Transportation Research Part B 83 (2016) 121–135

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

# Activity imputation for trip-chains elicited from smart-card data using a continuous hidden Markov model

Gain Han, Keemin Sohn*

*Department of Urban Engineering, Chung-Ang University, 221, Heukseok-dong, Dongjak-gu, Seoul 156-756, Republic of Korea*

ARTICLE INFO

*Article history:*
Received 5 June 2015
Revised 24 November 2015
Accepted 24 November 2015
Available online 11 December 2015

*Keywords:*
Smart-card data
Activity-based model
Activity imputation
Continuous hidden Markov model
Machine learning
Trip chain

ABSTRACT

Although smart-card data were expected to substitute for conventional travel surveys, the reality is that only a few automatic fare collection (AFC) systems can recognize an individual passenger's origin, transfer, and destination stops (or stations). The Seoul metropolitan area is equipped with a system wherein a passenger's entire trajectory can be tracked. Despite this great advantage, the use of smart-card data has a critical limitation wherein the purpose behind a trip is unknown. The present study proposed a rigorous methodology to impute the sequence of activities for each trip chain using a continuous hidden Markov model (CHMM), which belongs to the category of unsupervised machine-learning technologies. Coupled with the spatial and temporal information on trip chains from smart-card data, land-use characteristics were used to train a CHMM. Unlike supervised models that have been mobilized to impute the trip purpose to GPS data, A CHMM does not require an extra survey, such as the prompted-recall survey, in order to obtain labeled data for training. The estimated result of the proposed model yielded plausible activity patterns that are intuitively accountable and consistent with observed activity patterns.

## 1. Introduction

From the perspective of transportation planners, the Seoul metropolitan area is equipped with the world's best transit fare collection system. This system recognizes every passenger's origin, transfers, and destination stops (or stations) as well as providing exact time stamps. However, smart-card data will not replace conventional household surveys until the trip purpose can be identified in a reliable manner. In this regard, the present study proposed a robust methodology to impute activities for smart-card data by using a continuous hidden Markov model (CHMM). The model uses unsupervised machine-learning technology that requires no labeled data for training. When imputing the purpose, destination, or mode of GPS-based location data, many researchers have utilized various mathematical models that require a calibration procedure (Yang et al., 2010; Moiseeva et al., 2010; Allahviranloo and Recker 2013a, 2013b; Lu et al., 2013; Reumers et al., 2013; Liu et al., 2013), which computer scientists regard as supervised machine-learning technology. Furthermore, prompted-recall surveys have been a mainstream tool to obtain labeled data for calibrating and validating supervised imputation models (Feng and Timmermans, 2014; Giaimo et al. 2010; Greaves et al., 2010). Such surveys present the most probable activity to respondents and then ask them to check the correctness and to fill in the details of the true activity, all of which is usually conducted using a portable electronic device. The present study instead adopted an unsupervised model to recognize hidden activities behind a smart-card holder's trip chain.

The proposed unsupervised model incorporated two critical tasks in imputing activities of smart-card data. That is, clustering activities was done simultaneously with deriving both membership probabilities for each cluster and transition probabilities

* Corresponding author. Tel.: +82163058192.
*E-mail addresses:* a9yness@gmail.com (G. Han), kmsohn@cau.ac.kr (K. Sohn).

http://dx.doi.org/10.1016/j.trb.2015.11.015
0191-2615/© 2015 Elsevier Ltd. All rights reserved.

between clusters. Kroesen (2014) suggested a similar methodology, which is referred to as latent transition analysis, for both clustering travelers and deriving the transition probabilities between clusters. He used panel data to estimate the transition probabilities between traveler groups. If activity chains are viewed as consecutive panel data, the latent transition analysis can be directly applied to imputing the smart-card data. However, to the best of our knowledge, an unsupervised machine-learning technology that fully integrates both clustering and transition models has rarely been used when imputing the missing activities of human mobility data.

The proposed model requires neither labeled data for training nor subsequent measurements such as prompted-recall surveys. Instead, this model utilizes only smart-card and aggregate land-use data. The latter provides additional proxy variables for hidden activities. The spatial and temporal information of each activity within a trip chain is elicited from the smart-card data, and the building floor areas categorized for each land-use type are obtained from the national taxation data in GIS format. The present study is in parallel with many studies that use land-use data to deduce the trip purpose of GPS-based location data (Wolf et al., 2001; Stopher et al. 2007; Stopher et al. 2008; Bohte and Maat, 2009; Elango and Guensler, 2010).

GPS-based technology provides location data reported in short time intervals, which can be used to track the trajectory of travelers, but smart-card data contains only stop (or station) locations with time stamps when a passenger boards and alights. If, however, the latter form of information is incorporated with the actual history of transit operations, the exact route that a passenger used can be recognized. There is a distinction between the GPS-based trajectory data and the smart-card data. An individual trajectory identified by smart-card data is error-free at the spatial level of stops and stations. On the other hand, the GPS-based travel data must not circumvent errors in identifying locations, particularly in densely developed urban areas. Of course, the smart-card data also has an intrinsic problem wherein the exact location of activity is unknown, although it could be guessed to be within a certain distance from the identified stop or station. Furthermore, the data cannot offer information about activities before boarding or after alighting transit modes. The smart-card data has another drawback wherein only the trip chains of transit users are covered. Trips made by motorists and mix-mode users cannot be traced via smart-card data. The activity sequence of motorists could be inferred by incorporating ever-increasing GPS-based data with car navigation technologies. The contribution of the present study, however, is confined to inferring the activities of transit users. Annotating smart-card data with activity types based on an unsupervised machine-learning model was the ultimate objective of the present study.

The proposed methodology must be distinguished from existing attempts to impute the motivations behind human mobility data. While previous researchers have focused on deducing the true activity type of a specific traveler, the present study was concentrated on the probability of choosing the next activity given that the current activity is known. Researchers in the field of artificial intelligence are delving into identifying the exact trip purpose of each specific traveler within a small sample (Kohla and Meschik, 2013; Rasmussen et al., 2013; Bohte and Maat, 2009; Itsubo and Hato, 2006; Kelly et al. 2013). On the other hand, in a dimension of transportation planning and policy-making, it is meaningless to estimate the exact trip purpose of a specific individual. Rather, it is sufficient to know the probability that a traveler will have a certain purpose under certain conditions. The probability could then be used to synthesize the sequence of activities for a trip chain elicited from smart-card data, which could reproduce an imaginary population that is as close as possible to the real population. This synthesized population could provide good input for an activity-based demand-forecasting model.

An activity-based model has recently been spotlighted as an alternative to the conventional transportation demand analysis approach (Yang et al., 2014; Chow, 2014; Arentze and Timmermans, 2000; Bowman and Ben-Akiva, 2001), but it has the burden of validating the result directly against observed data. Recently, Liu et al. (2013) verified that mobile phone data could be a good candidate for observed data for the validation. Trip chains from smart-card data with the imputed trip purposes also could facilitate the validation of an activity-based travel analysis model.

The present study is formatted as follows. The next section will introduce a continuous hidden Markov model and will suggest how to apply it to imputing the trip purposes of smart-card data. The solution algorithm for the model will be addressed in the third section. In the fourth section, the nature of smart-card data of the Seoul metropolitan area will be described, and how to choose a sample to train the proposed model will be addressed. The training result of the proposed model will be discussed and validated in the fifth section. Finally, in the last section, conclusions will be drawn and further possible improvements of the model will be proposed.

## 2. Continuous hidden Markov model (CHMM) for activity imputation

### 2.1. Overview of CHMM

A Markov chain and a Gaussian mixture are fused together to form a continuous hidden Markov model (CHMM). A Markov chain accounts for the state transition between two consecutive activities of a traveler. As the title implies, determining a state is dependent only on its previous state. In the present study, a state corresponds to the missing activity of a transit user such as home, work, maintenance, or personal business. A memoryless transition of states has been widely adopted when estimating the trip purpose (Leszczyc and Timmermans, 2002; Goulias, 1999; Serfozo, 1979). Recently, Allahviranloo and Recker (2013b) extended this hypothesis, so that a next transition could be affected by the state history including the current state. For this extension, the typical Markov chain was discarded and a support vector machine (SVM) was adopted in order to recognize activity patterns, with the past states included in the input feature vector. The SVM required labeled data for training as a representative

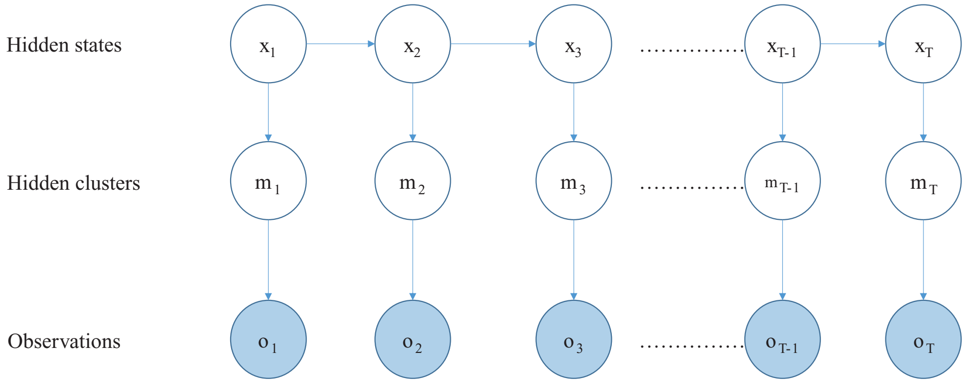

**Fig. 1.** The mechanics of determining states and observations behind a CHMM.

supervised machine-learning tool. On the other hand, the present study adhered to the conventional assumptions of the Markov process, incorporating it with an unsupervised machine-learning model. The title “hidden Markov” stems from the fact that the sequence of activities of a trip chain are unobservable.

For a CHMM, a Gaussian mixture model is responsible for the other side of a Markov chain. Each hidden state of an episode has the output probability of belonging to a cluster in the feature space. Each cluster is also unobservable, and the number of hidden clusters should be given in advance. Some researchers called these clusters the second-level hidden states (Movellan, 2003). There are two options when dealing with the hidden clusters. First, each state could be assumed to have an independent set of clusters. Although this specification has the advantage of addressing the unique properties of each state, there is a burden of computation time. The other model specification allows all hidden states to share a set of clusters. The present study adopted the latter convention to both save the computation time and secure the parsimony of the model specification.

Feature variables are only observed data in a CHMM. The number of feature variables determines the dimension of the feature space for a Gaussian mixture model in a CHMM. In the present study, a feature vector is composed of 6 feature variables: the start and duration times of activity and 4 land-use characteristics around activity locations (stops or stations). The mixture model clusters feature vectors into a predefined number of components. A CHMM then identifies the mean (or centroid) of each cluster, the variance and covariance in the features of each cluster, the transition probabilities between states, and the emission probabilities that each state is linked to each cluster. To avoid transportation researchers’ misunderstanding, the original term ‘emission’ will be changed to ‘membership’.

The procedure of imputing the activity sequence of trip chains based on a CHMM is summarized as follows. First, the sample trip chains are extracted from smart-card data. Second, the number of potential activities is determined. At this stage, there is no information about mapping imaginary activities with physical activities. Third, the possible number of clusters is determined. Fourth, feature data are collected for every hidden activity within the sequence of activities for trip chains in the sample (see Fig. 1). The fifth step is to implement a solution algorithm for estimating the parameters of a CHMM. The sixth step is to characterize clusters and to map each state with them by investigating the estimated parameters. Last, using the parameters, the most probable sequence of activities is imputed for trip chains. The details of the last three steps will be addressed in the next two sub-sections.

### 2.2. Formulation of a CHMM

A CHMM is constructed of state and observed variables. The number of possible values that a state variable can have should be determined in advance prior to setting up the model. Eq. (1) denotes the set of probabilities that an initial state is a specific activity.

$$\pi = \{\pi_i\} = \{P(x_1 = i)\}, \text{for } \; i = 1, \dots, N \tag{1}$$

where $x_1$ denotes the initial state variable of the activity sequence for a trip chain, $N$ is the number of possible activities that can be taken by a state variable, $i$ represents the $i^{th}$activity out of the entire $N$ of possible activities, $\pi_i$ is the probability that the first state would be a $i^{th}$ activity, and $\boldsymbol{\pi}$ is a vector of the initial probabilities.

Eq. (2) represents the matrix of transition probabilities between two consecutive states. In the present model, a trip chain is converted to a state sequence that follows the Markov process. That is, the state of an activity within a trip chain is assumed to depend solely on the state of the previous activity, regardless of past history.

$$\mathbf{A} = \{a_{ij}\} = \{P(x_t = j | x_{t-1} = i)\}, \; \text{for } \; i = 1, \dots, N, \text{and } \; j = 1, \dots, N \tag{2}$$

where $x_t$ denotes the $t^{th}$ state of the sequence of activities for a trip chain, $a_{ij}$ is the transition probability that the $t^{th}$ state selects the activity $j$ when the previous $(t-1)^{\text{th}}$ state was given as the activity, $i$, and $\mathbf{A}$ is a $N \times N$matrix that consists of the transition probabilities.

Eq. (3) stands for the output probability [$=b_i(\mathbf{o}_t)$] that $\mathbf{o}_t$ will be observed in the state $i$, which takes the form of a Gaussian mixture model.

$$b_i(\mathbf{o}_t) = \sum_{k=1}^{K} g_{ik} f(\mathbf{o}_t | \mu_{ik}, \Sigma_{ik}), \; for \;\; i = 1, \ldots, N \tag{3}$$

where $\mathbf{o}_t$ is an observed feature vector for the $t^{th}$ state of the sequence, $K$ is the number of hidden clusters in a feature space, $g_{ik}$ is the membership probability that an observation comes from the $k^{th}$ cluster when the current state is the activity $i$, $\boldsymbol{\mu}_{ik}$ is a mean feature vector of the $k^{th}$ cluster of the activity $i$, $\boldsymbol{\Sigma}_{ik}$ is a variance-covariance matrix of the $k^{th}$ cluster of the activity $i$, and $f(\mathbf{o}_t|\boldsymbol{\mu}_{ik}, \boldsymbol{\Sigma}_{ik})$ is a Gaussian probability density function.

For brevity, every state is assumed to share a common set of clusters [see Eq. (4)]. The $N \times K$ weight matrix ($G$) for a Gaussian mixture model can be interpreted as Eq. (5), which will hereafter be referred to as the membership probability matrix.

$$\mu_{ik} = \mu_k \text{ and } \boldsymbol{\Sigma}_{ik} = \boldsymbol{\Sigma}_k, \;\; for \;\; i = 1, \ldots, N \;\; and \;\; k = 1, \ldots, K \tag{4}$$

$$\mathbf{G} = \{g_{ik}\} = \{P(m_t = k | x_t = i)\}, \;\; for \;\; i = 1, \ldots, N \;\; and \;\; k = 1, \ldots, K \tag{5}$$

where $m_t$ denotes a hidden cluster in a feature space for the $t^{th}$ hidden state of a sequence.

The hypothetical mechanics behind a CHMM between hidden states and observations can be summarized as follows. For a certain state in the sequence of activities for a trip chain, an activity is determined according to the transition probabilities based on the previous state. The next step is to choose a hidden cluster for the state according to the determined activity and the membership probabilities. Last, based on the determined cluster, observations are drawn from a Gaussian probability distribution with the cluster's mean and variance–covariance. This procedure is represented by the simple diagram in Fig. 1.

As mentioned earlier, the observations in a CHMM are feature vectors that are generated from a multivariate Gaussian distribution. The parameters to be estimated based on the observations are represented by a vector collection [$\lambda = \langle \pi, \mathbf{A}, \mathbf{G}, \{\mu_k\}, \{\boldsymbol{\Sigma}_k\} \rangle$]. The likelihood function of the observation sequence for a trip chain must be set up to estimate the parameters. However, hidden states of activities hamper formulating the likelihood function in a straightforward manner. The likelihood function with hidden or latent variables should be integrated over all possible values of the variables. For a model with discrete hidden variables, the mathematical integration is reduced to a simple summation (or average) across all possible values of the hidden variables. Eq. (6) denotes the likelihood function [$L(\boldsymbol{\lambda})$] that will be maximized in the next sub-section. The first line of the equation shows that a marginal probability is factorized based on Bayes' theorem. The second line is derived from both the independence assumption of observations and the memorylessness assumption of the Markov process. The last line is derived easily from the structure of a CHMM, as described above.

$$\begin{aligned} L(\lambda) = P(\mathbf{o}_1, , , \mathbf{o}_T | \lambda) &= \sum_{all\ possible\ x_1,,,x_T} P(\mathbf{o}_1, , , \mathbf{o}_T | x_1, , , x_T, \lambda) p(x_1, , , x_T | \lambda) \\ &= \sum_{all\ possible\ x_1,,,x_T} \left( \prod_{t=1}^{T} p(\mathbf{o}_t | x_t, \mathbf{G}, \{\mu_k\}, \{\boldsymbol{\Sigma}_k\}) \right) \left( \prod_{t=1}^{T} p(x_t | x_{t-1}, \mathbf{A}) \right) \\ &= \sum_{all\ possible\ x_1,,,x_T} \left( \prod_{t=1}^{T} \left( \sum_{k=1}^{K} g_{x_t k} f(\mathbf{o}_t | \mu_k, \boldsymbol{\Sigma}_k) \right) \right) \left( \prod_{t=1}^{T} a_{x_{t-1} x_t} \right) \end{aligned} \tag{6}$$

where $T$ is the length of the activity sequence for a trip chain.

The likelihood function could be extended to accommodate multiple trip chains, each of which has a different length of the activity sequence. Under the assumption of independence between observations, Eq. (7) denotes the extended version of Eq. (6) wherein the superscript $l$ stands for a specific trip chain.

$$\hat{L}(\lambda) = \prod_{l=1}^{M} \left\{ \sum_{all\ possible\ x_1,,,x_{T^l}} \left( \prod_{t=1}^{T^l} \left( \sum_{k=1}^{K} g_{x_t k} f\left(\mathbf{o}_t^l | \mu_k, \Sigma_k\right) \right) \right) \left( \prod_{t=1}^{T^l} a_{x_{t-1} x_t} \right) \right\} \tag{7}$$

where $\hat{L}(\lambda)$ represents the extended likelihood function, $M$ is the number of trip chains, $T^l$ is the length of the activity sequence for the $l^{th}$ trip chain, and $\mathbf{o}_t^l$ represents an observed feature vector for the $t^{th}$ state within the activity sequence for the $l^{th}$ trip chain.

The extended likelihood function is more difficult to handle, since it contains the sum of individual likelihoods over every possible assignment of state variables. Maximizing the extended likelihood with hidden variables cannot be done when using a typical gradient method such as the Newton–Raphson algorithm. A more elaborated solution algorithm will be introduced in the next sub-section.

### 2.3. Solution algorithms for a CHMM

Regarding a CHMM, 3 typical problems are of interest. The first problem is how to compute the probability that a particular sequence of states is observed, given that the hidden states and parameters of the model are known. This problem can be solved using the forward–backward algorithm. The second problem is to estimate the optimal parameters given that a sequence of observations or a set of multiple observed sequences (or multiple trip-chains) are known. This problem can be resolved using the Baum–Welch algorithm. When implementing the Baum–Welch algorithm, the forward–backward algorithm is utilized to compute the probability of a certain state or a certain pair of two consecutive states, given that all observations are available. The third problem is to derive the most likely sequence of hidden states, given that the sequence of observations and the model parameters are known. The Viterbi algorithm is necessary to solve this problem. The present study used the former two algorithms to estimate the parameters of a CHMM, and applied the Viterbi algorithm to imputing the sequence of activities for a trip chain based on both observations and the estimated parameters.

It is impossible to handle the summation of individual probabilities across every possible activity sequence, when either the length of activity sequence or the number of possible hidden states becomes larger. The forward-backward algorithm is a key to tackling this problem. The algorithm facilitates the computation of the probability of either a hidden state or a consecutive pair of hidden states when a full sequence of observations is given. Backward and forward variables [$\alpha_t(j)$ and $\beta_t(i)$] must be created in order to compute the probability. More specifically, these variables provide an easy way to compute both $P(x_t|\mathbf{o}_1,,, \mathbf{o}_T, \boldsymbol{\lambda})$ and $P(x_t, x_{t+1}|\mathbf{o}_1, , , \mathbf{o}_T, \lambda)$, which are included in the Baum–Welch algorithm to train a CHMM. The forward variable, $\alpha_t(j)$, represents $P(x_t = j, \mathbf{o}_1, , , \mathbf{o}_t|\lambda)$, and the backward variable, $\beta_t(i)$, represents $P(\mathbf{o}_{t+1}, , , \mathbf{o}_T|x_t = i, \lambda)$. Both variables are computed recursively according to the procedures of Eqs. (8) –(11) . As a result, $P(x_t|\mathbf{o}_1,,, \mathbf{o}_T, \boldsymbol{\lambda})$ and $P(x_t, x_{t+1}|\mathbf{o}_1, , , \mathbf{o}_T, \lambda)$ are computed recursively by the forward and backward variables [see Eqs. (12) and (13)]. Details of the derivation can be found in the literature (Cappe et al., 2005; Zraiaa, 2010).

**Forward recursion procedure:**

1. Initially,

$$\alpha_1(i) = \pi_i b_i(\mathbf{o}_1), \, for \, i = 1, ..., N \tag{8}$$

2. For $t = 2, 3, ..., T$,

$$\alpha_t(j) = b_j(\mathbf{o}_t)\sum_{i=1}^{N}\alpha_{t-1}(i)a_{ij}, \, for \, j = 1, ..., N \tag{9}$$

**Backward recursion procedure:**

1. Initially,

$$\beta_T(i) = 1, \, for \, i = 1, ..., N \tag{10}$$

2. For $t = T - 1, T - 2, ..., 1$,

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}b_j(\mathbf{o}_{t+1})\beta_{t+1}(j), \, for \, j = 1, ..., N \tag{11}$$

**Forward–backward computation**

$$\begin{aligned} P(x_t = i|\mathbf{o}_1, , , \mathbf{o}_T, \lambda) &\propto P(x_t = i, \mathbf{o}_1, , , \mathbf{o}_T|\lambda) = P(x_t = i, \mathbf{o}_1, , , \mathbf{o}_t|\lambda)P(\mathbf{o}_{t+1}, , , \mathbf{o}_T|x_t = i, \lambda) \\ &= \alpha_t(i)\beta_t(i), \, for \, i = 1, ..., N \, and \, t = 1, ..., T-1 \end{aligned} \tag{12}$$

$$\begin{aligned} P(x_t = i, x_{t+1} = j|\mathbf{o}_1, , , \mathbf{o}_T, \lambda) &\propto P(x_t = i, x_{t+1} = j, \mathbf{o}_1, , , \mathbf{o}_T|\lambda) \\ &= P(x_t = i, x_{t+1} = j, \mathbf{o}_1, , , \mathbf{o}_{t+1}|\lambda)P(\mathbf{o}_{t+2}, , , \mathbf{o}_T|x_t = i, x_{t+1} = j, \lambda) \\ &= \alpha_t(i)a_{ij}b_j(\mathbf{o}_{t+1})\beta_{t+1}(j) \\ &\quad for \, i = 1, ..., N, \, j = 1, ..., N, \, and \, t = 1, ..., T-1 \end{aligned} \tag{13}$$

The Baum–Welch algorithm, a training methodology for a CHMM, is a specific instantiation of the more general expectation-maximization (EM) algorithm. The EM algorithm was developed as a powerful tool for solving a maximization problem when latent variables are involved (Dempster et al., 1977). The algorithm switches the original maximization of the log-likelihood function that contains multiple integrations (or summations) across all the possible latent variables into a simple recursive procedure. In summary, the Baum–Welch algorithm is an iterative procedure for estimating $\boldsymbol{\lambda}$ from only $\{\mathbf{o}_1,,, \mathbf{o}_T\}$. At each iteration, a proxy function $[Q(\boldsymbol{\lambda}, \boldsymbol{\lambda}^s)]$ is maximized instead of maximizing the original log-likelihood function. The proxy function can be interpreted as a weighted sum of the log-likelihood, wherein the weight represents the probability that hidden states are observed conditional on the parameters that were estimated in the previous iteration. The weight is originally set up as $P(x_1,,, x_T|\mathbf{o}_1,,, \mathbf{o}_T, \boldsymbol{\lambda}^s)$, but switches to $P(x_1,,, x_T, \mathbf{o}_1,,, \mathbf{o}_T|\boldsymbol{\lambda}^s)$ using the Bayes' theorem $[P(X|Y) = P(X, Y)/P(Y) \propto P(X, Y)]$. The Baum–Welch algorithm can be described simply as repeating the following expectation and maximization steps. Eqs. (14) and (15) are for a single trip chain, and Eqs. (16) and (17) are for multiple trip chains. The weight is computed depending on the previously derived parameters ($\boldsymbol{\lambda}^s$). Thus, only the log-likelihood in the proxy function includes decision parameters to be estimated in the maximization step. Repeating the procedure guarantees the convergence according to the theory of the conventional EM algorithm.

**For a single trip chain**

1. **Expectation step**:

$$\text{Compute}\, Q(\lambda, \lambda^s) = \sum_{\text{all possible } x_{1,,,}x_T} \log[P(\mathbf{o}_1,,,\mathbf{o}_T|\lambda)]P(x_1,,,x_T,\mathbf{o}_1,,,\mathbf{o}_T|\lambda^s) \tag{14}$$

2. **Maximization step**:

$$\text{Set}\, \lambda^{s+1} = \arg\max_{\lambda} Q(\lambda, \lambda^s) \tag{15}$$

**For multiple trip chains (Extended case)**

1. Expectation step:

$$\text{Compute}\, Q(\lambda, \lambda^s) = \sum_{l=1}^{M} \sum_{\text{all possible } x_{1,,,}x_{T^l}} \log\left[P\left(\mathbf{o}_1^l,,,\mathbf{o}_{T^l}^l|\lambda\right)\right]P\left(x_1,,,x_{T^l},\mathbf{o}_1^l,,,\mathbf{o}_{T^l}^l|\lambda^s\right) \tag{16}$$

2. Maximization step:

$$\text{Set}\, \lambda^{s+1} = \arg\max_{\lambda} Q(\lambda, \lambda^s) \tag{17}$$

The difference between the Baum–Welch algorithm and the conventional EM algorithm is that the former has constraints on the parameters to be estimated. As mentioned earlier, the parameters of a CHMM contain a vector of initial state probabilities and two matrices of transition and membership probabilities. The former vector's elements, as well as each column of the transition matrix and each row of membership matrix, should sum up to 1. Thus, in the maximization step, the Lagrangian relaxation for the original proxy function is necessary to accommodate the constraints [see Eq. (18)].

$$\mathcal{L}(\lambda, \lambda^s) = Q(\lambda, \lambda^s) - u_\pi\left(\sum_{i=1}^{N}\pi_i - 1\right) - \sum_{i=1}^{N}u_i^A\left(\sum_{i=1}^{N}a_{ij} - 1\right) - \sum_{i=1}^{N}u_i^B\left(\sum_{k=1}^{K}g_{ik} - 1\right) \tag{18}$$

where $\mathcal{L}(\lambda, \lambda^s)$ is the Lagrangian relaxation for the original proxy function, and $\langle u_\pi, u_1^A, ..., u_N^A, u_1^B, ..., u_N^B\rangle$ is a set of Lagrangian multipliers, each of which corresponds to its corresponding constraint.

The first-order condition of the Lagrangian function [Eq. (18)] is that the derivative of the function with respect to each original parameter ($\pi_i, a_{ij}, g_{ik}, \mu_{kd}, \sigma_{kd_1d_2}$), and to each Lagrangian multiplier, should be zero, which offers an incumbent parameter solution set in each iteration of the Baum–Welch algorithm. For details of the derivation, readers can refer to tutorials available on the Internet (Bishop, 2006; Movellan, 2003). Only the final procedures are introduced and interpreted in an intuitive manner, so that transportation researchers can simply track them to accommodate their own activity imputation work for various types of human mobility data.

The optimal parameters ($\hat{\pi}_i, \hat{a}_{ij}, \hat{g}_{ik}, \hat{\mu}_{kd}, \hat{\sigma}_{kd_1d_2}$) in an iteration of the Baum–Welch algorithm, which are derived from the first-order condition of the Lagrangian function, can be summarized as follows.

$$\hat{\pi}_i = \frac{\sum_{l=1}^{M} P(x_1 = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}{M},\ for\, i = 1, ..., N \tag{19}$$

$$\hat{a}_{ij} = \frac{\sum_{l=1}^{M}\sum_{t=1}^{T^l-1} P(x_t = i, x_{t+1} = j|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}{\sum_{l=1}^{M}\sum_{t=1}^{T^l-1} P(x_t = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)},\ \text{for}\, i = 1, ..., N\, \text{and}\, j = 1, ..., N \tag{20}$$

$$\hat{g}_{ik} = \frac{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}, \text{for } i = 1, ..., N \text{ and } k = 1, ..., K \tag{21}$$

$$\hat{\mu}_{kd} = \frac{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s) o_{td}^l}{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}, \text{for } k = 1, ..., K \text{ and } d = 1, ..., D \tag{22}$$

where, $\mathbf{o}_t^l = (o_{t1}^l, ..., o_{tD}^l)'$, $\hat{\mu}_k = (\hat{\mu}_{k1}, ..., \hat{\mu}_{kD})'$, and$D$ denotes the dimension of observed feature vector.

$$\begin{gathered}\hat{\sigma}_{kd_1d_2} = \frac{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)(o_{td_1}^l - \hat{\mu}_{kd_1})(o_{td_2}^l - \hat{\mu}_{kd_2})}{\sum_{l=1}^{M}\sum_{t=1}^{T^l} P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)}, \text{for} \\ k = 1, ..., K, d_1 = 1, ..., D, \text{ and } d_2 = 1, ..., D\end{gathered} \tag{23}$$

where,

$$\hat{\mathbf{\Sigma}}_k = \begin{pmatrix} \hat{\sigma}_{k1}^2 & \cdots & \hat{\sigma}_{k1D} \\ \vdots & \ddots & \vdots \\ \hat{\sigma}_{kD1} & \cdots & \hat{\sigma}_{kD}^2 \end{pmatrix}$$

Eqs. (19)–(23) contain 3 probability terms that cannot be computed directly when the number of states or clusters gets larger. The forward–backward algorithm is inevitable to compute the three probabilities: $P(x_t = i, x_{t+1} = j|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)$,$P(x_t = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)$, and $P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s)$. The computation method for the 3 probabilities can be summarized as follows. Eqs. (24) and (25) are simply scaled versions of Eqs. (12) and (13), respectively. Eq. (26) is self-evident, and accounts for the probability of a certain pair of a hidden state and a cluster, given that all observations are available. For details of the computations, refer to Movellan (2003) and Zraiaa (2010).

$$P(x_t = i, x_{t+1} = j|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s) = \frac{\alpha_t^l(i) a_{ij} \beta_{t+1}^l(j) b_j(\mathbf{o}_{t+1}^l)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t^l(i) a_{ij} \beta_{t+1}^l(j) b_j(\mathbf{o}_{t+1}^l)} \tag{24}$$

$$P(x_t = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s) = \frac{\alpha_t^l(i)\beta_t^l(i)}{\sum_{i=1}^{N} \alpha_t^l(i)\beta_t^l(i)} \tag{25}$$

where, $\alpha_t^l(i)$and $\beta_t^l(i)$ stand for the forward and backward variables computed for the $l^{\text{th}}$ trip-chain, respectively.

$$P(x_t = i, m_t = k|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s) = P(x_t = i|\mathbf{o}_1^l, ..., \mathbf{o}_{T^l}^l, \lambda^s) \frac{g_{ik} f(\mathbf{o}_t^l|\mu_k, \Sigma_k)}{b_i(\mathbf{o}_t^l)} \tag{26}$$

Eqs. (19)–(26) are all that transportation researchers should utilize to estimate the parameters for imputing the hidden activities of their own travel data. All the equations are simple and some of them take the recursive form. Repeating the computations yields robust parameter estimates for a CHMM: the probabilities of initial state, the most likely transition and membership probabilities, the mean, and variance information about each cluster of feature vectors. All proposed algorithms were coded using R programming language. The full procedure for the Baum–Welch algorithm is summarized as follows:

```
1. Randomly initialize parameters of $\pi_i, a_{ij}, g_{ik}$.
2. For each iteration for EM algorithm.
  2.1. Implement forward–backward algorithm [Eqs. (8)–(11)] for each trip chain to compute $\alpha_t^l(j)$ and $\beta_t^l(i)$.
  2.2. Compute three types of probabilities using Eqs. (24)–(26).
  2.3. Compute the incumbent parameters using Eqs. (19)–(23).
  2.4. Convergence test: If the incumbent solutions are different from those in the previous iteration, then repeat the
       loop. Otherwise, escape the loop and stop the procedure.
```

The initial values for the three parameter sets were determined randomly. The initial values for transition and membership matrices were chosen through the following two steps. That is, elements of each column for transition matrix (or each row of membership matrix) were chosen randomly, and then normalized to sum up to 1. The initial values for the probabilities of an initial state were also determined in the same manner. A simple criterion was adopted for the convergence of the Baum–Welch algorithm. The algorithm stopped if every estimated parameter in the current iteration was not different from that in the previous iteration. The threshold for the difference was set as $10^{-7}$. The threshold was sufficiently small, since most parameters to be estimated were probabilities ranging from 0 to 1.

The greatest strength of a CHMM is that it belongs to the category of unsupervised machine-learning tools. A CHMM requires no labeled data for the calibration procedure. The parameter estimates can be obtained without fitting the model to the observed activity sequences. Usually, the activity sequence for a trip chain cannot be observed without an additional survey, which entails considerable cost. Instead, a CHMM requires two intuitive processes in order to characterize the estimated clusters and to link each state to them. The former characterization is based on the means and variances of cluster centroids in the feature space. The latter mapping of a latent activity onto the relevant clusters is done by using the estimated membership probabilities. These processes seem unfamiliar to transportation researchers who are accustomed to calibrating models with observed data. However, researchers should understand the power of unsupervised machine-learning models, since it is clear that data technology (DT) is evolving in the direction of the target wherein the data itself reveals all. The future of DT will be governed by unsupervised machine-learning tools. In the same context, it is recommended that the typical activity types most researchers have adopted thus far should be reconsidered from a different point of view. All previous studies have confined human activities to several predefined types. Self-clustered activities could offer novel insight into understanding human mobility, even though they have neither a clear description nor an exact title. In other words, human activities could be clustered and interpreted without any fixed conceptual framework.

Once the Baum–Welch algorithm is used to estimate the parameters of a CHMM, the Viterbi algorithm can be used to derive the most probable sequence of activities for a trip chain based on both the estimated parameters and the observed feature data. The Viterbi algorithm also utilizes a recursive computation to derive the optimal state sequence for a trip chain. How to derive the algorithm are not also introduced here, since the objective of the present study was to provide a straightforward procedure that transportation researchers and practitioners could follow without confusion. Readers who are interested in the theory of the algorithm can refer to the original paper (Viterbi, 1967). The resultant process of the Viterbi algorithm can be summarized as follows:

**Initialization:** for all $j = 1, ..., N$

$$\delta_1(j) = \pi_j b_j(\mathbf{o}_1),\ \psi_1(j) = 0 \tag{27}$$

**Recursion:** for all $t = 2, ..., T$ and all $j = 1, ..., N$

$$\delta_t(j) = \max_i(\delta_{t-1}(i)a_{ij})b_j(\mathbf{o}_t),\ \psi_t(j) = \arg\max_i(\delta_{t-1}(i)a_{ij}) \tag{28}$$

**Termination:**

$$P^*(\mathbf{o}_1, ..., \mathbf{o}_T|\lambda) = \max_j(\delta_T(j)),\ x_T^* = \arg\max_j(\delta_T(j)) \tag{29}$$

**Backtracking of optimal state sequence:**

$$x_t^* = \psi_{t+1}(x_{t+1}^*),\ for\, t = T-1, T-2, ..., 1 \tag{30}$$

where, $\delta_t(j)$ is the probability of the most probable state sequence responsible for the first $t$ observations that have $j$ as the final state, $\psi_t(j)$ is the state that results in $\delta_t(j)$, $P^*(\mathbf{o}_1, ..., \mathbf{o}_T|\lambda)$ is the optimal probability for all observations, and $x_t^*$ is the optimal state at $t$.

## 3. Data preparation for training a CHMM

The automatic fare collection system in the Seoul metropolitan area can provide time stamps and the corresponding station (or stop) IDs, when a passenger boards, transfers, and alights a transit mode. The data, however, have not been fully utilized for the purpose of transportation planning, since privacy issues are yet to be addressed. The data have rarely been collected with passenger identification information kept for long-term periods. Data for only two weekdays were available in the present study to train the developed model. The raw data covered fare transactions during two specific weekdays (16/10/2013 and 17/10/2013), from which samples of trip chains were extracted.

The Seoul metropolitan government allows for free transfers unless the time spent between two different lines or modes exceeds 30 min. This rule clearly distinguishes an activity from simple waiting for transfers. Of course, there could be an activity shorter than 30 min or a passenger could have waited for more than 30 min for transfers, but these remote possibilities were ignored in the present study. After tracing the data for 2 days, only the second day's trip chains were sampled. The first-day trip data were used only for establishing that the first activity of a trip chain was an overnight activity that was most likely a home stay. We chose only passengers who started and terminated their trips at the same locations (approximately in the same TAZ) for both days. The last activity of the trip chains included no information about the duration time, since the third-day trip data were absent. Thus, the last activity's duration time was assumed to be identical to the first. This stopgap measure would have been unnecessary if multi-day smart-card data had been available. Although smart-card holders were categorized as adults, senior citizens, or students, only adults were included in the training sample, since the latter two took up a small proportion and might have shown different behaviors due to the advantage of free or discounted fares.

Featured variables encompassed the start and duration times of activity and the land-use characteristics around activity locations. An activity was regarded as an episode between two consecutive trips within a trip chain. The start and duration times of each activity were extracted from the trip chain data, whereas the land-use data were obtained from the other data source. The Korean government offers data for the floor area of each individual building, which was established for taxation purposes. The floor areas are categorized as either residential, commercial, office, or other. The floor area data for only 25 boroughs of Seoul city were available for 2008 in SHAPE, which is an open GIS format. Although a more recent taxation data set from 2013 was also released, it has not yet been linked to spatial data. It should thus be noted that the present study has a temporal gap between the smart card data of 2013 and the land-use data of 2008.

The exact location of the activity of a smart-card holder is unknown. Instead, activity locations were identified at the level of bus stops or metro stations. The present study did not identify the activity type of specific individuals, but focused instead on deriving the probability that an inter-trip episode would take a certain activity type. The land-use characteristics within a 200 m radius around a bus stop and a 400 m radius around a metro station were collected to establish the feature variables of activities. The floor area around every stop and every station was computed in advance for the four land-use types using a GIS buffering technology, and was reserved for training the developed model. Consequently, the dimension of a feature vector was set at 6. Two of them were for the start and duration times of an activity, and the remaining four were for the land-use characteristics. The former two time variables were scaled to range from 0 to 1 to facilitate the computation. When applying the floor data to the model, they were rescaled so that stops and stations could have the same influential area and then were standardized across stops and stations so that the distribution of floor areas for each land-use could have a zero mean and unity variance.

There is no rigorous methodology that can recognize the true number of hidden clusters in the feature space. The present study varied the number of clusters to investigate how well the clustering results matched prior expectations and/or common sense. As a result, 8 potential clusters were adopted as the most plausible number of clusters. The number of hidden states might differ from the number of clusters. After investigating the 8 clusters, 4 hidden states were found to be appropriate (see the next section for details). It is well known that a hidden Markov model could not always estimate membership probabilities with a small variance and nonzero values (Rank and Pernkopf 2004). Based on the numbers of 8 and 4, the model parameters were estimated in a very stable manner without failure. Further studies could adopt either an information criteria or a Bayesian approach as an alternative way to determine the optimal numbers of hidden states and clusters, which have already been applied to choosing the most plausible number of components for a mixture model (Richardson and Green, 1997; Park et al., 2010).

The smart card data of the Seoul metropolitan area contains more than 20 million transactions during a weekday. The number of trip chains elicited from the smart card data for 2 weekdays was tantamount to 11.76 million. After screening the data according to the criteria established above, and including only trip chains that had activities within a boundary of 25 boroughs of Seoul city, the number of trip chains was reduced to 306,766. The distribution of trip chains by the length of their activity sequence is shown in Table 1. The maximum length was found to be 6. The present study used a small sample, so that the model could be

**Table 1**
Distribution of trip chains according to the number of episodes.

| # of episodes (the length of activity sequence) | 3 | 4 | 5 | 6 | Total |
|---|---|---|---|---|---|
| Population trip chains | 271,971<br>(88.65%) | 23,829<br>(7.76%) | 9428<br>(3.07%) | 1538<br>(0.5%) | 306,766<br>(100%) |
| Sample trip chains | 1268<br>(88.92%) | 113<br>(7.92%) | 39<br>(2.74%) | 6<br>(0.42%) | 1426<br>(100%) |

trained in a realistic computation time by using a PC. The reason the sample proportion was slightly different from the population proportion in Table 1 was because the land-use data of some chosen trip chains were unavailable and thus they were excluded from the sample. In fact, there were no missing values in the raw data for building floor areas since they were made for taxation purposes. However, a considerable amount of data was discarded when being matched with the existing imperfect spatial data.

## 4. Results and discussion

### *4.1. Model training results*

Implementing the Baum–Welch algorithm based on the sample data, we obtained estimates for the 4 parameter sets: the probabilities of initial state, the transition probabilities, the membership probabilities, and the mean and variance (including co-variances) of each cluster in the feature space. First, we focused on the latter to characterize the resultant clusters and to confer plausible titles to them. Proper descriptions were given to the imaginary clusters that were recognized by the proposed model. They were characterized by considering its mean vector and variance-covariance matrix. The resultant characterizations are listed in Table 2. Since transit users usually accommodate miscellaneous activities within walking distance of their main activity venue, relatively short activities were not apparent from smart-card data. This is an intrinsic handicap of the proposed model, because it totally depends on the location resolution at the stop or station level. Clusters that were assigned the title flexible activity had start and duration times that fluctuated more than those titled a typical activity, and also took place in the area of mixed land-use. For example, two clusters corresponding to the flexible home stay had larger standard deviations in both start and duration times, and their dominant land-use was mixed rather than residential.

The next step was to match the derived clusters to hidden activities (or states) based on the estimated membership probabilities. Table 3 shows the membership probabilities for each activity. For clarity, the dominant membership probabilities are highlighted in bold font. According to these probabilities, the first activity (Activity 1) was associated with clusters 1 and 8. Both were linked to flexible home stay, since their start and duration times fluctuated more than the typical home stay, which was represented by the sixth and seventh clusters. The second activity (Activity 2) was linked to clusters 2 and 4, which were associated with out-of-home activities. Flexible out-of-home activity differed from typical work in that it started later and showed a large variance in both start and duration times. That activity also had a modest probability for the fifth cluster that represented typical work in a mixed land-use area. The third activity (Activity 3) was governed by clusters 3 and 5, which were closely associated with typical work in commercial, office, or mixed land-use areas. The last activity (Activity 4) was strongly linked to clusters 6 and 7, which represented a typical home stay in residential areas. In summary, Activity 1 likely belongs to flexible home stays, Activity 2 is likely to encompass flexible out-of-home activities, Activity 3 strongly implies typical work, and Activity 4 could be defined as a typical home stay.

The next step was to investigate the transition matrix to show how the activity sequence for a trip chain is generated. Table 4 shows the transition matrix with a possible title for each activity. The reason the transition probabilities are simple

**Table 2**
Description of the resultant clusters based on $(\hat{\mu}_k, \hat{\boldsymbol{\Sigma}}_k)$.

| Cluster description | Cluster centroid | | | | Most likely title |
|---|---|---|---|---|---|
| | Average start time (standard deviation) | Average duration time (standard deviation) | Average end time | Dominant land-use | |
| Cluster 1: long overnight activity in mixed land-use area of low density with large variance in start and duration times | 7:30 P.M. ( 2:31) | 13 h (2:49) | 8:30 A.M. | Mixed (low density) | Flexible home stay |
| Cluster 2: afternoon activity in commercial/office area with large variance in start and duration times | 4:30 P.M. (4:10) | 7 h (4:57) | 11:30 P.M. | Commercial/office | Flexible out-of-home activity (recreation, societal activity, afternoon work, etc.) |
| Cluster 3: early and long diurnal activity in office/commercial area | 8:30 A.M. (0:56) | 10.5 h (2:04) | 07:00 P.M. | Office/commercial | Typical work |
| Cluster 4: afternoon activity in other land-use area with large variance in start and duration times | 3:30 P.M. (4:53) | 8 h (4:27) | 11:30 P.M. | Others | Flexible out-of-home activity (recreation, societal activity, afternoon work, etc.) |
| Cluster 5: early and long diurnal activity in mixed land-use area | 8:30 A.M. (1:01) | 10.5 h (2:14) | 7:00 P.M. | Mixed | Typical work |
| Cluster 6: long overnight activity in residential area of high density | 8:00 P.M. (1:52) | 12 h (2:00) | 08:00 A.M. | Residential (high density) | Typical home stay |
| Cluster 7: long overnight activity in residential area of low density | 8:00 P.M. (1:43) | 12 h (1:49) | 08:00 A.M. | Residential (low density) | Typical home stay |
| Cluster 8: long overnight activity in mixed land-use area of high density with large variance in start and duration times | 8:00 P.M. (2:44) | 14.5 h (3:42) | 10:30 A.M. | Mixed (high density) | Flexible home stay |

**Table 3**
Estimated membership probabilities ($\hat{g}_{ik}$).

| Land-use | Mixed | Commercial/ office | Office/ commercial | Others | Mixed | Residential | Residential | Mixed |
|---|---|---|---|---|---|---|---|---|
| | Cluster 1 (flexible home stay) | Cluster 2 (flexible out-of-home activity) | Cluster 3 (typical work) | Cluster 4 (flexible out-of-home activity) | Cluster 5 (typical work) | Cluster 6 (typical home stay) | Cluster 7 (typical home stay) | Cluster 8 (flexible home stay) |
| Activity 1 | **0.341** | 0.107 | – | 0.023 | – | – | – | **0.530** |
| Activity 2 | – | **0.290** | 0.159 | **0.313** | 0.239 | – | – | – |
| Activity 3 | – | 0.013 | **0.410** | 0.048 | **0.529** | – | – | – |
| Activity 4 | – | – | – | 0.055 | – | **0.521** | **0.424** | – |

**Table 4**
Estimated transition probabilities ($\hat{a}_{ij}$).

| Transition probabilities | Activity 1 (flexible home stay) | Activity 2 (flexible out-of-home activity) | Activity 3 (typical work) | Activity 4 (typical home stay) |
|---|---|---|---|---|
| Activity 1 (flexible home stay) | – | 1.000 | – | – |
| Activity 2 (flexible out-of-home activity) | 0.771 | 0.116 | – | 0.114 |
| Activity 3 (typical work) | – | 0.117 | – | 0.883 |
| Activity 4 (typical home stay) | – | – | 1.000 | – |

is that the lengths of the trip chains in the sample ranged from 3 to 6, which is very short. The estimated transition patterns were so straightforward that such a complex methodology could be unnecessary. However, if longer trip chains were available from multi-day smart-card data, the transition probability would provide in-depth understanding for use in composing the activity sequence for a trip chain. It should be noted that the trip chains used in the proposed model were extracted from only two consecutive days. Nonetheless, as a whole, the matrix well reflected both priori expectations and intuitions. For example, when the current state was a flexible out-of-home activity (Activity 2), the next state was most likely to be a flexible home stay (Activity 1). An interesting finding was that a flexible out-of-home activity (Activity 2) might switch to another flexible out-of-home activity (Activity 2) with a considerable degree of probability (0.116). This implies that trip chains other than piston trips of home-to-work-to-home, the length of which were longer than 3, included consecutive flexible out-of-home activities. A current typical home stay (Activity 4) could be perfectly paired with the next typical work activity (Activity 3). On the other hand, a current typical work activity (Activity 3) did not perfectly match the next typical home stay (Activity 4). There was a considerable probability (=0.117) that a current typical work activity (Activity 3) would transition to the next flexible out-of-home activity (Activity 2). This implies that passengers might have flexible activities after work before going home. It was rational that a current flexible home stay (Activity 1) perfectly switched to the next flexible out-of-home activity (Activity 2). Of course, if longer trip chains were available, the model would estimate more diverse transition probabilities.

The remaining parameters to be estimated for a CHMM were the initial probabilities for choosing each activity ($\hat{\pi}_i$). Since all trip chains in the sample were chosen such that they started with an overnight stay from the first day to the next day, only two activities (Activity 1 and Activity 4) were estimated to have an initial probability. While the typical home stay covered 62.4% of the initial state, the flexible home stay accommodated the remaining 37.6%.

The last step was intended to generate the most probable sequence of activities for a trip chain based on the parameters estimated above and on the observed feature vectors. This procedure was accomplished using the well-known Viterbi algorithm. It should be noted again that a CHMM is an unsupervised machine-learning tool that requires no labeled data for training. That is, there is no need to calibrate the model based on surveyed activity sequences. The Viterbi algorithm guarantees the estimated activity sequences to be the best reflection of the available observations. Table 5 shows the results of the activity imputation of trip chains in the sample.

**Table 5**
Distribution of the most probable activity sequences in the training sample.

| Activity sequence | Counts (%) | Activity sequence | Counts (%) | Activity sequence | Counts (%) |
|---|---|---|---|---|---|
| 4-3-4 | 810 (56.8) | 1-2-1-2-1 | 11 (0.77) | 4-3-2-2-1 | 1 (0.07) |
| 1-2-1 | 439 (30.79) | 1-2-1-2 | 8 (0.56) | 4-3-2-2 | 1 (0.07) |
| 4-3-2-4 | 53 (3.72) | 1-2-2 | 6 (0.42) | 4-3-2 | 1 (0.07) |
| 1-2-2-1 | 34 (2.38) | 4-3-2-2-4 | 5 (0.35) | 1-2-2-4 | 1 (0.07) |
| 4-3-2-1 | 16 (1.12) | 4-3-2-1-2 | 5 (0.35) | 1-2-1-2-2 | 1 (0.07) |
| 4-3-4-3-4 | 14 (0.98) | 1-2-2-1-2 | 5 (0.35) | Total | 1426 (100) |
| 1-2-4 | 12 (0.84) | 1-2-1-2-4 | 3 (0.21) | | |

### *4.2. Model validation results*

Although the proposed methodology adopted an unsupervised machine learning tool that requires no calibration process based on labeled data, the result should be validated against observed activity data. Liu et al. (2014) shed light on the validation issue by providing a robust method to validate activity-based transportation models based on mobile phone data. Since it was impossible to identify the real trip purpose of smart-card holders, another data set that contained observed trip purposes was utilized for validation. Fortunately, a conventional travel survey was conducted on a regular basis in the same area where the sample smart-card data was collected. The validation data, however, had a less accurate resolution of activity locations. Since the main purpose of the survey was to identify both the origin and destination of individual trips, the spatial unit was limited to a traffic analysis zone (TAZ) that is much wider than the catchment area of bus stops and metro stations adopted in the proposed model. Another limitation was that the survey was conducted for a single day, while the training sample of trip chains was taken on two consecutive dates. For the travel survey data, the duration of the initial and the last activities within a trip chain must be inferred under the assumption that travelers repeated the same trip chains every weekday.

These two handicaps entailed a discrepancy between the estimated and observed activity types later. Nonetheless, the travel data was the only option we could adopt. For validation, sample trip chains were taken from the survey data that contained only travel modes of public transportation. Table 6 shows the profile of travelers in the survey data across different travel modes. As expected, mix-mode users who utilized both car and transit were negligible. The validation sample was compatible with the training sample from smart-card data with respect to the distribution of the number of episodes within a trip chain, as shown in Table 7. As in the training sample, the validation sample also included only trip chains that started from and ended at home.

To estimate activities of data from the direct travel survey, a Viterbi algorithm was implemented again using the parameters estimated from smart-card data. The resultant activity sequences shown in Table 8 reveal a pattern that differed from the result of the training sample in Table 5. The trip chains that included flexible activities showed a larger proportion in the estimation result. This suggests that the validation sample was less accurate in both temporal and spatial resolutions.

Table 9 shows 4 activity types derived from smart-card data and 10 observed trip purposes in the travel survey. Among the trip purposes, “Work” represents a job requiring a commute, “Business” encompasses other works associated with a regular job, “Academy” represents a private education after school, and “Others” includes all other purposes that do not belong to any of the 9 predefined purposes. Since 4 activity types derived from smart-card data were not consistent with ten trip purposes employed by the travel survey, a direct comparison between the estimated and observed activities was not possible. Thus, the profile of observed activity sequences was examined for an activity sequence estimated using the Viterbi algorithm under the parameters derived from smart-card data. The relationship between two different classifications of activities will be discussed in the following.

For brevity, only profiles corresponding to the three most frequent activity sequences (i.e., “121”, “434”, and “1221”) were provided in Table 10. As expected, the observed work and school activities accounted for the relatively smaller portion (66.5%) out of the estimated activity of “flexible out-of-home activity” than that (92%) out of the estimated activity of “typical out-of-

**Table 6**
Distribution of trip chains chosen from a direct travel survey.

| | Car-only users | Transit-only users | Mix-mode users |
|---|---|---|---|
| Number of trip chains (%) | 34,038 (32.13) | 66,547 (63.83) | 4285 (4.03) |

**Table 7**
Comparison of samples with respect to the number of episodes within a trip chain.

| Number of episodes | 3 | 4 | 5 | 6 | Total |
|---|---|---|---|---|---|
| Sample from smart-card data | 1268 (88.92%) | 113 (7.92%) | 39 (2.73%) | 6 (0.42%) | 1426 (100%) |
| Sample from direct travel survey | 61,839 (92.93%) | 3099 (4.66%) | 1476 (2.22%) | 133 (0.19%) | 66,547 (100%) |

**Table 8**
Distribution of the most probable activity sequences in the validation sample.

| Activity sequence | Counts (%) | Activity sequence | Counts (%) | Activity sequence | Counts (%) |
|---|---|---|---|---|---|
| **“121”** | **37,607 (56.51%)** | “43224” | 44(0.07%) | “1224” | 6(0.009%) |
| **“434”** | **24,052 (36.14%)** | “122121” | 44(0.07%) | “43221” | 4(0.006%) |
| **“1221”** | **1777 (2.67%)** | “12212” | 43(0.06%) | “1222” | 4(0.006%) |
| “12121” | 1027 (1.54%) | “121221” | 33(0.05%) | “12122” | 3(0.005%) |
| “4321” | 530 (0.8%) | “12221” | 32(0.05%) | “4322” | 2(0.003%) |
| “4324” | 469 (0.7%) | “12124” | 32(0.05%) | “121224” | 2(0.003%) |
| “1212” | 293 (0.44%) | “432124” | 20(0.03%) | “432122” | 1(0.002%) |
| “43434” | 241(0.36%) | “4343” | 18(0.03%) | “12224” | 1(0.005%) |
| “122” | 180(0.27%) | “121212” | 18(0.03%) | | |
| “43212” | 49(0.07%) | “432121” | 15(0.02%) | | |

**Table 9**
Comparison of activity types.

| Proposed model | Travel survey |
|---|---|
| 1. Flexible home stay | 1. See someone off |
| 2. Flexible out-of-home activity | 2. Return home |
| 3. Typical work | 3. Work |
| 4. Typical home stay | 4. School |
| | 5. Academy |
| | 6. Business |
| | 7. Return to work |
| | 8. Shop |
| | 9. Recreation |
| | 10. Others |

**Table 10**
Profiles of observed activity sequences for the top three estimated activity sequences.

| (a) For estimated sequence “121” (flexible home stay → flexible out-of-home activity → flexible home stay) | |
|---|---|
| Observed activity sequence | Counts (%) |
| Home stay → work → home stay | 19,966 (53.09%) |
| Home stay → school → home stay | 5091 (13.54%) |
| Home stay → others → home stay | 4719 (12.55%) |
| Home stay → recreation → home stay | 2865 (7.62%) |
| Home stay → shopping → home stay | 2819 (7.50%) |
| Home stay → academy → home stay | 1663 (4.42%) |
| Home stay → business → home stay | 416 (1.11%) |
| (b) For estimated sequence “434” (typical home stay → typical out-of-home activity → typical home stay) | |
| Observed activity sequence | Counts (%) |
| Home stay → work → home stay | 18880 (78.5%) |
| Home stay → school → home stay | 3253 (13.52%) |
| Home stay → others → home stay | 758 (3.15%) |
| Home stay → academy → home stay | 597 (2.48%) |
| Home stay → recreation → home stay | 319 (1.33%) |
| (c) For estimated sequence “1221” (flexible home stay → flexible out-of-home activity → flexible out-of-home activity → flexible home stay) | |
| Observed activity sequence | Counts (%) |
| home stay → work → business → home stay | 279 (15.70%) |
| home stay → others → others → home stay | 242 (13.62%) |
| home stay → others → shopping → home stay | 105 (5.91%) |
| home stay → work → recreation → home stay | 84 (4.73%) |
| home stay → work → others → home stay | 84 (4.73%) |
| home stay → school → academy → home stay | 79 (4.45%) |
| home stay → recreation → recreation → home stay | 63 (3.55%) |
| home stay → school → recreation → home stay | 62 (3.49%) |
| home stay → school → others → home stay | 57 (3.21%) |
| home stay → work → shopping → home stay | 56 (3.15%) |
| home stay → recreation → others → home stay | 46 (2.59%) |
| home stay → recreation → shopping → home stay | 42 (2.36%) |
| home stay → shopping → shopping → home stay | 42 (2.36%) |
| home stay → business → business → home stay | 37 (2.08%) |

Observed sequences that took up less than 1% were omitted for (a) and (b).
Observed sequences that took up less than 2% were omitted for (c).

home activity”. The remaining proportion (33.5%) was taken by various observed activities such as recreation, shopping, academy, business, and other activities. The observed work activity was more likely to be clustered to the estimated flexible out-of-home activity, if either its start time did not correspond to the morning peak hours or the venue where it took place did not belong to locations of the typical office or commercial land-uses. On the other hand, observed activity sequences “home-to-work-to-home” and “home-to-school-to-home” covered more than 92% of the estimated activity sequence of “typical home stay → typical out-of-home activity → typical home stay”. As a result, estimated activity sequences, the length of which was 3, turned out to be well validated with the observed activities. Table 10 (c) shows the validation result for a longer activity sequence, the length of which was 4. Observed activity sequences for the estimated sequence of “flexible home stay → flexible out-of-home activity → flexible out-of-home activity → flexible home stay” were distributed across a wide range. The most typical patterns encompassed various non-work activities after completing work or school, which covered business, recreation, shopping, academy, and other activities. The remaining estimated activity sequences were also found to be intuitively accountable, but details of them were omitted for brevity. It should be noted that the validation performance could be enhanced if the observed data were collected in a more precise manner and more plausible features were added in the model. The present study did not offer the completed version of the imputation model, but introduced a novel possibility to accurately annotate various human mobility data.

## 5. Conclusions

The current stream of machine-learning study concentrates more than ever on the advantage of an unsupervised model. At the stage when massive amounts of transportation-related data are accessible, it is high time for transportation researchers to turn their eyes to unsupervised machine learning tools and to excavate precious values hidden inside the data itself. The proposed model to infer latent activities behind smart-card data could be regarded as such an effort.

Despite the limitation that multi-day smart-card data were unavailable, very promising results were yielded from the proposed model based only on smart-card data collected during only two consecutive days. Feature vectors were properly clustered into eight categories, each of which was both intuitively accountable and sufficiently specific. The estimated membership probabilities characterized four distinctive activities in a straightforward manner without confusion. The estimated transition probabilities also accounted well for the relationship between the derived activities. Finally, the proposed model provided an efficient way to estimate the most probable activity sequence for a trip chain, given that the observed features were known. This could contribute to developing a more practical activity-based transportation demand analysis model.

With few reservations, we recommend that the typical activity types most researchers have adopted thus far should be reconsidered from a different point of view. All previous studies have confined human activities to several predefined types. Self-clustered activities could offer novel insight into understanding human mobility, even though these have neither a clear description nor an exact title. In other words, human activities could be clustered and interpreted without any fixed conceptual framework.

If the provision that the number of hidden states and clusters should be arbitrarily chosen could be overcome, this would be a great improvement for the proposed model. Recently, Lee and Sohn (2015) shed light on the possibility that parameters changing the model specifications could be determined in a rigorous manner. They used a reversible jump Markov chain Monte Carlo simulation to determine the most probable number of used routes when recognizing transit-route use patterns within a Bayesian framework. Even though the present model is much more complex than their model, the methodology could be applied to the present model without the necessity of large-scale changes to the frame. The fact that the proposed model cannot accommodate human interactions is another handicap. We continue the search for a way to include them into the model, and expect the next version will resolve the problem.

## Acknowledgments

This research was supported by the Chung-Ang University Excellent Student Scholarship, and also supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean government (MSIP) (2015005843).

## References

Allahviranloo, M., Recker, W., 2013. Modeling uncertainty in households' activity engagement decisions. In: The Proceedings of 92nd Annual Meeting of Transportation Research Board of the National Academies. Washington, DC, (No. 13-2876).

Allahviranloo, M., Recker, W., 2013. Daily activity pattern recognition by using support vector machines with multiple classes. Transportation Research Part B 58, 16–43.

Arentze, T., Timmermans, H., 2000. Albatross: A Learning Based Transportation Oriented Simulation System. Eirass, Eindhoven.

Bishop, C., 2006. Pattern Recognition and Machine Learning. Springer Science + Business Media, Singapore.

Bohte, W., Maat, K., 2009. Deriving and validating trip purposes and travel modes for multi-day GPS-based travel surveys: a large-scale application in the Netherlands. Transportation Research Part C: Emerging Technologies 17 (3), 285–297.

Bowman, J.L., Ben-Akiva, M.E., 2001. Activity-based disaggregate travel demand model system with activity schedules. Transportation Research Part A 35 (1), 1–28.

Cappe, O., Moulines, E, Ryden, T., 2005. Inference in Hidden Markov Models. Springer Series in Statistics, Springer.

Chow, J.Y., 2014. Activity-based travel scenario analysis with routing problem reoptimization. Computer-Aided Civil and Infrastructure Engineering 29 (2), 91–106.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39 (1), 1–38.

Elango, V., Guensler, R., 2010. An automated activity identification method for passively collected GPS data. In: Paper Presented at the 3rd Conference on Innovations in Travel Modelling. Phoenix, AZ.

Feng, T., Timmermans, H. J. P. (2014). Multi-week travel surveys using GPS devices: experiences in The Netherlands. Mobile Technologies for Activity–Travel Data Collection & Analysis, ed. S. Rasouli, H.J.P. Timmermans, 104-118.

Giaimo, G., Anderson, R., Wargelin, L., Stopher, P., 2010. Will it work? Pilot results from the first large-scale GPS-based household travel survey in the United States. Transportation Research Record (2176) 26–34.

Greaves, S., Fifer, S., Ellison, R., Germanos, G., 2010. Development of a global positioning system web-based prompted recall solution for longitudinal travel surveys. Transportation Research Record (2183) 69–77.

Goulias, K.G., 1999. Longitudinal analysis of activity and travel pattern dynamics using generalized mixed Markov latent class models. Transportation Research Part B 33 (8), 535–558.

Itsubo, S., Hato, E., 2006, January. A study of the effectiveness of a household travel survey using GPS-equipped cell phones and a WEB diary through a comparative study with a paper based travel survey. In: Presented in Transportation Research Board (TRB) 85th annual meeting. Washington, DC.

Kelly, P., Krenn, P., Titze, S., Stopher, P., Foster, C., 2013. Quantifying the difference between self-reported and global positioning systems-measured journey durations: a systematic review. Transport Reviews: A Transnational Transdisciplinary Journal 33 (4), 443–459.

Kohla, B., Meschik, M., 2013. Comparing trip diaries with GPS tracking: an Austrian study. In: Zmud, J., Lee-Gosselin, M. (Eds.), Transport Survey Methods: Best Practice for Decision Making. Emerald Group Publishing Limited, Bingley, pp. 306–320.

Kroesen, M., 2014. Modeling the behavioral determinants of travel behavior: an application of latent transition analysis. Transportation Research Part A 65, 56–67.

Lee, M., Sohn, K., 2015. Inferring the route-use patterns of metro passengers based only on travel-time data within a Bayesian framework using a reversible-jump Markov chain Monte Carlo (MCMC) simulation. Transportation Research Part B: Methodological 81, 1–17.

Leszczyc, P.T.L.P., Timmermans, H., 2002. Unconditional and conditional competing risk models of activity duration and activity sequencing decisions: an empirical comparison. Journal of Geographical Systems 4 (2), 157–170.

Liu, F., Janssens, D., Wets, G., Cools, M., 2013. Annotating mobile phone location data with activity purposes using machine learning algorithms. Expert Systems with Applications 40 (8), 3299–3311.

Liu, F., Janssens, D., Cui, J.X., Wang, Y.P., Wets, G., Cools, M., 2014. Building a validation measure for activity-based transportation models based on mobile phone data. Expert Systems with Applications 41 (14), 6174–6189.

Lu, Y., Zhu, S., Zhang, L., 2013. Imputing trip purpose based on GPS travel survey data and machine learning methods. In: Transportation Research Board 92nd Annual Meeting (No. 13-3177).

Moiseeva, A., Jessurun, J., Timmermans, H., 2010. Semiautomatic imputation of activity travel diaries. Transportation Research Record: Journal of the Transportation Research Board 2183, 60–68.

Movellan, J.R. (2003) Tutorial on Hidden Markov Models, http://mplab.ucsd.edu/tutorials/hmm.pdf

Park, B.J., Zhang, Y., Lord, D., 2010. Bayesian mixture modeling approach to account for heterogeneity in speed data. Transportation Research Part B: Methodological 44 (5), 662–673.

Rank, E., Pernkopf, F., 2004. Hidden Markov models. Lecture Notes in Speech Communication 2, Signal Processing and Speech Communication Laborator. Graz University of Technology, Graz, Austria.

Rasmussen, T., Ingvardson, J.B., Halldórsdóttir, K., Nielsen, O.A. (2013). Using wearable GPS devices in travel surveys: a case study in the Greater Copenhagen Area. Transport Conference at Aalborg University. ISSN 1603–9696. Retrieved from http://www.trafikdage.dk/papers_2013/188_ThomasKjaerRasmussen.pdf

Reumers, S., Liu, F., Janssens, D., Cools, M., Wets, G., 2013. Semantic annotation of GPS traces: activity type inference. In: 92nd Annual Meeting of the Transportation Research Board. Transportation Research Board of the National Academies.

Richardson, S., Green, P.J., 1997. On Bayesian analysis of mixtures with an unknown number of components (with discussion). Journal of the Royal Statistical Society Series B 59 (4), 731–792.

Serfozo, R.F., 1979. An equivalence between continuous and discrete time Markov decision processes. Operations Research 27 (3), 616–620.

Stopher, P., Zhang, J., FitzGerald, C., 2007. Deducing mode and purpose from GPS data. In: Presented at 11th National Transportation Planning Applications Conference of the Transportation Research Board. Daytona Beach, Fla. May 2007. 14.

Stopher, P., FitzGerald, C., Zhang, J., 2008. Search for a global positioning system device to measure person travel. Transportation Research Part C: Emerging Technologies 16 (3), 350–369.

Viterbi, A.J., 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory 13 (2), 260–269.

Wolf, J., Guensler, S., Bachman, W., 2001. Elimination of the travel diary: an experiment to derive trip purpose from GPS travel data. Transportation Research Record 1768, 125–134 Paper No. 01-3255.

Yang, Y., Enjian, Y.A.O., Hao, Y.U.E., Yuhuan, L.I.U., 2010. Trip Chain's activity type recognition based on support vector machine. Journal of Transportation Systems Engineering and Information Technology 10 (6), 70–75.

Yang, M., Yang, Y., Wang, W., Ding, H., Chen, J., 2014. Multiagent-based simulation of temporal-spatial characteristics of activity-travel patterns using interactive reinforcement learning. Mathematical Problems in Engineering 2014.

Zraiaa, M., 2010. Hidden Markov Models: A Continuous-Time Version of the Baum–Welch Algorithm. Dissertation for the MSc Degree in Advanced Computing of Imperial College, London, UK.