home to rest until next morning. Where a person has been, when the person was there, and how long the person has stayed shapes the life of the person.

In addition, people may take different transportation modes in their daily lives. When traveling long distance crossing counties, people may utilize the rails or high speed roads. Inside a city, cars and motorcycles are common choices to move from one place to another place. Sometimes, when time permits and haste is unnecessary, people would ride bikes or walk on foot instead. Understanding the transportation mode people choose may give more information on how people feel the travel experience.

As defined in Equation 3.2, the location-transportation sequence, a concatenation

of stop events and transportation modes, could be used to summarize the trajectory during a time period. The sequence is constructed based on the output of the general mode function described in Equation 3.3. In this chapter, the problem of learning the transportation mode and visiting status from GPS trajectories is tackled.

4.1 Preprocessing

4.1.1 Trajectory Segmentation

Most of the time, people do not change transportation mode frequently; people will utilize the transportation taken for some amount of time and travel some amount of distance. As result, we can first partition the trajectories into segments, and learn the transportation mode for each segment instead of for each position log. The advantages of using segments instead of logs in learning are two-fold. On one hand, moving is a continuous behavior, and more features could be extracted from the segments to help judge the transportation mode. On the other hand, some position logs may be noisy due to uncontrollable reasons such as sensor errors, and the incorrect logs may cause inference error. By observing the whole segments and extracting statistics features, the effect of noisy logs may reduced. Figure 4.1.1 illustrates the concept of segmenting the positions logs into chunks and further forms the chain of segments.

Intuitively, there are two kinds of segmentation methods: uniform duration segmentation and uniform length segmentation. The assumption is that people do not change activities during a short period of time and a short travel distance. The uni-

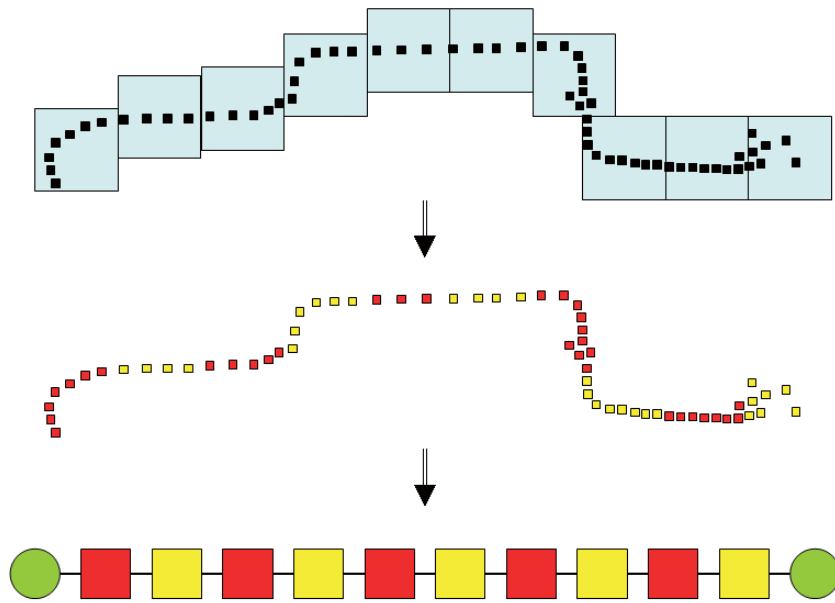


Figure 4.1: Trajectory segmentation process. Position logs are segmented into small chunks as the units in transportation mode and visiting status learning.

form duration segmentation partition the trajectories every θ_t seconds, and the uniform length segmentation partition the trajectories at every θ_d meters of moving. The difference of both methods is that uniform duration segmentation produces segments with the same duration, one minute for example, while the uniform length segmentation produces segments with the same length, 50 meters for example. Once segmented into several short trajectories, features could be extracted, and the general mode could be learned.

In addition to uniform duration segmentation and uniform length segmentation, we tried another grid-based segmentation approach to chunk the trajectories, which we called uniform grid segmentation. The concept behind the uniform grid segmentation is that people seldom change activities when they stay at a small region. The Algorithm

Algorithm 3 UniformGridSegmentation($trajectory, \theta_g$)

inputs: $trajectory$, a sequence of position logs
 θ_g , the length threshold of the grid side

returns: $segments$, a sequence of segments

- 1: $segments \leftarrow \{\}$
- 2: $segment \leftarrow \{\}$
- 3: **ResetBound**($bound, trajectory(0)$) {reset bounds according to first log}
- 4: **for** $i = 0$ to $sizeof(trajectory)$ **do**
- 5: $log \leftarrow trajectory(i)$
- 6: **UpdateBounds**($bound, log$) {extend bounds to cover log }
- 7: **if** $bounds.side > \theta_g$ **then**
- 8: $segments.append(segment)$
- 9: $segment \leftarrow \{\}$
- 10: **ResetBound**($bound, log$) {reset bounds according to log }
- 11: **end if**
- 12: $segment.append(log)$
- 13: **end for**
- 14: $segments.append(segment)$
- 15: **return** $segments$

3 describes the steps to generate segments by uniform grid segmentation. For each new segment, the bound of the grid is incrementally extended with the next position log until the side length is larger than a given threshold. Thus, the algorithm produces segments all within a square with side length θ_g .

In the training phase, we annotate each segment as the majority transportation mode or visiting status of the position logs containing in the segment. As a result, each segment has only one annotation representing the transportation mode and visiting status to remove the ambiguity at the boundary of different transportation mode and visiting status.

4.1.2 Feature Extraction

After partition the trajectory into segments, features including the duration and distance traveled, the average of instantaneous speed, the change of heading direction, and other measures of observed evidences. The features considered in this research are listed in Table 4.1, and details explanations are given in the following paragraphs.

A position log contains the information of time, latitude and longitude. While some loggers may record additional information such as instantaneous speed and heading direction (or *bearing*), some loggers do not have these data. As a result, we apply formulas to calculate distance and bearing between two positions and further derive the instantaneous speed. The most common distance measure of two points on the map is the Euclidean distance. However, the earth is roughly a great sphere, and the latitude and longitude are defined globally in respect to the earth surface instead of a plane; that means, the Euclidean distance is not an accurate measure and the scaling parameter depends on the latitude value. We modified Vincenty's formulas [23] to approximate the distance and heading direction between two positions when the information is not available from sensors. Vincenty's formulas [23] were designed to calculate the distance between two points on an ellipsoid. In this work, the formulas are simplified assuming that the Earth is a sphere with radius 6372.795 kilometers, and the geodesic distance between two points in meters is the radius times the angular distance $\Delta\hat{\sigma}$ which is given in the Equation 4.1; the heading direction is given in the Equation 4.2. In consequence, we estimate the value concerning the distance to the next position log as in the Equation 4.3. In addition, we calculate the speed change ratio using the Equation 4.4.

Table 4.1: Features extracted from trajectory segments.

Category	Notation	Description
Duration	Δt	The total duration traveled ($\Delta t = t_e - t_b$)
Distance	Δd	The total distance traveled ($\Delta d = \sum dist(g_i, g_{i+1})$)
Position	R_ϕ, R_λ	The range of latitude and longitude
	$\sigma_\phi, \sigma_\lambda$	The standard deviation of latitude and longitude
	$\sigma_{\phi,\lambda}$	The co-variance of latitude and longitude
	$\overline{L2C}$	The mean of L2C (length to center)
	σ_{L2C}	The standard deviation of L2C (length to center)
	$G_{2,L2C}$	The kurtosis of L2C
Speed	\bar{v}	The average speed ($\bar{v} = \Delta d / \Delta t$)
	$\mathbf{E}[v]$	The average of instantaneous speeds ($\mathbf{E}[v] = \sum v_i / n$)
	v_{max_i}	The i -th largest speed, $i = 1, 2, 3$
	v_{min_i}	The i -th smallest speed, $i = 1, 2, 3$
	v_{p_i}	The i -th percentile of speed $i = 10, 90$
	$SR(\alpha)$	The portion of speed below α over the duration
Speed Change	\bar{v}'	The average of instantaneous speed change
	$SCR(\beta)$	The portion of speed change greater than β percent
Direction Change	\bar{h}'	The average of heading direction change in degrees
	$DCR(\gamma)$	The portion of heading direction change greater than γ
Temporal Context	$\mathbf{T}_{Mor}([t_b, t_e])$	The membership of morning over the segment
	$\mathbf{T}_{Noo}([t_b, t_e])$	The membership of noon over the segment
	$\mathbf{T}_{Aft}([t_b, t_e])$	The membership of afternoon over the segment
	$\mathbf{T}_{Eve}([t_b, t_e])$	The membership of evening over the segment
	$\mathbf{T}_{Nig}([t_b, t_e])$	The membership of night over the segment
	$WD_i([t_b, t_e])$	Whether it is the i -th day of week. (Sun, Mon, ..., Sat)

$$\phi_i = g_i \cdot \phi$$

$$\phi_j = g_j \cdot \phi$$

$$\Delta\lambda = g_j \cdot \lambda - g_i \cdot \lambda$$

$$dist(g_i, g_j) = 6372795 \times \arctan(\vartheta_1(g_i, g_j)) \quad (4.1)$$

$$\vartheta_1(g_i, g_j) = \frac{\sqrt{(\cos \phi_j \sin(\Delta\lambda))^2 + (\cos \phi_i \sin \phi_j - \sin \phi_i \cos \phi_j \cos(\Delta\lambda))^2}}{\sin(\phi_i) \sin(\phi_j) + \cos(\phi_i) \cos(\phi_j) \cos(\Delta\lambda)}$$

$$heading(g_i, g_j) = \arctan(\vartheta_2(g_i, g_j)) \quad (4.2)$$

$$\vartheta_2(g_i, g_j) = \frac{\cos(\phi_j) \sin(\Delta\lambda)}{\cos(\phi_i) \sin(\phi_j) - \sin(\phi_i) \cos(\phi_j) \cos(\Delta\lambda)}$$

$$v_i = dist(g_i, g_{i+1}) / (g_{i+1}.t - g_i.t) \quad (4.3)$$

$$v'_i = |v_{i+1} - v_i| / v_i \quad (4.4)$$

$$h_i = heading(g_i, g_{i+1}) \quad (4.5)$$

$$h'_i = h_{i+1} - h_i \quad (4.6)$$

After estimating the speed and heading direction, we can extract features from the segments. For a trajectory segment during the interval $[t_b, t_e]$, the duration is the length of the interval in seconds ($\Delta t = t_e - t_b$) and the total distance is the summation of the distance between two consecutive position logs. The ranges and standard deviations of latitude and longitude are used to estimate the spatial span of the segment. The mean of latitude and longitude is regarded as the center of the segment, and $L2C$ is the length from the log position to the center. To measure whether the trajectory is on a straight line or nearly randomly around the center, the correlation coefficient and the Kurtosis of $L2C$ are used to capture this characteristic.

The average speed over the segment and the average of instantaneous speed are both used to represent the pace. In addition, we picked the largest three, the 9 decile, the smallest three, and the 1 decile of speed to estimate the range. Other statistical features about speed change and heading direction change are also used. Moreover, we adopted the stop rate (SR), speed change rate (SCR), and direction change rate (DCR) from [29].

Temporal contextual features are expected to enhance classification results. For each time period of a day, a discrete membership function is defined on the domain of seconds. For each position log, the membership function tells how likely the time belongs to the time period. For a trajectory with n logs, the membership value is defined as the average of the membership values for each log.

In total, there are 37 features extracted for each segments. Most statistic measures, including average, standard deviation, variance, and kurtosis, are calculated using the Apache Commons Math package [7].

4.2 Classification and Sequence Labeling

There are two different ways to solve the transportation mode learning problem. In one way, the segments could be viewed as independent instance given all the observed evidences. General classifiers such as Decision Tree and Support Vector Machine (SVM) could be used for inference. In the other way, trajectories are sequential data, and different transportation modes taken for previous segments may bring different transition probabilities for the following segments. From this viewpoint, the transportation mode

learning problem is a sequence labeling problem, and probabilistic models like Hidden Markov Chain (HMM) and linear chain Conditional Random Field (LCRF) [15] could be adopted to perform the inference. In this research, we compared SVM and LCRF in experiment, and we use libSVM [5] and CRF++ [13] as the inference tool.

4.2.1 SVM

Support vector machine (SVM) solves the classification problem by learning the separating hyperplanes in the n -dimension feature space. In this thesis, we viewed each segment as a vector in the 37-dimension feature space, where each dimension represents one feature listed in Table 4.1. We use libSVM [5] as the inference tool to classify these vectors. As the parameter selection, we used radial basis function as the basic function, and we tested 5 values, 0.1, 0.5, 1.0, 5.0, and 10.0, as the cost of misclassification penalty.

4.2.2 LCRF

A Conditional Random Field (CRF) uses an undirected graph to model the dependency structure among hidden nodes and the observation nodes, and learns the conditional probability distribution over the hidden nodes given the observations. In this thesis, we viewed each label of transportation mode and visiting status of segments as the hidden nodes, and the representing features as the observations. Figure 4.2 shows the dependency structure between the labels and the observations. Each segment is represented by the 37 features. Without loss of generality, we use $X_i^1, X_i^2, \dots, X_i^{37}$

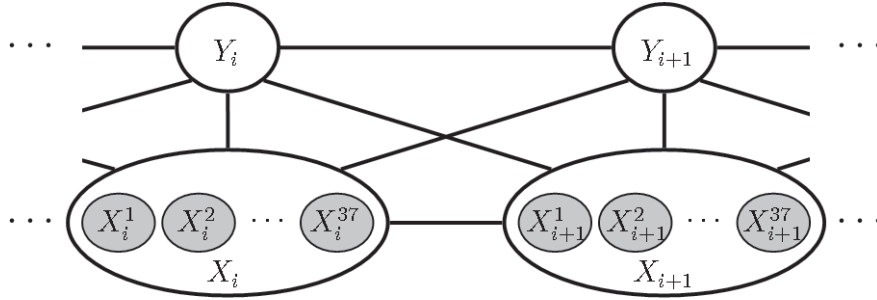


Figure 4.2: The CRF model used in this thesis. The assumption is that the transportation mode and visiting status of segment i depends on the 37 features of observations of segment $i-1$, i , and $i+1$.

to represent the features respectively. The assumption is that the transportation mode and visiting status of segment i (Y_i) depends on the label of previous and next segment (Y_{i-1}, Y_{i+1}) and the 111 features of observations of the previous segment, the current segment, and the next segment ($X_{i-1}^1, \dots, X_{i-1}^{37}, X_i^1, \dots, X_i^{37}, X_{i+1}^1, \dots, X_{i+1}^{37}$).

CRF uses *feature functions* to calculate the conditional probability distribution (Equation 2.3). The *feature functions* compose of two parts: the temporal relationship between two consecutive labels and the uni-gram functions between the label and a feature of one segment.

For each combination of consecutive hidden nodes and two label values, the indicative *feature function* is defined as

$$\mathbf{f}_{a_1, a_2}(Y_{i-1}, Y_i) = \begin{cases} 1, & \text{if } Y_{i-1} = a_1 \text{ and } Y_i = a_2 \\ 0, & \text{otherwise} \end{cases}$$

where $a_1, a_2 \in \mathbf{TM} \cup \mathbf{P}$.

For each combination of a hidden node and its surrounding 111 features, a set of

feature functions is defined for every transportation mode and visiting status value as

$$\mathbf{f}_{a,j,k}(X_{i+j}, Y_i) = \begin{cases} x, & \text{if } Y_i = a \text{ and } X_{i+j}^k = x \\ 0, & \text{otherwise} \end{cases}$$

where $a \in \mathbf{TM} \cup \mathbf{P}$, $j = -1, 0, 1$, and $k = 1 \dots 37$. The value is rounded to seven places after decimal separator.

In this research, we use CRF++ [13] for the implementation. CRF++ is an open source implementation for labeling sequential data. One advantage of CRF++ is that it allows flexible definition of the graph structure and the cliques. CRF++ uses forward/backward algorithms for computing the marginal probabilities and the normalization constant. Its training process is based on L-BFGS, a quasi-newton algorithm for large scale numerical optimization problem.

In training phase, given the training data $D = (D_1, D_2, \dots, D_N)$ where $D_i = (A_i, X_i)$, the learning criteria is to find the weight vector w that maximizes the log-likelihood of the training data. In prediction phase, given the observation of segments $X = X_1, X_2, \dots, X_N$, we picked the label sequence $Y = Y_1, Y_2, \dots, Y_N$ with the maximum conditional probability as the output.

The primary difference between a CRF model and an SVM model is that CRF can consider the temporal relationship of transportation mode and visiting status between segments, while SVM views all segments as independent observations without temporal relationship. However, trajectories are sequential data. It is commonly believed that the transportation mode and visiting status of current segment has relation to the ones of previous and next segments. As a result, we expect that CRF model should be more capable than SVM model of dealing the transportation mode learning problem.

4.3 Performance Measures

For a trajectory $Traj(t_b, t_e) = (g_1, g_2, \dots, g_n)$ of n position logs, the ground truth $A = (A_1, A_2, \dots, A_n)$ is the annotated visiting status or transportation mode. The segmentation process partitioned the trajectory into a segmented trajectory $T_s = (T_1, T_2, \dots, T_k)$ of k segments, and the corresponding ground truth $A' = (A'_1, A'_2, \dots, A'_k)$ is reassigned to the segments. And we denoted the inference result as $Y' = (Y'_1, Y'_2, \dots, Y'_k)$.

To evaluate the performance of the learning algorithm, the accuracy per segment (*APS*) is used. The accuracy per segment (*APS*) is the rate of matching labels between the inference result and the ground truth. Thus, it is defined as

$$APS(Y', A') = \frac{\sum_{i=1}^k \delta(Y'_i, A'_i)}{k} \quad (4.7)$$

$$\delta(Y'_i, A'_i) = \begin{cases} 1, & \text{if } Y'_i = A'_i \\ 0, & \text{otherwise} \end{cases}$$

However, the accuracy per segment is easily affected by the segmentation method. As a result, the accuracy per log (*APL*), the rate of matching labels between the inference result and the ground truth on the log granularity, is measured. To measure the per log accuracy, we mapped the learned labels Y' on segments to logs as $Y = (Y_1, Y_2, \dots, Y_n)$, where $Y_i = Y'_j$ if $g_i \in T_j$. Thus, the accuracy per log (*APL*) is defined as

$$APL(Y, A) = \frac{\sum_{i=1}^n \delta(Y_i, A_i)}{n} \quad (4.8)$$

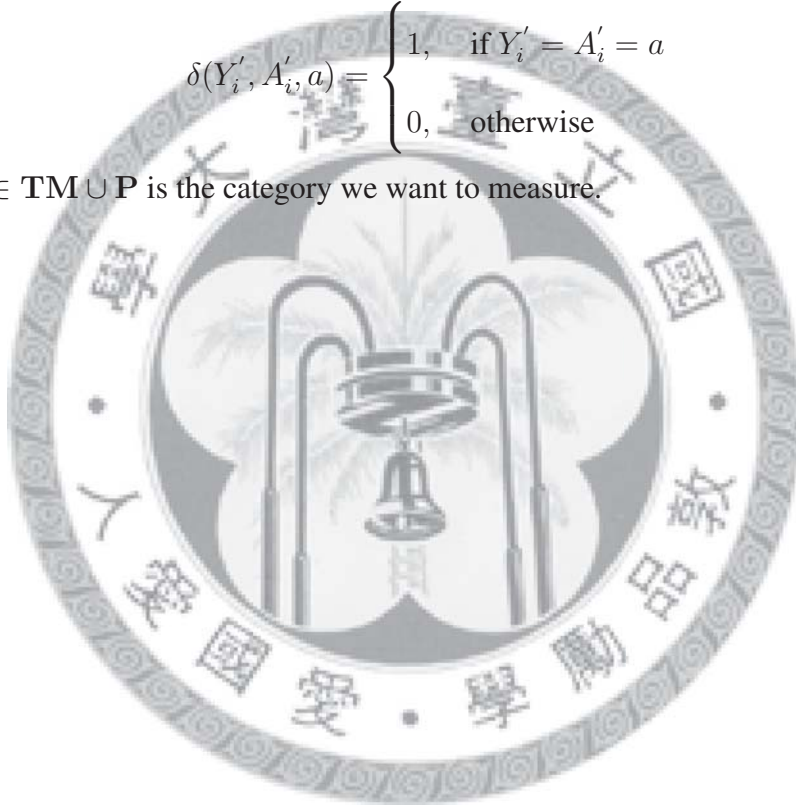
The accuracy per segment and accuracy per log measure the overall inference performance. However, these measures may be affected by the dominant transportation

mode and visiting status. For example, to a person who walk a lot, a system that always predicts the transportation mode as “walk” can get higher accuracy than random guess due to the high coverage of “walk”. As a result, we compare the recall value of each class to determine the performance on each category. The measure is defined as

$$Recall(Y', A', a) = \frac{\sum_{i=1}^k \delta(Y'_i, A'_i, a)}{\sum_{i=1}^k \delta(A'_i, a)} \quad (4.9)$$

$$\delta(Y'_i, A'_i, a) = \begin{cases} 1, & \text{if } Y'_i = A'_i = a \\ 0, & \text{otherwise} \end{cases}$$

where $a \in \mathbf{TM} \cup \mathbf{P}$ is the category we want to measure.





Chapter 5

Significant Place Set

In this chapter, the problem of mining personal significant places from GPS trajectories is tackled. After generating the location-transportation sequences using features and algorithms mentioned in Chapter 4, we obtain a collection of stop events S from the sequences. The places of these stop events form the place set for the user. Since there may be shifts on position log between each visiting to one place, clustering is applied to avoid redundancy. Features like visiting frequency, staying duration and travel efforts are extracted for each place in the place set. Then significances are estimated on the basis of the features.

5.1 Location Clustering

Since people may visit a place at different entrances at different time, it is possible that the locations identified in visiting events may be slightly shifted around the real position of the place. Hence, after identifying the stop locations, we cluster the nearby

locations into places. We adapted OPTICS (Ordering Points To Identify the Clustering Structure) [1] as the clustering algorithm, which is briefly described in Section 2.2.3. In implementation, we used WEKA 3.6 [24] as the tool to generate the ordering.

5.2 Significance Estimation

Among the places people visit in their daily lives, some places are more important and carry more semantic information to the person than other places. For example, the convenient store which was visited every day may play important role in one's life, while the restaurant visited once a month may be not that important. Generally speaking, significant places are those where user visits often and regularly and stays for a long period of time. In addition, the places where people may take longer travel time and distance to visit are potentially important.

5.2.1 Feature Selection

To estimate the significance of a place, we use the features listed in Table 5.1. The following paragraph explains the features extracted from the location-transportation sequences.

- **Visit frequency** The visit frequency (fr_v) counts the rate of visiting a place during the given interval. General speaking, places where people visit often, like convenient stores, are more important and significant than those seldom visited, like supermarkets. But the significance may be biased due to intensive visiting

Table 5.1: Features describing the visits of a place p .

Notation	Description
fr_v	The number of all stop events at the place
fr_{vd}	The number of days when more than one stop event occurred
fr_{vid}	The average number of visits in a day when more than one stop event occurred at that day
dr_s	The total length of stay duration
$\overline{dr_s}$	The average stay duration per stop event
$\overline{dr_{sd}}$	The average stay duration per day
$\overline{dr_b}$	The average duration between two visit to the place
$\overline{dr_{bd}}$	The average duration (in days) between two days when visiting to the place
$\overline{tf_d}$	The average travel distance from previous location
$\overline{tf_t}$	The average travel time from previous location

in particular day. For example, when moving house, people may travel back and forth between the old and the new houses. In this case, the trajectory will be summarized with a location-transportation sequence with several short stop events at the two places, hence increasing the frequency of both places. To capture the characteristic of this kind of situation, the number of visiting days (fr_{vd}) and the average frequency of visits per day (fr_{vid}) are used. Number of visiting days (fr_{vd}) views visits happened at the same day as one occurrence and ignores the frequency of visits during a day, while frequency of visits per day (fr_{vid}) estimates the amount of visit happened during a day.

- **Stay duration** Another intuitive measure of significance is the total duration spent at a place (dr_s). General speaking, places where the person spent more

time are more important and significant. However, duration may be affected by visit frequency, and the average stay duration ($\overline{dr_s}$) could reduce the effect of visit frequency. Similar to frequency measure, the stay duration may be biased due to intensive visiting in particular day. Hence, we accumulate the stay durations of a place of a day when more than one visits occurred, and calculate the average stay duration per visiting day ($\overline{dr_{sd}}$).

- **Interval between two visits**

In contrast to stay duration, the interval between two visits captures how possibly the user will return to the place again. The shorter the interval between two visits, the more significance the place plays in one's life. Similar to stay duration, intensive visiting in particular day may affect the average value of interval length. Hence, in addition to the average interval length ($\overline{dr_b}$) we also calculate the average day length ($\overline{dr_{bd}}$) between two stop events and visiting days.

- **Travel efforts**

Some places are substitutable, such as convenient stores. Most people prefer go to the convenient store with the least travel effort instead of spending hours to reach one specific store miles away. Real significant places are those people will visit even with much travel effort. In our work, we use the travel distance and travel time from previous stop location as the travel efforts, and average of travel efforts of distance ($\overline{tf_d}$) and time ($\overline{tf_t}$) are used.

5.3 Performance Measures

To measure the performance of significance estimation, we ask users to annotate the significant places from their viewpoints in the decreasing order. The user assigns an ordered list of significance places $SP_A = (SP_{A,1}, SP_{A,2}, \dots, SP_{A,k})$ as the ground truth where each place is represented by a point position and the relative significance is in decreasing order. That is, from the user's own viewpoint, $SP_{A,1}$ is more significant than $SP_{A,2}$. Places not listed in the ground truth list are regarded as insignificant places to the user. We denoted the α -cut significant place set derived from the location-transportation sequences as $SP_Y(\alpha) = (SP_{Y,1}, SP_{Y,2}, \dots, SP_{Y,k'})$. Noted that the derived place list may not be total ordered by the significance. Take Figure 5.1 for example. The user assigns five significant places as (A, B, C, D, E) while the system finds six places represented in BOPOMOFO alphabets.

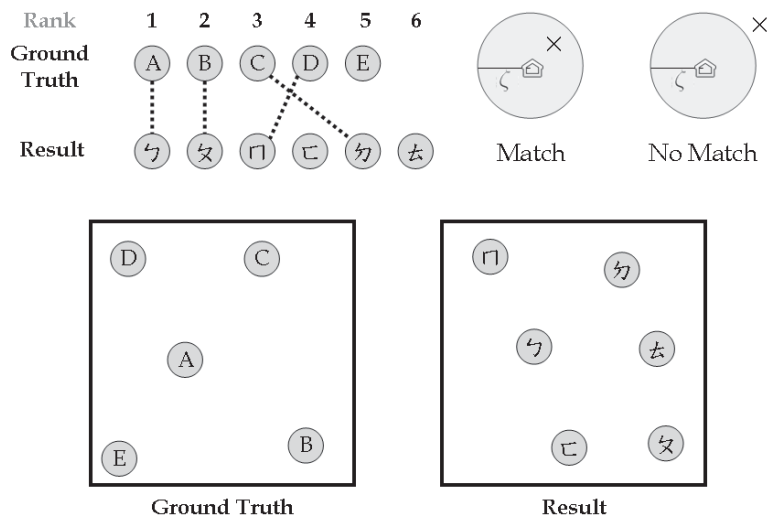


Figure 5.1: Example result of significant place set mining.

5.3.1 Precision and Recall

Given the annotated places by the user as the ground truth (SP_A) and the inferred places as prediction (SP_Y), we use precision and recall to measure the prediction accuracy. An inferred place $SP_{Y,i}$ is regarded as the same as a place in the ground truth list $SP_{A,j}$ if the distance between the two place positions are within ζ meters and they are mutually the closest neighborhood. We use the delta function to define the matching between two places.

$$\delta(SP_{Y,i}, SP_{A,j}) = \begin{cases} 1, & \text{if } dist(SP_{Y,i}, SP_{A,j}) < \zeta \text{ and} \\ & dist(SP_{Y,i}, SP_{A,j}) \leq dist(SP_{Y,i'}, SP_{A,j}) \text{ and} \\ & dist(SP_{Y,i}, SP_{A,j}) \leq dist(SP_{Y,i}, SP_{A,j'}) \\ 0, & \text{otherwise} \end{cases}$$

Take Figure 5.1 for example. The pairs of places linked with dashed lines match each other because they are mutually closest to each other and the distance is smaller than the threshold ζ . Thus, the precision and recall are defined as follows.

$$Precision(SP_Y, SP_A) = \frac{\sum_{i=1}^{k'} \sum_{j=1}^k \delta(SP_{Y,i}, SP_{A,j})}{k'} \quad (5.1)$$

$$Recall(SP_Y, SP_A) = \frac{\sum_{j=1}^k \sum_{i=1}^{k'} \delta(SP_{Y,i}, SP_{A,j})}{k} \quad (5.2)$$

5.3.2 Significance Ordering

People can compare the relative significance order of two places easily, but can hardly estimate a real value as the significance. Therefore, we evaluate the significant function

by pairwise comparison of the places. Given two derived places $SP_{Y,i}$ and $SP_{Y,j}$ and their corresponding significance $Sig(SP_{Y,i})$ and $Sig(SP_{Y,j})$, the comparison function is defined as

$$\text{sgn}(SP_{Y,i}, SP_{Y,j}) = \begin{cases} 1, & \text{if } Sig(SP_{Y,i}) < Sig(SP_{Y,j}) \\ -1, & \text{if } Sig(SP_{Y,i}) > Sig(SP_{Y,j}) \\ 0, & \text{otherwise} \end{cases} \quad (5.3)$$

If the ground truth is given, the real order of two derived places $SP_{Y,i}$ and $SP_{Y,j}$ should be the order of corresponding places in the ground truth list. Hence, the comparison function is defined as

$$C(SP_{Y,i}, SP_{Y,j}, SP_A) = \begin{cases} 1, & \text{if } \delta(SP_{Y,i}, SP_{A,i'}) = \delta(SP_{Y,j}, SP_{A,j'}) = 1 \text{ and } i' < j' \\ -1, & \text{if } \delta(SP_{Y,i}, SP_{A,i'}) = \delta(SP_{Y,j}, SP_{A,j'}) = 1 \text{ and } i' > j' \\ 0, & \text{otherwise} \end{cases} \quad (5.4)$$

The strict ordering accuracy (OA_{str}) is defined as portion of pairwise comparison which is consistent with the ground truth list. In the definition, $H_\alpha[x]$ is the Heaviside step function which defines $H[0]$ as α .

$$OA_{str}(SP_Y, SP_A) = \frac{\sum_{i=1}^{k'} \sum_{j=i+1}^{k'} H_0[\text{sgn}(SP_{Y,i}, SP_{Y,j})C(SP_{Y,i}, SP_{Y,j}, SP_A)]}{\sum_{i=1}^{k'} \sum_{j=i+1}^{k'} |C(SP_{Y,i}, SP_{Y,j}, SP_A)|} \quad (5.5)$$

$$H_\alpha[x] = \begin{cases} 1, & \text{if } x > 0 \\ \alpha, & \text{if } x = 0 \\ 0, & \text{if } x < 0 \end{cases}$$

5.3.3 NDCG

NDCG (normalized discounted cumulative gain) [11] is a common measure of effectiveness in information retrieval when items have different relevance to the users and highly relevant items are expected to appear early in the the ranked result. In our problem, the relevance of each place is represented by the significance given by the user, and an ideal significant measure should ranked the places in accordance with the significance values. Figure 5.2 illustrates the calculation of NDCG. First, each significant place on the list of user's annotation receives a score as the length of the list plus one minus the order of the place. That is, the n -th least significant place annotated by the user has score of n . For each place in the result list, the gain value is set as the score of its corresponding matched place. Then, the NDCG value of a ranked result list can be calculated.

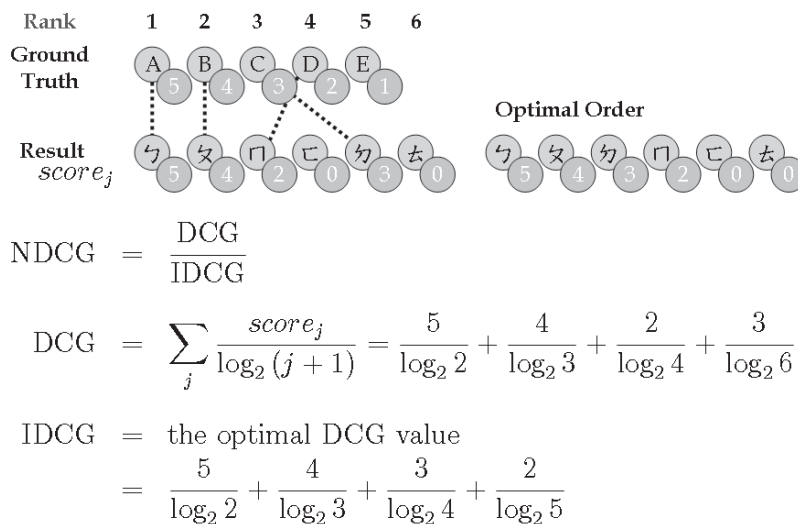


Figure 5.2: Example of NDCG calculation.

Chapter 6

Experiment and Evaluation

To evaluate the capability of the proposed trajectory model, we invited some participants to collect their trajectories with commercial GPS loggers. The participants carried the GPS loggers, and their positions were automatically recorded without manual operation. The ground truth of trajectory segments were manually labeled off-line either with a separate record sheet or on the experiment website.

6.1 The Dataset

6.1.1 Data Collection

In most previous work, researchers adopted the solution of connecting a Bluetooth GPS receiver to the smart phone or PDA to record the GPS signals and display the position immediately on the screen with the demo applications. In this research, two commercial GPS loggers, Holux 241 (Figure 6.1(a)) and NCSNavi R150+ (Figure 6.1(b)), were used to collect the user trajectory.

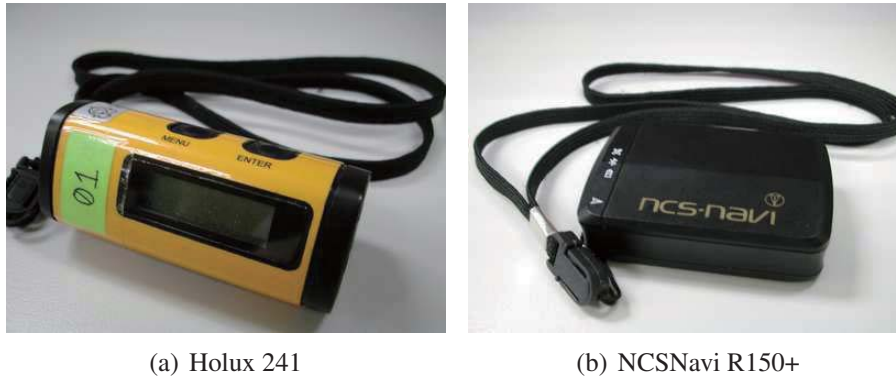


Figure 6.1: GPS loggers used in the experiments.

We invited 12 individuals to carry the GPS log with them for a long period of time. Each person was asked to open the log when they are traveling among different places in the city. For reducing the power consumption and increasing the time between charges, participants were allowed to turn off the loggers when they entered indoor environments and the GPS signals were lost. Due to different personal schedules and willingnesses to reveal the schedules, different participants had different length of logging period. But all participants recorded their trajectories for at least one week. Table 6.1 shows the data collection period and actual logged length for each participant.

6.1.2 Data Annotation

After collecting the logs, participants were asked to annotate the ground truth with the annotation web page (See Figure 6.2). The page contains 3 parts: a map showing the spatial part of trajectory, a time line showing the speed and heading direction change along the time, and the annotation utilities. The annotation labels are: “HOME”, “WORK”, “DINING”, “STORE”, “HAVING FUN”, “TRANSIT”, “STOP”, “WALK”,

Table 6.1: Statistics of dataset, including data collection period, logged length of trajectories, annotated length of trajectories, and number of significant places. (a) Number of collected trajectories. (b) Number of annotated trajectories. (c) Users with “*” sign are selected in the experiment of visiting status and transportation mode learning. (d) Number of significant places annotated by user.

user	begins mm/dd/yy	ends mm/dd/yy	(a)	logged length	(b)	annotated length	(c)	(d)
user01	08/12/08	03/06/09	426	546:14:05	63	87:27:19	*	9
user02	02/07/09	03/05/09	11	11:36:40	10	10:23:56	*	5
user03	11/12/08	03/05/09	130	42:55:09	38	06:50:16	*	9
user04	01/04/09	03/31/09	74	129:44:57	1	04:00:14		—
user05	01/21/09	02/05/09	15	27:21:35	12	25:04:28	*	6
user06	12/25/08	03/06/09	38	32:34:31	0	—		4
user07	02/16/09	03/06/09	8	11:12:43	0	—		18
user08	01/21/09	03/05/09	39	80:01:20	38	72:01:20	*	5
user09	03/18/09	04/03/09	32	74:25:17	0	—		—
user10	12/31/08	01/14/09	29	25:12:18	2	01:37:42		5
user11	01/21/09	03/05/09	39	69:44:32	37	36:35:53	*	10
user12	11/14/08	01/09/09	21	05:45:52	0	—		—

“BIKE”, “SCOOTER”, “CAR”, “METRO”, “MOVE”, “UNKNOWN”, and “NOISE”. The label “STOP” and “MOVE” stand for a place or a transportation not listed in the available selection, while the label “UNKNOWN” and “NOISE” are used to get rid of unsure or noisy data. We use Google Maps API and Google Visualization API to implement the map and time line.

The annotation process is as follows. Participants use the time line to select an interval of time, and the trajectory during that interval is highlighted on the map. Then participants select the corresponding labels from the label list and annotate the interval. After annotating the whole trajectory, participants can save their annotations. If the participants cannot recall any information about the trajectory, they can skip the tra-

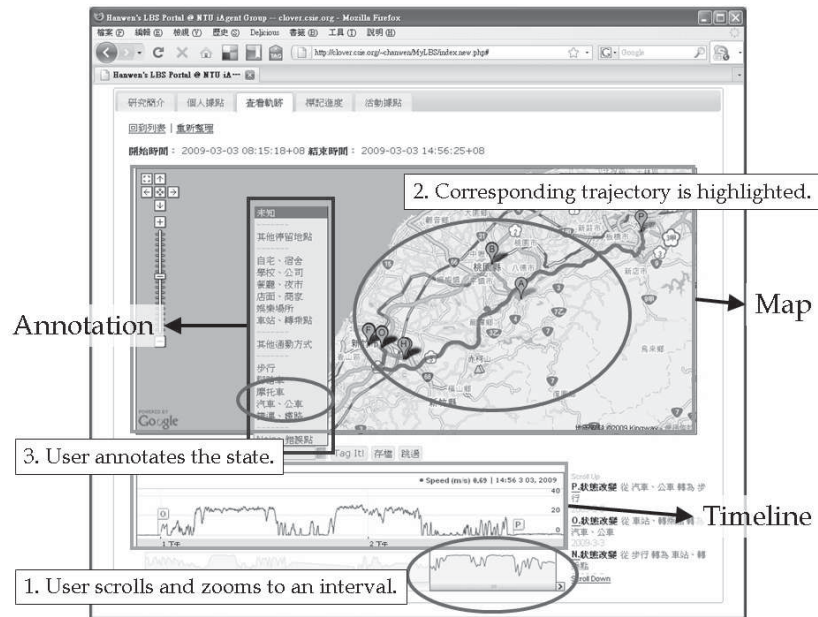


Figure 6.2: A screen shot of the off-line annotation web page.

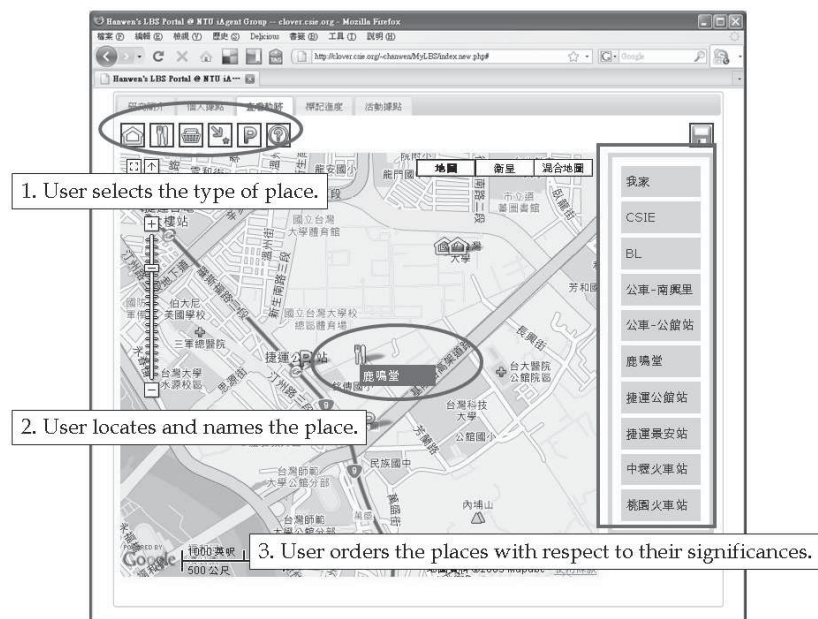


Figure 6.3: A screen shot of the significant place annotation web page.

jectory without annotation. Four of the 12 participants annotated all their trajectories, while two of other eight participants annotated more than 30 trajectories. Their annotated trajectories were used in the experiment of transportation mode and visiting status learning. Table 6.1 shows the portion of annotated trajectories for each participant, and those marked with a “*” sign at the last column are the ones whose trajectories were used in the following experiment.

In addition to annotating the trajectories, participants were asked to mark their significant places in their mind. On the labeling web page (See Figure 6.3), participants first select the corresponding icon of the place at the upper-left corner and a new marker will be generated. These icons respectively, from left to right, represent “Home or Office”, “Restaurants”, “Stores”, “Fun places”, “Parking lots/Transit Stop” and “Other places.” Participants can drag the markers to the position of the place and enter a name for the place. As a consequence, a list item will be generated at the right side, and a significant place is successfully created. After creating the significant place set, participants can re-order the places by dragging the list items at the right side. Thus, we obtained the ordered significant place set from the participants’ viewpoints.

6.2 Transportation Mode Learning

6.2.1 Experiment Steps

In our experiment, we compared nine segmentation methods: uniform duration segmentation with 60 seconds, 90 seconds and 120 seconds, uniform length segmentation

with 10 meters, 50 meters and 100 meters, and uniform grid segmentation with 10 meters, 50 meters, and 100 meters as the side length. For each trajectory segments obtained, we extracted the features listed in Table 4.1, and we used CRF++ [13] and libSVM [5] as the tools to implement the CRF and SVM model. For each participant in the test group and each segmentation method, we first chronicled their trajectories and divided the trajectories into 5 non-overlapping sets one-by-one to run 5-fold cross validation. Figure 6.4 illustrates the composition of each fold. In addition, we ran cross subject validation, which use trajectories of 5 participants to train the model and use the other one's trajectories to test.

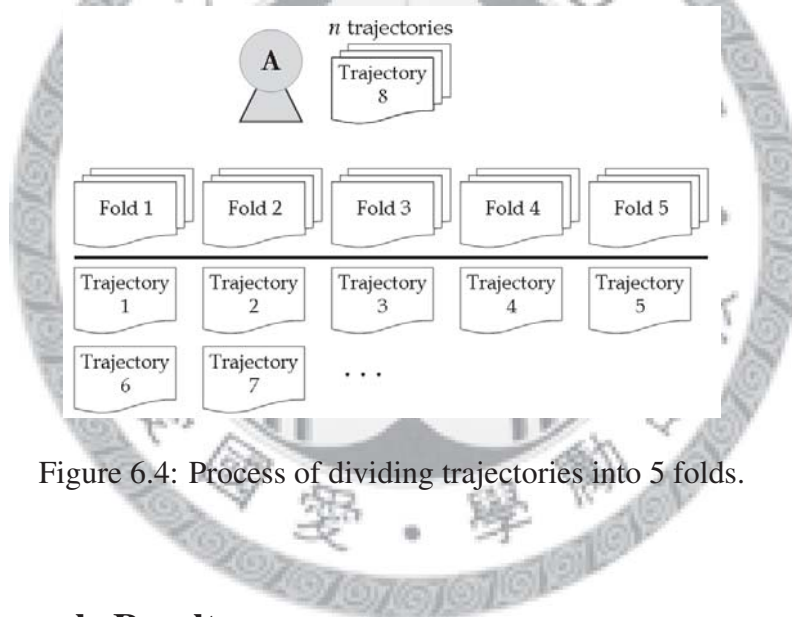


Figure 6.4: Process of dividing trajectories into 5 folds.

6.2.2 Example Result

Table 6.2 and Table 6.3 show the accuracy per segment (APS) of 5-fold cross validation using CRF and SVM for different users and segmentation methods. From these two tables, we found that when we choose uniform duration segmentation, CRF outperforms SVM in all cases; when we choose uniform length segmentation and uniform

grid segmentation, CRF outperforms SVM in most cases. And in general, using the uniform grid segmentation achieves the best accuracy per segment. Table 6.4 shows the accuracy per log (APL) of 5-fold cross validation using CRF for different users and segmentation methods. However, different from the results of accuracy by segment, the dominance on accuracy by log of uniform grid segmentation is not obvious.

Since the ultimate goal of transportation learning is to automatically identify the transportation modes without bothering users to make annotations, we used leave-one-subject-out validation to test whether the model trained by other users could be used on a new user without any annotations. For each user, we learned the model from trajectories of other five users and used the learned model to test the accuracy of the specific user. Table 6.5 and Table 6.6 show the accuracy per segment (APS) of leave-one-subject-out cross validation using CRF and SVM for different users and segmentation methods. From these two tables, we found that CRF model is more capable of labeling sequences from unseen users than SVM. Table 6.7 shows the accuracy per log (APL) of leave-one-subject-out cross validation using CRF for different users and segmentation methods.

However, the accuracy may be affected by the distribution of each transportation mode and visiting status and their corresponding recall. In Table 6.8 and Table 6.9, we list the confusion matrix of 5-fold cross-validation using CRF model on trajectories of user11 with uniform length segmentation with 100 meters and uniform duration segmentation with 60 seconds respectively. Due to display limitation on paper, the rows and columns with labels not in the ground truth or inference result are removed. From the rows of both tables, clearly that the lengths of different transportation modes

Table 6.2: Accuracy per segment of 5-fold cross validation using CRF.

APS (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Grid (m)		
	10	50	100	60	90	120	10	50	100
user01	41.78	39.05	40.80	37.02	36.49	35.81	46.71	39.02	40.83
user02	66.17	67.24	65.88	38.57	39.64	37.60	70.04	70.78	68.28
user03	76.33	78.24	79.07	61.46	61.40	66.99	80.23	82.20	81.75
user05	40.52	51.39	60.04	27.16	27.31	26.21	49.08	54.82	61.98
user08	38.24	43.73	45.14	42.74	43.28	43.99	37.82	47.12	48.52
user11	82.64	83.84	85.87	81.03	81.21	79.34	83.01	81.53	85.59
average	57.62	60.58	62.80	48.00	48.22	48.32	61.15	62.58	64.49

Table 6.3: Accuracy per segment of 5-fold cross validation using SVM.

APS (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Grid (m)		
	10	50	100	60	90	120	10	50	100
user01	16.04	10.78	8.10	25.31	24.51	23.58	14.15	7.14	5.74
user02	72.69	72.98	72.14	37.84	29.97	24.80	75.17	73.23	69.67
user03	78.54	80.79	80.00	58.99	60.96	59.14	79.37	82.07	80.79
user05	43.51	50.46	51.94	22.33	27.31	26.21	45.60	50.76	50.88
user08	36.11	41.93	37.08	24.24	22.55	22.89	39.05	46.16	41.09
user11	81.86	85.05	85.33	72.38	70.15	67.68	82.40	85.53	85.78
average	54.79	57.00	55.76	40.18	38.17	35.85	55.96	57.48	55.66

Table 6.4: Accuracy per log of 5-fold cross validation using CRF.

APL (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Grid (m)		
	10	50	100	60	90	120	10	50	100
user01	33.17	41.86	43.94	37.28	36.83	36.16	34.67	30.19	27.23
user02	46.73	45.73	43.08	39.22	40.00	38.07	50.07	45.65	40.36
user03	57.66	58.17	61.10	62.21	62.49	68.43	60.58	61.50	63.13
user05	34.19	40.94	35.61	26.90	26.90	26.10	39.70	45.92	39.92
user08	41.19	45.25	40.02	43.11	43.46	44.22	42.22	36.93	29.38
user11	75.98	77.27	77.29	82.07	81.46	79.84	75.14	75.52	76.25
average	48.15	51.54	50.18	48.46	48.52	48.80	50.40	49.29	46.05

Table 6.5: Accuracy per segment of cross subject validation using CRF.

APS (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Grid (m)		
	10	50	100	60	90	120	10	50	100
user01	35.27	63.18	34.58	14.85	12.72	11.07	40.84	72.66	57.79
user02	55.81	62.08	59.37	41.63	43.24	46.80	55.78	72.16	70.55
user03	7.89	8.52	9.40	39.29	28.31	28.71	10.35	9.45	10.02
user05	45.81	28.54	36.77	39.14	40.16	37.73	49.34	27.90	48.98
user08	30.61	38.12	33.81	17.34	20.33	21.06	32.51	43.67	39.32
user11	59.26	63.32	67.82	65.11	68.39	67.29	63.12	64.47	69.34
average	39.11	43.96	40.29	36.23	35.53	35.44	41.99	48.38	49.33

Table 6.6: Accuracy per segment of cross subject validation using SVM.

APS (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Grid (m)		
	10	50	100	60	90	120	10	50	100
user01	3.79	3.01	2.49	4.10	3.64	3.77	3.78	2.08	1.73
user02	0.05	0.18	0.00	0.24	0.00	0.00	0.12	0.00	0.00
user03	36.41	36.76	34.60	8.21	4.49	0.57	32.69	43.69	26.23
user05	30.45	38.81	34.56	21.77	19.21	14.12	32.65	40.20	36.55
user08	1.46	1.93	1.89	0.91	0.94	1.20	1.53	2.21	2.47
user11	4.30	3.33	3.20	3.32	3.31	3.19	4.13	3.37	3.19
average	12.74	14.00	12.79	6.43	5.27	3.81	12.48	15.26	11.69

Table 6.7: Accuracy per log of cross subject validation using CRF.

APL (%)	Uniform Length (m)			Uniform Duration (s)			Uniform Length (m)		
	10	50	100	60	90	120	10	50	100
user01	17.94	18.14	16.47	14.68	12.32	10.93	18.89	19.90	16.80
user02	46.79	51.37	46.18	42.14	43.58	47.29	46.05	54.06	49.78
user03	16.23	16.71	16.09	38.66	28.31	27.80	18.83	19.86	17.23
user05	38.57	32.30	33.88	39.03	40.11	38.10	46.38	32.63	37.44
user08	22.75	27.33	26.86	17.41	19.93	20.43	22.82	26.51	26.53
user11	60.31	62.64	66.16	68.60	70.19	69.98	65.31	65.07	66.10
average	33.77	34.75	34.27	36.75	35.74	35.76	36.38	36.34	35.65

Table 6.8: Confusion matrix of 5-fold cross-validation using uniform length segmentation with 100 meters and CRF model on trajectories of user11.

#seg	WALK	BIKE	SCOOTER	CAR	METRO	Recall (%)
STOP	3	0	0	0	0	0
WALK	505	4	3	66	0	87.37
BIKE	51	9	0	28	0	10.23
SCOOTER	0	0	0	349	0	0
CAR	15	3	157	8884	141	96.57
METRO	4	0	0	598	0	0
HOME	3	0	0	0	0	0
WORK	26	0	0	0	0	0
DINING	16	0	0	0	0	0
STORE	2	0	0	4	0	0
TRANSIT	22	0	2	33	0	0

Table 6.9: Confusion matrix of 5-fold cross-validation using uniform duration segmentation with 60 seconds and CRF model on trajectories of user11.

#seg	WALK	BIKE	CAR	TRANSIT	Recall (%)
STOP	4	0	0	0	0
WALK	810	0	42	19	93.00
BIKE	18	5	31	0	9.26
SCOOTER	0	0	96	0	0
CAR	37	0	1542	7	97.23
METRO	8	0	62	0	0
HOME	10	0	1	0	0
WORK	27	0	0	0	0
DINING	35	0	0	0	0
STORE	6	0	0	0	0
TRANSIT	124	0	28	61	28.64

were not balanced, and the user was in car for most of the time. In addition, except transits, most of stop events were classified as walking instead of corresponding stop events. There are two reasons to explain the difficulty to correctly label the visiting status. On one hand, most of the visited places are indoor and few position logs were recorded due to GPS opacity. However, when moving between places, users is in the outdoor environment and plenty position logs were recorded. As a result, the data are unbalanced. On the other hand, when users enter an indoor location, the GPS signal is lost in a short response time. Hence, the position logs of stop events may only last a short period. Moreover, when users stay at one location, the travel distances between position logs are short. As a result, when segmenting the trajectories with large time and distance interval, the position logs of stop events may be segmented together with some logs labeled with transportation modes. In this case, the annotation for the segment may be dominated by the other transportation mode, and the data are even more unbalanced.

6.3 Significant Location Mining

6.3.1 Experiment Steps

The mining of significant place includes four steps: detecting the stop events, clustering the stop positions into places, calculating features, and estimating the significance.

In our experiment, we detected three kinds of stop events: the stop segments in the location-transportation sequences, ends of trajectories, and losses of GPS signals.

The stop events detected by the three conditions are respectively denoted as SE_{lts} , SE_{end} , SE_{lost} . And the union of the three sets are denoted as SE_{all} . The last two conditions deal with the indoor visits and the first one deals with the outdoor visits. However, temporary signal losses may be due to factors other than indoor-opacity. In our experiment, we directly use the annotations on the trajectories given by the users to construct the location-transportation sequences, and extracted the location parts and the transit events as the stop events. As to the detection of signal losses, we only considered the last positions before the signal being lost for more than 5 minutes as the stop events. After detecting the stop events, we calculated the length of stay duration and the distance and time effort from the last detected stop event. However, at the end of trajectory, the information of the stay duration is not provided. In these cases, we use 4 hours as the default setting when the information is not available.

We use OPTICS in the clustering process, and we set the Eps as 1000 meters and $MinPts$ as 2 to generate the clustering ordering. Because the clustering process is to group nearby stops into places, a small $MinPts$ is enough. After generating the ordering, we choose $\xi = 0.3$ to detect steep areas in the sequence and find the possible clusters. After some trials, we observed that the granularity of detected clusters at the region where user frequently visits may be too small, and more than one clusters may be created for one place. As a result, we added a 10-meter filter in the clustering process that steep areas with reachability less than 10-meter are omitted. Although OPTICS can generate clusters of different size into a hierarchical structure, we only considered the leaf clusters as the places.

6.3.2 Example Result

In Figure 6.5, parts of the significant places annotated by user11 and the ones inferred from the stop events detected by all three conditions are shown on the map. When calculating the location accuracy, we set ζ as 100 meters. The precision and recall of inferred places without pruning by significance are organized in Table 6.10. Since there may be some significant places users do not notice when annotating the ground truth, and the places user explicitly pointed out should be really significant, the precision is more important than recall in the analysis. From the results in Table 6.10, we found that end points are more representative than stop segments and signal losses in predicting the significant places, and combing all the three sets of stop events gives higher recall.

Table 6.11 shows the strict ordering accuracy (OA_{str}) of the inferred significant places using exactly one of the ten features. From the results, we found that the ordering of interval in days between two visit days and the average time effort best fit the ordering in the user's mind. The average NDCG (Figure 6.6), however, shows that visit frequency and stay duration predict the most significant places more accurately.



(a) Ground truth



(b) Inferred places

Figure 6.5: The ground truth and inferred significant places of user11.

Table 6.10: Precision and recall of inferred significant places using the parameters ($Eps = 1000$ meters, $MinPts = 2$, $\xi = 0.3$, $\zeta = 100$ meters, $\alpha = 0$)

Precision / Recall	SE_{lts}	SE_{end}	SE_{lost}	SE_{all}
user01	75.00 / 33.33	30.77 / 44.44	33.33 / 33.33	27.27 / 100.00
user02	66.67 / 40.00	100.00 / 40.00	— / 0.00	42.86 / 60.00
user03	— / 0.00	40.00 / 22.22	100.00 / 11.11	50.00 / 33.33
user05	0.00 / 0.00	60.00 / 50.00	— / 0.00	57.14 / 66.67
user08	0.00 / 0.00	0.00 / 0.00	50.00 / 20.00	7.69 / 20.00
user11	42.86 / 30.00	57.14 / 40.00	33.33 / 10.00	12.50 / 20.00

Table 6.11: Strict Ordering Accuracy (OA_{str}) of the inferred significant places using exactly one of the ten features.

OA_{str} (%)	fr_v	fr_{vd}	fr_{vid}	dr_s	$\overline{dr_s}$	$\overline{dr_{sd}}$	$\overline{dr_b}$	$\overline{dr_{bd}}$	$\overline{tf_d}$	$\overline{tf_t}$
user01	30.6	38.9	25.0	36.1	50.0	44.4	36.1	75.0	50.0	63.9
user02	33.3	33.3	0.0	66.7	66.7	66.7	33.3	66.7	66.7	66.7
user03	0.0	0.0	0.0	0.0	0.0	0.0	33.3	33.3	0.0	66.7
user05	66.7	0.0	83.3	50.0	50.0	50.0	33.3	16.7	50.0	66.7
user08	—	—	—	—	—	—	—	—	—	—
user11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	100.0	100.0

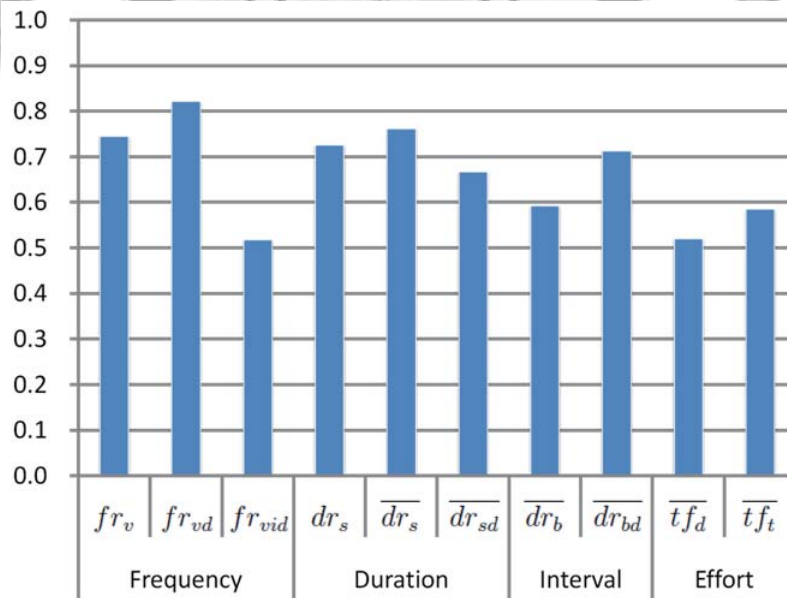
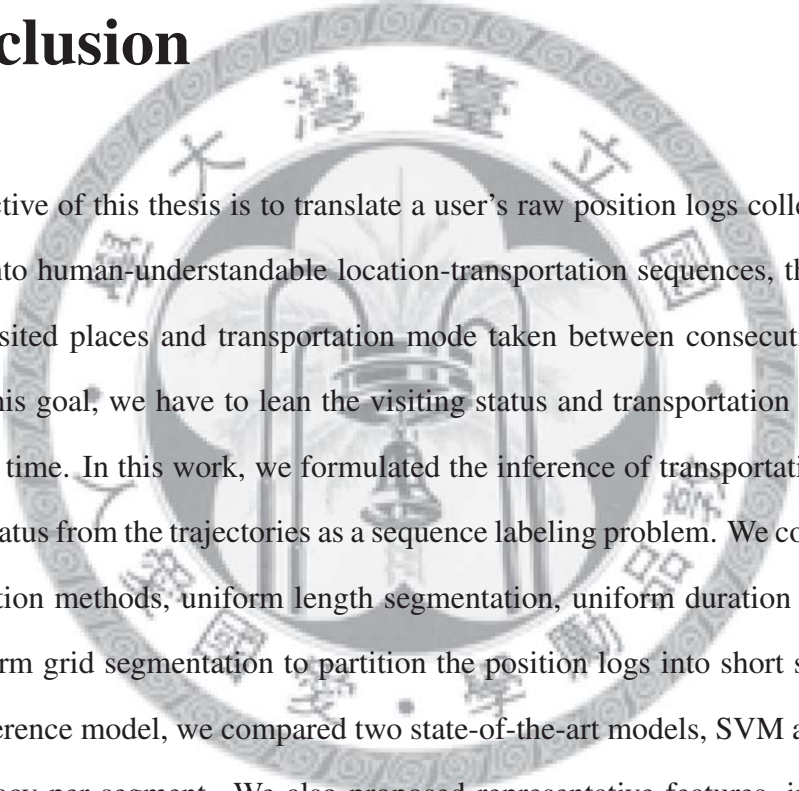


Figure 6.6: Average NDCG values of ten measures.



Chapter 7

Conclusion



The objective of this thesis is to translate a user's raw position logs collected by GPS loggers into human-understandable location-transportation sequences, the concatenation of visited places and transportation mode taken between consecutive stays. To achieve this goal, we have to learn the visiting status and transportation mode at each particular time. In this work, we formulated the inference of transportation mode and visiting status from the trajectories as a sequence labeling problem. We compared three segmentation methods, uniform length segmentation, uniform duration segmentation and uniform grid segmentation to partition the position logs into short segments. As to the inference model, we compared two state-of-the-art models, SVM and LCRF, by the accuracy per segment. We also proposed representative features, including spatial features and temporal context features, to capture the characteristics of trajectory segments. As to the significant places mining, we considered the stop segments, the ends of trajectories, and the losses of GPS signals as the stop events. To reduce the redundancy of stop positions, we used OPTICS as the clustering tools to group nearby

stop positions into places. The strength of OPTICS is the capability to generate dense clusters at different granularities. For each place, we calculated 10 measures about the visit frequency, stay duration, visit interval and travel effort. For each measure, we compared the ordering accuracy in respect to annotations given by the users.

In this research, we collected our own dataset. We used commercial GPS loggers to record trajectories of 12 participants without distracting user's attention to annotate the change of status immediately. In contrast, we created an independent trajectory managing website for the users to archive their trajectories, view the trajectories, and annotate the trajectories when they have time. From the experiment results, we showed that CRF outperforms SVM in the transportation mode and visiting status mining problem. As to the segmentation method, uniform grid segmentation resulted in the highest accuracy per segment among the three approaches. And from detailed examination of the results, we found that visiting status is more difficult to be learned than transportation mode. About significant location mining, we found that the ordering of interval in days between two visit days and the average time effort are accurate predictions to fit the ordering in the user's mind. However, with discount considered, visit frequency and stay duration predict the most significant places more accurately.

Bibliography

- [1] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander. OPTICS: Ordering points to identify the clustering structure. *ACM SIGMOD Record*, 28(2):49–60, June 1999.
- [2] D. Ashbrook and T. Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal Ubiquitous Computing*, 7(5):275–286, October 2003.
- [3] N. Bicocchi, G. Castelli, M. Mamei, A. Rosi, and F. Zambonelli. Supporting location-aware services for mobile users with the whereabouts diary. In *Proceedings of the 1st International Conference on MOBILE Wireless MiddleWARE, Operating Systems, and Applications (MOBILWARE 2008)*, pages 1–6. ICST, February 2008.
- [4] D. Brosset, C. Claramunt, and E. Saux. A location and action-based model for route descriptions. In *Proceeding of the 2nd International Conference on GeoSpatial Semantics (GeoS 2007)*, pages 146–159. Springer-Verlag, November 2007.
- [5] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [6] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996)*, pages 226–231. AAAI Press, August 1996.

- [7] T. A. S. Foundation. *Commons-Math: The Apache Commons Mathematics Library*, 2007. Software available at <http://commons.apache.org/math/>.
- [8] L. Fritsch. *Profiling and Location-Based Services (LBS)*, chapter Profiling and Location-Based Services (LBS), pages 147–168. Springer Netherlands, May 2008.
- [9] J. Froehlich, M. Y. Chen, I. E. Smith, and F. Potter. Voting with your feet: An investigative study of the relationship between place visit behavior and preference. In *Proceedings of the 8th International Conference on Ubiquitous Computing (UbiComp 2006)*, volume 4206 of *Lecture Notes in Computer Science*, pages 333–350. Springer, September 2006.
- [10] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi. Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2007)*, pages 330–339. ACM Press, August 2007.
- [11] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, October 2002.
- [12] J. Krumm and E. Horvitz. Predestination: Where do you want to go today? *IEEE Computer*, 40(4):105–107, 2007.
- [13] T. Kudo. *CRF++: Yet Another CRF toolkit*, December 2007. Software available at <http://crfpp.sourceforge.net/>.
- [14] V. Kulyukin, J. Nicholson, D. Ross, J. Marston, and F. Gaunet. The blind leading the blind: Toward collaborative online route information management by individuals with visual impairments. In *Proceedings of AAAI 2008 Spring Symposium Series on Social Information Processing*. AAAI, March 2008.
- [15] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning (ICML)*, June 2001.
- [16] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma. Mining user similarity based

- on location history. In *Proceedings of the 16th Annual ACM International Symposium on Advances in Geographic Information Systems (GIS 2008)*, pages 1–10. ACM, November 2008.
- [17] L. Liao, D. Fox, and H. Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *International Journal of Robotics Research*, 26(1):119–134, January 2007.
- [18] N. Marmasse and C. Schmandt. A user-centered location model. *Personal Ubiquitous Computing*, 6(5-6):318–321, December 2002.
- [19] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares. A clustering-based approach for discovering interesting places in trajectories. In *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC 2008)*, pages 863–868. ACM, March 2008.
- [20] J. Rekimoto, T. Miyaki, and T. Ishizawa. Lifetag: Wifi-based continuous location logging for life pattern analysis. In *Proceedings of the 3rd International Symposium on Location- and Context-Awareness (LoCA 2007)*, volume 4718 of *Lecture Notes in Computer Science*, pages 35–49, September 2007.
- [21] S. D. Sabbata, S. Mizzaro, and L. Vassena. Spacerank: Using pagerank to estimate location importance. In *Proceedings of ECAI 2008 Workshop on Mining Social Data*, July 2008.
- [22] Y. Takeuchi and M. Sugimoto. A user-adaptive city guide system with an unobtrusive navigation interface. *Personal and Ubiquitous Computing*, 13(2):119–132, February 2009.
- [23] T. Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, 22(176):88–93, April 1975.
- [24] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2 edition, June 2005.

- [25] X. Xiao, X. Xie, Q. Luo, and W.-Y. Ma. Density based co-location pattern discovery. In *Proceedings of the 16th Annual ACM International Symposium on Advances in Geographic Information Systems (GIS 2008)*, pages 1–10, 2008.
- [26] J. Ye, L. Coyle, S. Dobson, and P. Nixon. A unified semantics space model. In *Proceedings of the 3rd International Symposium on Location- and Context-Awareness (LoCA 2007)*, volume 4718 of *Lecture Notes in Computer Science*, pages 103–120, September 2007.
- [27] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie. Mining individual life pattern based on location history. In *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM 2009)*, May 2009.
- [28] K. Zhang, H. Li, K. Torkkola, and M. Gardner. Adaptive learning of semantic locations and routes. In *Proceedings of the 1st International Conference on Autonomic Computing and Communication Systems (Autonomics 2007)*, pages 1–10. ICST, 2007.
- [29] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma. Understanding mobility based on gps data. In *Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp 2008)*, pages 312–321. ACM Press, September 2008.
- [30] Y. Zheng, L. Wang, R. Zhang, X. Xie, and W.-Y. Ma. GeoLife: Managing and understanding your past life over maps. In *Proceedings of the 9th International Conference on Mobile Data Management (MDM 2008)*, pages 211–212, April 2008.
- [31] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining interesting locations and travel sequences from GPS trajectories. In *Proceedings of the 18th International Conference on World Wide Web (WWW 2009)*, pages 791–800, April 2009.
- [32] C. Zhou, N. Bhatnagar, S. Shekhar, and L. Terveen. Mining personally important places from gps tracks. In *Proceedings of the 23rd IEEE International Conference on Data Engineering Workshop*, pages 517–526, April 2007.

- [33] C. Zhou, D. Frankowski, P. Ludford, S. Shekhar, and L. Terveen. Discovering personal gazetteers: an interactive clustering approach. In *Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems (GIS 2004)*, pages 266–273. ACM, November 2004.
- [34] B. Ziebart, A. L. Maas, A. K. Dey, and J. A. Bagnell. Navigate like a cabbie: probabilistic reasoning from observed context-aware behavior. In *Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp 2008)*, pages 322–331. ACM, September 2008.

