Understanding Occupancy Patterns of Stay Locations using Semantic Trajectory Analytics

Muhammad Arslan
Univ. Bourgogne Franche-Comte, Dijon, France
muhammad.arslan@u-bourgogne.fr

Christophe Cruz
Univ. Bourgogne Franche-Comte, Dijon, France
christophe.cruz@u-bourgogne.fr

Ana-Maria Roxin
Univ. Bourgogne Franche-Comte, Dijon, France
ana-maria.roxin@u-bourgogne.fr

Dominique Ginhac
Univ. Bourgogne Franche-Comte, Dijon, France
dom@le2i.cnrs.fr

Abstract

The analysis of dynamic interactions of objects has always received great attention due to widespread applications of positioning technologies such as the Global Positioning System (GPS). However, in order to completely understand meanings behind objects` mobility related interactions, spatio-temporal GPS trajectories need enrichment with semantic data sources including application domain knowledge and geographic databases. For this, a prototype system is proposed that is built using a data model that has an ability to hold information of moving and changing objects in the context of dynamic built environment. After pre-processing and semantic enrichment processes of GPS trajectories, stay locations of users have been identified along with their frequency of visiting to understand the criticality of such locations in daily use. Stay locations of a building are considered for the analysis than moving locations because stay behavior shows that something interesting is happened over there. After extracting the stay locations and annotating them with corresponding semantic information, Hidden Markov Model (HMM) along with the Viterbi algorithm is used to understand the occupancy patterns of these stay locations. Visualization of occupancy patterns of stay locations has a potential to help facility managers in making improved decision making for monitoring and controlling building activities and ultimately can help in safety management of buildings.

KEYWORDS

Trajectories, stay locations, GPS, occupancy

1 INTRODUCTION

Technological developments towards ubiquitous information and computing has led to the generation of huge amount of real-time sensor data on a daily basis from different urban sources [1]. Smart cities built on using such technological infrastructure offer novel numerous applications that can support decision making in multiple situations [2]. For example, time series trajectory data [3] acquired from smart phones equipped with positional technologies such as GPS can be useful to understand the movement dynamics of moving objects. Knowledge extracted from such data can help us to find existing relevant patterns in order to understand the complex and unpredictable behavior of objects [1, 2]. A raw time series trajectory data is a collection of ordered sequences of spatio-temporal triples in the form of $\langle latitude, longitude, timestamp \rangle$ extracted at equally spaced time intervals from the movement data of an object moving in a geographical space as shown in Figure 1 [4]. A trajectory can be of a person, vehicle or an animal carrying a device that generates location data [5]. In general, mobility data of moving objects is continuous in nature however, trajectories are discrete traces collected at some sampling periods [6]. Therefore, a trajectory can be considered as an approximation of the real mobility data.

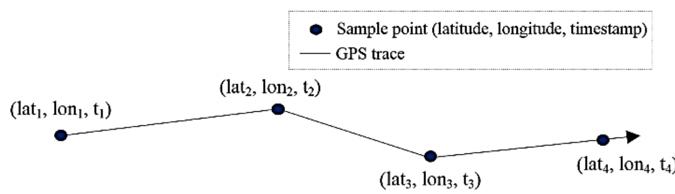

Figure 1 **A sampled time series trajectory**

There exists numerous technologies that provides location data such as Radio Frequency Identification (RFID), location estimation using 802.11, GSM beacons, GPS and so on [7]. The most common and widely used method to collect location data is by using GPS-equipped devices [8]. The raw trajectories acquired from location acquisition devices are well-suited for applications which aim to find the location of a moving object (e.g., where was John at 9pm on Dec 01, 2017?) or to calculate statistics such as distance, direction, acceleration, etc. (e.g., John was driving at what speed to which direction?) [9]. However, additional information from the application context can be used to make raw trajectories more meaningful for supporting advanced trajectory applications, for example traffic monitoring application [4, 5]. City map information, details of on-going events in the city and weather forecast can be used with the trajectory data to distinguish different traffic conditions. This additional information is called as annotation that can be collected using various external data sources such as web pages, application and geo databases [5]. The annotation mainly involves mapping the trajectory to places of interest, road networks for a traffic monitoring application or geographical regions. The process of adding annotations to a trajectory as a whole or its subpart is known as semantic enrichment [4,5].

Once a trajectory data is semantically annotated, it offers remarkable opportunities to predict the human movements by understanding their mobility patterns. However, in order to make efficient use of a trajectory data in a comprehensive way, several pre-processing techniques [10] and annotation methods [5] should be applied to transform raw trajectory data into information enriched processed trajectories for real-time applications [11]. To understand the process of trajectory preprocessing and semantic enrichment, a brief survey is presented in the next section.

The rest of the paper is organized as follows: Section 2 introduces the background literature along with research methodology used for this work. Section 3 is based on the discussion on the STriDE model along with semantic enrichment and extraction of users' patterns using Hidden Markov Model. Section 4 presents the limitations of the presented work and a conclusion.

2 BACKGROUND

A literature review was initially performed to methodically collect information for identifying and understanding the problem domain, which is to extract occupancy patterns of building users from their semantically enriched trajectories after finding their stay locations. At first, a review of literature on smart buildings reveals that a trajectory data for understanding fine-grained user-building interactions is only useful if data related to changing and moving of building objects is incorporated [1, 2]. Figure 2 shows the research methodology framework adopted for this research. In order to address this dynamic building environment scenario, a data model named STriDE (Semantic TRajectories In Dynamic Environments) was initially constructed after identifying the relevant data structures which are required to store such dynamic information to understand the users activities followed by the industry feedback [12]. Using STriDE model, a prototype system is developed that captures raw spatio-temporal GPS data of moving objects, transforming it into trajectories after data cleaning and segmentation processes and then enriching it with semantic information captured from geographical data sources and contextual repositories. However, extraction of behavior information of user-building interactions from semantically enriched stored GPS trajectories is still missing.

In order to address this gap, a study based on statistically learning is proposed by using probabilistic framework HMM to understand users' behavior in a building using semantic trajectories. The transitions of users within a building across different stay locations are employed to model user occupancy behavior for prediction. In addition, after computing transition probabilities of users to move between different stay locations using Baum-Welch algorithm, most probable occurring patterns are also extracted using a Viterbi algorithm based on the output of learned parameters. The extracted patterns will be useful for the facility managers to visualize the occupancy behavior of users which can lead in improved space management intervention strategies and will also help in identifying high-risk user behaviors to prevent safety hazards.

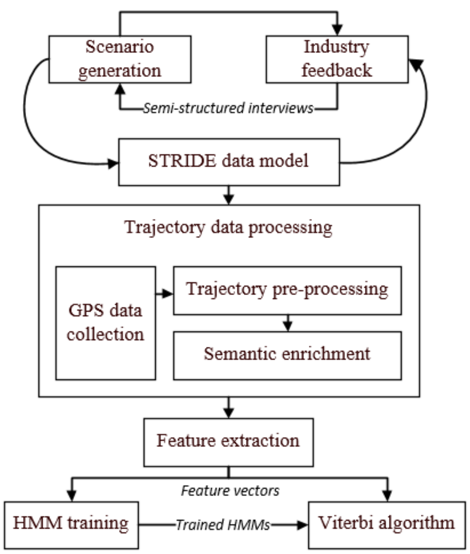

Figure 2 **Research methodology**

2.1 Trajectory modeling

Trajectory modeling in the first step to construct semantically enriched trajectories. It is a process to define and analyze data requirements to support trajectories` applications. In the literature [10, 13], there exist four main classes of data models that can be used to model trajectories before saving them to the actual database, which are mentioned below;

a) Data type-based

Zheni et al. presented a model to represent trajectories as an abstract data type (ADT) [14]. The model integrates spatial, temporal and thematic dimensions for representing and manipulating trajectory data. A language to manipulate trajectories is also introduced to query data using spatial, temporal and set-based operations. The major limitation of abstract data type-based approach is the dependency of the trajectory data type on the application as model represents trajectories as a series of connected trips and activities. However, ADT model offers a sufficient support to extract additional information on the evolution of the represented trajectories.

b) Design pattern-based

Data type based models alone are not enough for constructing trajectory applications because it imposes the use of a generic data type to represent trajectories for all the applications [13]. In order to address this drawback, Parent et al. designed pattern-based model [15], that is based on Model Analysis and Decision Support (MADS). It supports spatial and temporal objects and relationships by providing a way to describe the spatial extent and life span of the trajectory. It represent trajectories as a series of 'stops' and 'moves' segments having 'begin' and 'end' timestamps. However, in order to define trajectory components that are 'stops' and 'moves', there is a need to manually input the contextual information into the model according to the application [13].

c) Ontology-based

Ontology is the conceptualization of a specific domain for showing relationships between concepts in the form of a hierarchy [13]. An example of a model based on ontologies is presented by Yan [16]. It offers a multi-layered model to represent trajectories. At first, the raw GPS data is cleaned and raw trajectories are constructed. Then these trajectories are transformed into structured trajectories using segmentation approaches. After extracting meaningful segments, these trajectories are mapped on ontologies to add semantic information. In addition, Noël et al. presented ontological design patterns to model trajectories for understanding trajectory events in a better way [17]. Then, ontologies are constructed by the exploitation of proposed patterns. Finally, SPARQL language is used to query the model to retrieve spatial, temporal and thematic dimensions of a trajectory data. If ontology based modeling is compared with formally discussed two modeling approaches, they represent richer semantic information by integrating different types of information enrichment processes.

d) Hybrid-based

Existing literature also presents solutions based on hybrid modeling approach in order to combine best features of different models discussed above for constructing trajectories [13]. The hybrid model presented by Yan et al. [5] offers different levels of data abstraction by encapsulating geometry and semantics of trajectories. Within the proposed model, there are three different sub models which are: (1) raw trajectory model, to clean trajectories, (2) conceptual mode, to transform raw trajectories into a series of episodes by calculating velocity, acceleration, direction, density and time interval parameters, (3) semantic model, to enrich processed trajectories with the third party information. The important aspect of using the hybrid approach is to cover requirements of a variety of application by offering three different types of data abstraction that are raw trajectories, trajectory episodes and semantically enriched trajectories.

2.2 Trajectory pre-processing

Once location data of moving objects is acquired and data requirements for modeling trajectory database is done, raw trajectories should be pre-processed according to the application requirements [18]. The basic idea of pre-processing trajectories is to reduce the storage, processing and communication overheads without compromising in precision of a trajectory data [18]. Generally, there exists two modes to pre-process a trajectory; (a) online mode, where trajectories pre-processing algorithms are executed in real time, and (b) offline mode, where all the trajectories` processes are done in offline mode [4, 18]. The basic tasks of pre-processing includes noise filtering, reduction, segmentation and stay point detection [18]. The objective of noise filtering is to remove noise from trajectories that can be caused by weak signals of location acquisition systems. Trajectory reduction processes are meant to reduce the size of the trajectory to minimize the computation overhead, while keeping the usefulness of the trajectory [4]. Segmentation deals with the algorithms to divide trajectories into segments. The criteria of segmentation can be based on the time interval, spatial property or semantic meaning [19]. Stay points detection techniques identifies the location points where the moving object has spent some time by staying over there within a specific distance. A stay point can be a shopping mall, restaurant or an office.

2.3 Trajectory semantic enrichment

Once pre-processing of trajectories is completed, process of annotation is applied to transform GPS trajectories into semantic trajectories. In this process, trajectory`s segments having specific intervals are assigned meaningful information regarding their mobility tracks [5]. As trajectories are already divided into stops and moves episodes during the segmentation process, now it's important to mark stops because stopping in a location means that something interesting is happened. Annotation is a process to add application context data to raw spatio-temporal trajectories [4, 5]. A conventional semantic enrichment process for trajectories receives a collection of raw trajectories as an input and produces annotated trajectories, which are known as semantic trajectories [18]. The annotation process involves mapping episodes to the meaningful information such as; the mapping of places of interests that can be in the form of Points of Interest (POI), roads in the form of lines or the geographical regions as discussed below [5].

a) Annotation with semantic regions

Annotating trajectories with meaningful geographic regions requires computation of topological correlations [5]. These correlations should be made with third party data sources that contains spatial regions having semantic places. Spatial regions can be of two types: First one is the free-style spatial region having irregular shapes having no restrictions on the shape, e.g. circle, triangle, polygon, etc. that can corresponds to real-life geographic regions [5]. Second type is using the well divided grid areas, e.g. land use data containing $50m \times 50m$ cells. In order to use free-style method, data of regions can be extracted from Openstreetmap (OSM) file that offers editing in the world map [5]. Once data is extracted, it can be stored in a trajectory database for further annotation. To annotate a trajectory episode with a region, spatial join can be used i.e. $Trajectory\ epidode \bowtie_{\theta} Region$. However, parameter 'θ' is very crucial to compute. Calculation of parameter 'θ' can be based on a single or the combination of multiple topological spatial relations such as distance, displacement, etc. Once 'θ' is computed, spatial join can be performed either with the boundary of the trajectory episode or with its center. In this way, region with its associated metadata can be annotated with episodes of a trajectory.

b) Annotation with semantic lines

It will annotate trajectories with semantic lines by inferring transportation modes of moving objects such as walking, running, cycling and public transport for example tram, bus, etc. [5]. These transportation modes are determined based on the characteristics of the mobility along considering the information related to landmarks/POIs where the trajectory`s episode has passed. Transportation modes can be determined directly by calculating average speed of a moving object. As a walk speed of a person is usually less than 1.4 meter per second (m/s), whereas a person moving with a speed greater than 1.4 m/s will be annotated as a 'run' segment.

c) Annotation with semantic points

When there are stop episodes in a trajectory, it means that something interesting has happened during stops. Stopping on any location with a certain time duration shows the existence of the POI [4, 5, 18]. It is important to know that which type of POI motivated the moving object to stop. POI can be the shopping centers, restaurants, offices, movie theater, post office, etc. The more densely populated the landscapes, it will be more difficult to identify the POIs because there will be many candidate POIs for making a stop.

2.4 Semantic trajectory analytics

After enriching trajectories with semantic information, now it's time to perform some data analytics for patterns extraction. A HMM is a statistical tool to represent probability distributions (PDs) over sequences of observations and widely used for predictive analytics [20]. In statistics, a PD is used to provide probabilities of occurrence of different possible outcomes in an experiment. For example, if the random variable X is used to denote the outcome of a coin toss ("the experiment"), then the PD of X would take the value 0.5 for $X = heads$ and 0.5 for $X = tails$ with an assumption that the coin is fair. In short, the PD in terms of an underlying sample space that is collection of all possible outcomes of the random process being under observation [21]. The sample space may contains real numbers (univariate PD) or a vector space (multivariate PD) or it may be a collection of non-numerical values; for instance, the sample space of a coin flip experiment would be $\{heads, tails\}$. PDs are generally categorized into two main types, which are (1) discrete PD (having collection of possible outcomes in a discrete form, for example a discrete list of probabilities of the outcomes in a coin toss experiment can be defined using probability mass function), (2) a continuous PD (having the collection of possible outcomes that take values in a continuous range, for instance, humidity of a day is defined by probability density function). It was a short introduction to PDs [22]. Now, coming back to HMM. Let us denote the observation at time t by the variable A_t. This variable can have discrete PD or a continuous PD over time. We will make an assumption that the observations are sampled in equally-spaced discrete time intervals so that value of t can be an integer value. The HMM has three main properties that defines it [23, 24]. First, it makes assumption that the observation was generated by a process at time t whose state S_t is hidden from the observer. Second, it assumes that the state of this hidden process fulfills the Markov property. That is, the current state S_t of the process is only dependent on the only previous state S_{t-1} and independent of all the states prior to $t-1$. It means that, in order to predict the future of process, the hidden state encapsulates all the information we need to know from the process history. A third supposition of the HMM is that the hidden state variable can only take K integer values $\{1, 2, 3, \dots, K\}$. In general, a HMM is described by a set of three parameters which can be written as the 3-tuple $\lambda = (A, B, \pi)$. The description of the HMM parameters are mentioned in Table 1.

There are three basic scenarios for which HMM can be useful for real world application [23, 24]. These scenarios are the following;

1. Computing the probability of the observation sequence $P(O|\lambda)$ using the given HMM (λ) and the observation sequence $O = \{O_1, O_2, O_3, \dots, O_T\}$.
2. Extracting the most optimal hidden state sequence $(Q = q_1, q_2, q_3 \dots q_T)$ which best explains the observations using the given HMM (λ) and the observation sequence $O = \{O_1, O_2, O_3, \dots, O_T\}$.
3. Adjusting the values of the state transition probabilities (A) and the output emission probabilities (B) of the HMM (λ) to maximize the probability of the observation sequence $O = \{O_1, O_2, O_3, \dots, O_T\}$. This process is called training as the observation sequence is used to train the HMM to make it best for observing real phenomena.

However, this work will focus on the latter two scenarios.

Table 1 **A Hidden Markov Model Parameters**

$O = \{o_1, o_2, o_3, \dots, o_T\}$	The observation sequence, T is the number of observations in a sequence.
$Q = q_1, q_2, q_3 \dots q_T$	A set of T hidden states; Number of states does not depend on the number of observations.
$\pi = \pi_1, \pi_2, \pi_3, \dots, \pi_N$	$\pi = \{\pi_i\}$ is the probability that the model is in the state i at time $t = 0$. Some states j may have $\pi_j = 0$, it means that they cannot be initial states. Sum of all the initial probabilities should be equal to 1. $\sum_{i=1}^{n} \pi_i = 1$
$A = a_{11}, a_{12}, a_{13}, \dots, a_{nn}$	A transition probability matrix, each a_{ij} representing the probability of moving from state i to state j. Sum of transition probabilities should be equal to 1. $\sum_{j=1}^{n} a_{ij} = 1$
$B = b_i(o_t)$	An emission probability matrix also known as observation likelihoods, each $b_i(o_t)$ representing the probability of an observation o_t being generated by a state i.

3 A PROTOTYPE SYSTEM FOR UNDERSTANDING BUILDING OCCUPANCY PATTERNS

After reviewing the literature regarding all the steps involved in transforming the raw trajectories to semantic trajectories. A prototype system is proposed which uses pre-existing data model STriDE, performs pre-processing and semantic enrichment processes. Finally, occupancy patterns of stay locations of a building are extracted using HMM.

3.1 The STriDE model for dynamic building environment

The idea of dynamic environment contains both natural and artificial objects built by humans for example cities, infrastructure, or public spaces [25]. An environment is called as dynamic when the objects have the ability to move and can even change its shape, size, change its attributes [25]. To address the challenges of objects` properties changing over time from a dynamic built environment, a hybrid data model (already discussed in Section 2.1) for Semantic Trajectories in Dynamic Environments named STriDE was proposed. This model obtains tracking results of changing entities by combining raw data pre-processing and semantic definition of the environment. Based on the Continuum model [26-29], the model combined 2D/3D semantic environments and semantic trajectories, an indexing model for time-dependent large-scale graph with semantic resources. The Continuum model was defined to capture moving and changing objects. The STriDE derived from the Continuum model did the same but has an ability to capture at the same time moving and changing entities of the building environment. More details on the STriDE model can be found in [12].

3.2 Trajectory pre-processing and semantic annotation

To understand the interactions of users in the dynamic environment, use of GPS trajectory data has been considered. A single GPS data log consists of a user identification (ID), timestamp, floor, latitude and longitude value. After collecting GPS data from users` handheld devices, it will undergo the preprocessing steps to transform it into trajectories. The first step is to use an appropriate filter to remove outliers from the data. There exists many outliers removing filters to improve the data quality. However, median filter is used on an acquired GPS data (as shown is Figure. 3) because it depicts robustness property in filtering [4]. After removing the outliers, stay points of users are calculated to enrich a trajectory with semantic points in the form of stop and move segments. Stay points are the location points where a user has spent a considerable amount of time within a specified distance. By adjusting the distance (D_{thresh}) and time threshold (T_{thresh}) values, stay points in a trajectory have been identified along with its frequency of visiting the same location (as shown in Figure. 4) using Zheng et al. approach [4]. The factor of frequency is taken into account to understand which stay locations are visited more often by the users.

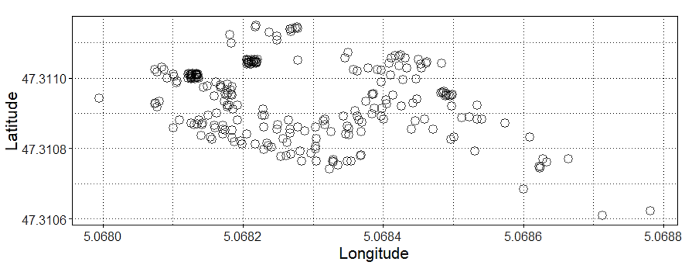

Figure 3 **Trajectory data of a user**

The main purpose of calculating stay points in a user trajectory data is to find locations in a building where users are spending more time than required. This information will help to track the occurrence of an unexpected situation in a building if a stay duration is greater or less than the required. Once stay points are identified, there is a need to label each stay point with a building identification (ID) that corresponds to different building locations such as room215, room215, room217 and 219. To tag building identification (ID), we need to map trajectory points with a geographic region and it can be done by using third party data sources. To do this, we have used information extracted from an OSM file to tag trajectory points with spatial regions using free-style mapping technique. After mapping the spatio-temporal locations in a trajectory to building identifications, a trajectory is enriched with semantic points (room ids), semantic lines in the form of movement types which are being carried by users (speed values calculated between two segments in m/s) and a semantic region (building structure) as show in Figure 5.

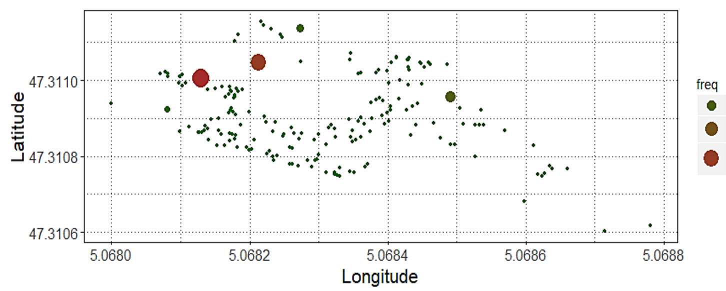

Figure 4 **Stay points detection and their frequency of visiting**

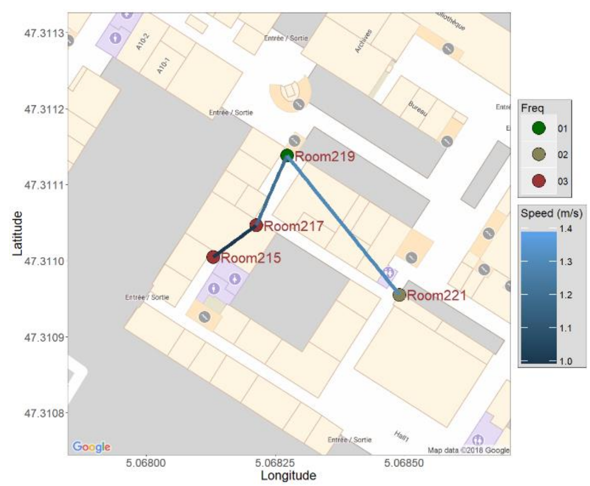

Figure 5 **Visualizing a user` trajectory with semantic stay points in three different colors (room identifications for e.g. Room215), semantic lines (based on speed values calculation) and a semantic region (building structure)**

3.3 Extracting occupancy patterns from semantic trajectories

In order to extract occupancy patterns from semantically enriched trajectories (as discussed in Section 3.2), features are extracted from the processed trajectory data. Then, HMM is trained for each stay location and its corresponding transition and emission matrices are computed. Finally, the occupancy patterns of stay locations are visualized using R studio.

3.3.1 Feature extraction for semantic points

For the example: there are four locations in the building that are identified as stay locations which are 'room 215', 'room 217', 'room219' and room 221. After tagging semantic points to trajectory stay locations, we want to extract occupancy behavior of occupants within these stay locations. Stay locations of a building occupant trajectory are considered for the analysis than moving locations because stay behavior shows that something interesting is happened over there. These stay locations have already been identified using a stay point detection algorithm during the pre-processing of GPS trajectories. Processed trajectory data shows that occupant stayed in some areas in the building for a longer duration than the other areas. As spatial information (latitude and longitude coordinates) of stay location is already transformed into corresponding semantic points. These semantic stay points are made distinct in a way that no location should be repeated consecutively.

For each stay point, an independent HMM is considered. In addition, to use HMM method we have to qualify some measurement variables as observables. The process of transforming input data into set of features using measurement variables is known as feature extraction [30]. The benefit of using features is that it allow to perform the required analysis using reduced representation of data instead of the full size input [30]. This task is achieved using stay point detection algorithm in trajectories pre-processing. [19] The stay point algorithm grouped the data points that were giving same location information within a defined time threshold. For our analysis, feature value is a real number, which is calculated after pre-processing the sensor data values. For occupancy recognition of the users in any of the stay location, there are two possible values; the high-occupied (observable value 1) and the low-occupied (observable value 0) and modeling each observable value as a hidden state. For our case, the number of hidden states will be 2 as shown in Table 2 as we need to model the transitions in the stay point in term of two levels that are "Occupancy-Low" and "Occupancy-High". In addition, all the considered trajectories for training the HMM have equal number of data points in order to keep the length of trajectories same. For understanding the occupancy behavior of users in the stay locations, we have first calculated the transition and emission matrices then constructed a HMM to extract occupancy patterns as discussed in the below section.

Table 2 **Criteria for the assignment of observation values to identify hidden states**

Occupancy level	**Users threshold**	**Observation value**
Occupancy-Low	1-3 users inside the room	0
Occupancy-High	greater than 3 users inside the room	1

3.3.2 Computing transition and emission matrices

First thing we need to know the transition probabilities of users within the stay locations during different time interval of a day. For this, the most likely set of state transitions and output probabilities are need to be calculated. Transitions between the same or different hidden behavioral states (as discussed in Table 2) can be predicted from the state transition matrix (A) and emission matrix also known as the state dependent observation matrix (B). Given a sequence of hidden states $q_1, q_2\ and\ q_3$, the probability of occurrence of a future state q_{n+1} depends only on the current state q_n. These probabilities are grouped together to form a transition matrix holding transition probabilities from one state to another state and vice versa.

The likelihood of a trajectory point to be generated is estimated and estimations are formed as the emission matrix. The model is initially trained to compute transition and emission matrixes. The HMM model at first check the emission matrix to search the state that best describes the observable under consideration. Then, the most probable future state is determined using the transition matrix. Once the HMM is completely trained, the resulting transition and emission matrices are used for predicting the occupancy of stay locations. As an example, a single stay location (room 212) has been used to show the process of calculating transition and emission probabilities as shown in Table 3, 4 and 5. New trajectory data collected from users can be used as an input to the model to predict the immediate future occupancy state of a location. For training the model, Baum-welch algorithm [24] is used to estimate the model parameters A and B to find which HMM best explains the observation sequence and maximizes the probability of the observation sequence. Computation of the parameters of a posteriori HMM is done using below mentioned equations;

$$Expected\ number\ of\ times\ of\ visiting\ state\ i\ at\ time\ (t=1) = \bar{\pi}_i = P(q_1 = i|O, \acute{\lambda})$$

$$\boldsymbol{Transistion\ matrix}\ (\bar{\boldsymbol{A}}_{\boldsymbol{ij}})$$

$$= \frac{Expected\ number\ of\ transitions\ from\ state\ i\ to\ state\ j}{Expected\ number\ of\ transitions\ from\ state\ i}$$

$$= \frac{\sum_{t=2}^{T} P(q_{t-1} = i, q_t = j|O, \acute{\lambda})}{\sum_{t=2}^{T} P(q_{t-1} = i|O, \acute{\lambda})}$$

$$\boldsymbol{Emission\ matrix}\ (\bar{\boldsymbol{B}}_{\boldsymbol{i}}(\boldsymbol{z_k}))$$

$$= \frac{\text{Expected number of times in state i and observing symbol } z_k}{Expected\ number\ of\ times\ in\ state\ i}$$

$$= \frac{\sum_{t=1, o_t = z_k}^{T} P(q_t = i|O, \acute{\lambda})}{\sum_{t=1}^{T} P(q_t = i|O, \acute{\lambda})}$$

Table 3 **HMM for a room 212 during 9 am to 12 pm.**

State transitions (A)			
		$q_{t+1} = 1$	$q_{t+1} = 2$
		Low-occupancy	High-occupancy
$q_t = 0$	Low-occupancy	0.402	0.598
$q_t = 1$	High-occupancy	0.336	0.664
Probability of observation (B)			
		$O_t = 0$	$O_t = 1$
$q_t = 0$	Low-occupancy	0.269	0.731
$q_t = 1$	High-occupancy	0.240	0.760
Initial probability of each hidden state (π)			
		0.8	0.2

3.3.3 Computing most probable state sequences

After training the HMM, the next objective is to compute the most probable sequence of states Q to have generated an observation O that is, finding the hidden state sequence $Q = \{q_1, q_2, q_3 \dots q_T\}$ that maximizes probability $P(O, Q|\lambda)$ for a given observation. In our case, as discussed above occupancy levels are the hidden states and we want to find the most probable sequence of these location that a user follows inside the building. For this, we have used Viterbi algorithm [23, 24], a form of dynamic programming to extract most probable sequence of states for a given observation. For each symbol o_T in the observation O, the algorithm calculates the probability of its emission for each possible hidden state. The algorithm starts by computing the initial probability of the emission of the symbol o_1 in all possible states. Then, it computes again the emission of the symbol o_2 for each state transition. This process is repeated

Table 4 **HMM for a room 212 during 12 pm to 2 pm.**

State transitions (A)			
		$q_{t+1} = 1$	$q_{t+1} = 2$
		Low-occupancy	High-occupancy
$q_t = 0$	Low-occupancy	0.381	0.619
$q_t = 1$	High-occupancy	0.435	0.565
Probability of observation (B)			
		$O_t = 0$	$O_t = 1$
$q_t = 0$	Low-occupancy	0.257	0.743
$q_t = 1$	High-occupancy	0.265	0.735
Initial probability of each hidden state (π)			
		0.2	0.8

Table 5 **HMM for a room 212 during 2 pm to 6 pm.**

State transitions (A)			
		$q_{t+1} = 1$	$q_{t+1} = 2$
		Low-occupancy	High-occupancy
$q_t = 0$	Low-occupancy	0.421	0.579
$q_t = 1$	High-occupancy	0.504	0.496
Probability of observation (B)			
		$O_t = 0$	$O_t = 1$
$q_t = 0$	Low-occupancy	0.180	0.820
$q_t = 1$	High-occupancy	0.325	0.675
Initial probability of each hidden state (π)			
		0.6	0.4

for every observation symbol until the observation sequence ends that's is at step T. Finally, having all possible paths covered, the Viterbi algorithm [24] look for the path and output the most probable state sequence $Q = \{q_1, q_2, q_3 \ldots q_T\}$. Figure 6 shows the most probable state patterns of all identified stay locations using the trained HMM.

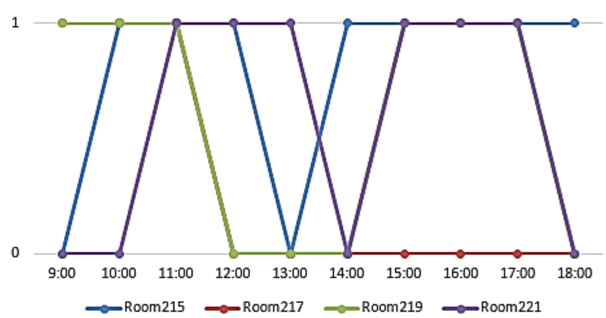

Figure 6 **Most probable state sequences of stay locations**

As it can been seen from the extracted patterns that there are fluctuations in occupancy levels observed during the different intervals of a day. However, these fluctuations are completely dependent on the type of a room (its properties can be extracted using a STriDE model). For example, meeting rooms (room217 and room219) will be occupied for a short period of a day whereas, work places (such as room215 and room 221) will be mostly occupied throughout a day. We have used 4 stay locations to predict their occupancy behaviors. Though, the number of stay locations should be dynamically extracted using the STriDE model that has an ability to hold the information of dynamic environments. As the proposed system prototype is still in the development phase and this article presents some initial results, further works need to be done to test the dynamic environment scenarios where purposes of locations are keep changing with the passage of time. With the change in the purpose, their properties as well as attributes in the data model will be changed as well. In addition, there will be different users trajectories and will have different stay locations in a building. These all dynamic interactions can be managed using the STriDE model that is designed in a way to capture the information of changing as well as moving objects in a dynamic building environment scenarios [12].

4 CONCLUSION

In this paper, we have used pre-existing STriDE model to enrich user trajectories with semantic information for understanding occupancy patterns of stay locations. The proposed system will serve not only as a semantic trajectory visualization platform for building facility managers but also acts as a predictive analytic solution to understand building occupancy for dynamic environments. Though, GPS technology is not recommended for indoor tracking solution as it introduce large measurement errors in trajectories because of the interference effects. These errors are observable in Figure 05 as collected trajectory data points didn't completely joined spatially with the semantic region information extracted from the OSM file. Consequently, the closest possible semantic regions have been mapped. After semantic enrichment, we have used HMM to estimate the occupancy patterns of a building by training from historical data. The trained HMM will allow to make real-time occupancy estimations by providing high-level information in improving the building performance as well as providing a solution to monitor critical locations of a building for safety monitoring applications.

5 ACKNOWLEDGMENTS

The authors thank the Conseil Régional de Bourgogne-Franche-Comté and the french government for their fundings.

6 REFERENCES

[1] Panagiotou, N., Zygouras, N., Katakis, I., Gunopulos, D., Zacheilas, N., Boutsis, I., Kalogeraki, V., Lynch, S. and O'Brien, B., Intelligent urban data monitoring for smart cities, Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2016, pp. 177-192.

[2] Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M., Internet of things for smart cities, IEEE Internet of Things journal, 1(1), 2014, pp.22-32.

[3] Gonzalez, M.C., Hidalgo, C.A. and Barabasi, A.L., Understanding individual human mobility patterns, Nature 453, 2008, pp. 779–782.

[4] Zheng, Y, Trajectory data mining: An overview, Transactions on Intelligent Systems and Technology 6(3), 2015, pp.1-41.

[5] Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S. and Aberer, K., 2013. Semantic trajectories: Mobility data computation and annotation. ACM Transactions on Intelligent Systems and Technology (TIST), 4(3), p.49.

[6] Zhao, N., Huang, W., Song, G. and Xie, K., Discrete trajectory prediction on mobile data, Asia-Pacific Web Conference, Springer, Berlin, Heidelberg, pp. 77-88.

[7] Zafari, F., Gkelias, A. and Leung, K., A Survey of Indoor Localization Systems and Technologies, arXiv preprint arXiv: 1709.01015, 2017.

[8] Ashbrook, D. and Starner, T., Using GPS to learn significant locations and predict movement across multiple users, Personal and Ubiquitous computing 7(5), 2003, pp.275-286.

[9] Xia, H., Qiao, Y., Jian, J. and Chang, Y., Using smart phone sensors to detect transportation modes, Sensors 14(11), 2014, pp.20843-20865.

[10] Spaccapietra S, Parent C, Damiani ML, de Macedo JA, Porto F and Vangenot C, A conceptual view on trajectories, Data and Knowledge Engineering 65(1), 2008, pp.126-146.

[11] Wu, F., Wang, H., Li, Z., Lee, W.C. and Huang, Z., SemMobi: A semantic annotation system for mobility data, Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion), 2015. pp. 255-258.

[12] Cruz, C., Semantic Trajectory Modeling for Dynamic Built Environments, IEEE International Conference on Data Science and Advanced Analytics (DSAA), Tokyo, Japan, 2017, pp. 468-476.

[13] Albanna, B.H., Moawad, I.F., Moussa, S.M. and Sakr, M.A, Semantic trajectories: a survey from modeling to application, Information Fusion and Geographic Information Systems (IF&GIS'2015), Springer International Publishing, 2015, pp. 59-76.

[14] Frihida, A., Zheni, D., Ghezala, H.B. and Claramunt, C., Modeling trajectories: A spatio-temporal data type approach, 20th International Workshop on Database and Expert Systems Application, DEXA'09, 2009, pp. 447-451.

[15] Parent, C., Spaccapietra, S., and Zimányi, E., Conceptual modeling for traditional and spatiotemporal applications: the MADS approach, Springer, New York, 2006, p 450.

[16] Yan, Z., and Spaccapietra, S., Towards semantic trajectory data analysis: a conceptual and computational approach, VLDB PhD Work 15(2), 2009, pp.165–190.

[17] Noël, D., Villanova-Oliver, M., Gensel, J. and Le Quéau, P., Design patterns for modelling life trajectories in the semantic web, International Symposium on Web and Wireless Geographical Information Systems, 2017, pp. 51-65.

[18] Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Damiani, M.L., Gkoulalas-Divanis, A., Macedo, J., Pelekis, N. and Theodoridis, Y., Semantic trajectories modeling and analysis, ACM Computing Surveys (CSUR) 45(4), 2013, pp.1-37.

[19] Zheng, Y., and Zhou, X. eds., Computing with spatial trajectories, Springer Science & Business Media, 2011.

[20] Qiao, Y., Si, Z., Zhang, Y., Abdesslem, F.B., Zhang, X., and Yang, J., 2018. A hybrid Markov-based model for human mobility prediction, Neurocomputing 278, pp.99-109.

[21] Leon-Garcia, A., Probability, statistics, and random processes for electrical engineering, 2017.

[22] Chen, Y., Introduction to probability theory, The lecture notes on information theory, Duisburg-Essen University, 2010.

[23] Rabiner, L., and Juang, B., An introduction to hidden Markov models, IEEE assp magazine 3(1), 1986, pp.4-16.

[24] Rabiner, L.R., 1989, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings of the IEEE 77(2), pp.257-286.

[25] Wong, J., Li, H., and Wang, S., Intelligent building research: a review, Automation in Construction 14 (1), 2005, pp.143 – 159.

[26] Harbelot, B., Arenas, H., and Cruz, C., LC3: A spatio-temporal and semantic model for knowledge discovery from geospatial datasets, Web Semantics: Science, Services and Agents on the World Wide Web, 35(1), 2015, pp. 3-24.

[27] Harbelot, B., Arenas, H., and Cruz, C., Continuum: A spatiotemporal data model to represent and qualify filiation relationships, Proceedings of the 4th ACM SIGSPATIAL International Workshop on GeoStreaming (IWGS), 2013.

[28] Arenas, H., Harbelot, B., and Cruz, C., A Semantic analysis of moving objects using as a case study maritime voyages from eighteenth and nineteenth centuries, the Sixth International Conference on Advanced Geographic Information Systems, Applications, and Services, 2014.

[29] Harbelot, B., Arenas, H., and Cruz, C, Using semantic web technologies to follow the evolution of entities in time and space, International Journal On Advances in Intelligent Systems, Freimut Bodendorf, University of Erlangen-Nuernberg, Germany, 2013, pp.256-265.

[30] Kamila, N.K. ed., Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis and Image Processing, IGI Global, 2015.