# Mobilytics: Mobility Analytics Framework for Transferring Semantic Knowledge

Shreya Ghosh, *Member, IEEE*, Soumya K. Ghosh, *Senior Member, IEEE*, Sajal K Das, *Fellow, IEEE*, and Prasenjit Mitra, *Senior Member, IEEE*

***Abstract*—The proliferation of sensor-equipped smartphones has led to the generation of vast amounts of GPS data, such as timestamped location points, enabling a range of location-based services. However, deciphering the spatio-temporal dynamics of mobility to understand the underlying motivations behind travel patterns presents a significant challenge. This paper focuses on how individuals' GPS traces (latitude, longitude, timestamp) interpret the connection and correlations among different entities such as people, locations or point-of-interests (POIs), and semantic contexts (trip-purpose). We introduce a mobility analytics framework, named *Mobilytics* designed to identify trip purposes from individual GPS traces by leveraging a "mobility knowledge graph" (MKG) and a deep learning architecture that automatically annotates the GPS log. Additionally, we propose a novel "transfer learning" approach to explore movement dynamics in a geographically distant area by leveraging knowledge obtained from a comparable region, such as an academic campus. In terms of major contributions and novelty, this is the first work to present end-to-end daily mobility trip purpose extraction and mobility knowledge transfer for trip annotation and POI-tagging where the labeled data are insufficient. Experimental results on real-life datasets of five different regions demonstrate the efficacy of our proposed Mobilytics framework which outperforms the baselines for trip-purpose extraction and POI annotations by a significant margin ($\approx$ 18% to $\approx$ 30%). Moreover, the analysis on huge volume of simulated traces (10,000 users) illustrates the scalability and robustness of the framework.**

***Index Terms*—Mobility knowledge graph, POI (point-of-interest), semantics, spatio-temporal trajectory, transfer learning.**

## I. Introduction

THE time-stamped sequence of location data (e.g., latitude, longitude) of any moving agent such as people or vehicle, is referred to as the *trajectory*. The huge volume of trajectory traces facilitate location based services such as individual mobility-based trip-recommendations, travel-time prediction, effective road traffic monitoring by mobile crowdsourcing etc. [1], [2]. However, most of the applications require semantic description of users' movement for better understanding of users' activity. This opens up an important research question: *how to interpret the intention or reason (i.e., trip purpose) behind individuals' movements?* Such trip-purposes are essential for travel behaviour modelling and travel demand estimation, and business investment decisions. Furthermore, it is the first step to creating text descriptions from GPS log that may be useful to develop an automated system for creating travel diary. However, extracting such interpretation of trajectory, and deriving usable knowledge from raw GPS trajectory/log is very challenging due to the diverse nature of human movement patterns, and deep learning based algorithms are inadequate in most of the real-life scenarios due to scarcity of labels and training data.

Our work generates a descriptive form of GPS trajectory by providing trip-purposes and activities performed at each trajectory segment and stay-point (see Section III for the definition). We propose an end-to-end mobility analytics framework, called *Mobilytics*, which is capable of extracting correlations among points-of-interests (POIs), temporal-scale and user's trip-purposes in the movement log. For example, the GPS footprint density of students' hangout spot ($H$) in an academic campus is closely related to the time-schedule of lectures and the distance between the lecture-hall and $H$. The footprint density at places like the library, sports facility, and cafeteria largely depend on the time-stamp of a day as well as external contexts (e.g., examinations, annual cultural-events). Furthermore, such contexts significantly influence (or alter) the time spent and the timestamp of visits at these stay-points.

The scarcity of labelled GPS logs (GPS trajectory with trip-purpose annotated) is the major limitation in any supervised learning technique to annotate trip-purposes. Transfer learning [3] is gaining significant attention for mitigating the gap between an incomplete learning task due to insufficient training data in the target region and extracted knowledge from the learning task in the source region. In this regard: *Can we map the mobility knowledge consisting of interrelations and connections among entities (users, POIs) from one academic campus (source region) to another campus (target region) where the labeled data (trip-purposes and POI-tags) is inadequate and achieve*

Manuscript received 21 March 2023; revised 27 February 2024; accepted 30 April 2024. Date of publication 12 June 2024; date of current version 5 November 2024. The work of Shreya Ghosh and Soumya K. Ghosh was partially supported by the SPARC Phase III under Grant 3385. The work of Sajal K Das was partially supported by the US National Science Foundation (NSF) under Grant SCC-1952045, Grant CNS-2008878, Grant OAC-2104078, and Grant EPCN-2319995. Recommended for acceptance by C. Assi. *(Corresponding author: Shreya Ghosh.)*

Shreya Ghosh is with the Department of Computer Science and Engineering, Indian Institute of Technology Bhubaneswar, Bhubaneswar 752050, India (e-mail: shreya.cst@gmail.com).

Soumya K. Ghosh is with the Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, India (e-mail: skg@cse.iitkgp.ac.in).

Sajal K Das is with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO 65409 USA (e-mail: sdas@mst.edu).

Prasenjit Mitra is with the College of Information Sciences and Technology, Pennsylvania State University, University Park, PA 16802 USA (e-mail: pmitra @psu.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TMC.2024.3413589, provided by the authors.

Digital Object Identifier 10.1109/TMC.2024.3413589

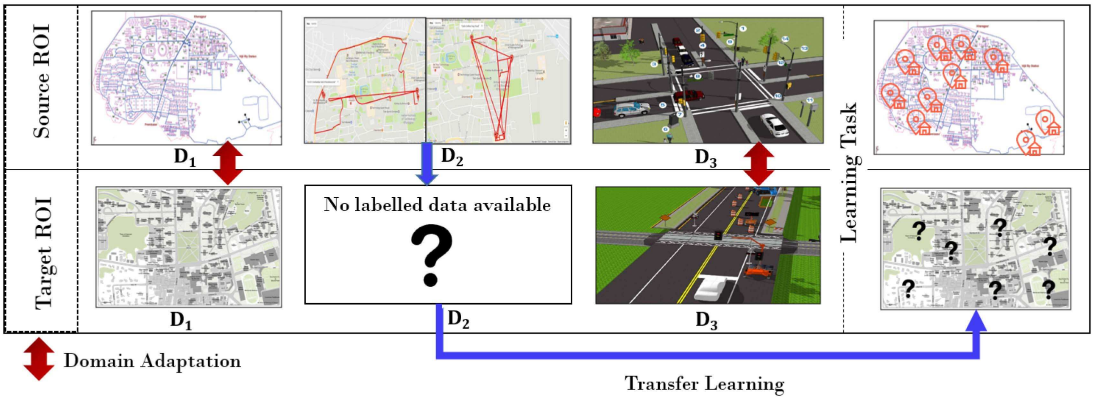


Fig. 1. General structure of mobility knowledge transfer set-up.

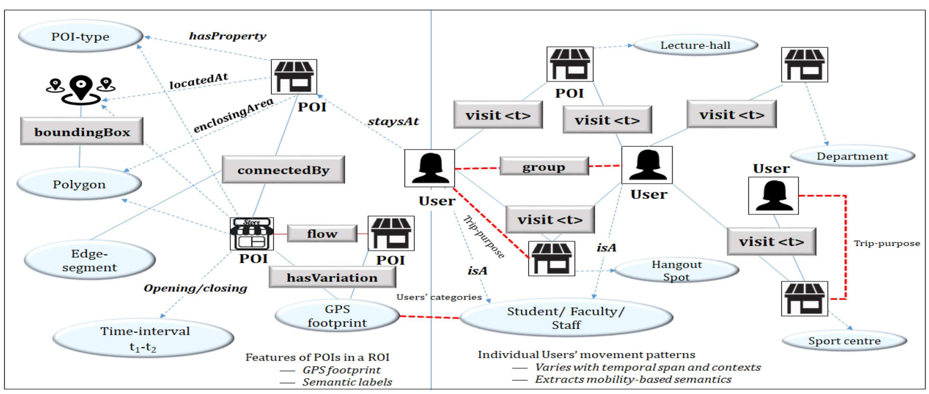


Fig. 2. Snapshot of a typical Mobility Knowledge Graph in a ROI (academic-campus). It shows how Mobilytics can extract structured knowledge from varied sources (GPS traces of individual users, aggregated footprint density, POI-types and related information directly from map services or using our analytics method) in a cohesive manner.

*accurate trip-purpose and POI identification?* To this end, we propose a *mobility-knowledge graph* (MKG) to capture the interdependence and connections among the users' movement behavior and POIs (see Fig. 2) at different time-scales, and subsequently transfer the extracted mobility knowledge to other regions for mapping the intent of movements. The knowledge graph [4] has the potential to model the complex relationship among users with varied trajectory patterns and POIs with different social functional spots.

However, the travel behavior changes based on the time, and the context (e.g., same POI can be used for two different activities at two different times of a day, or trip-purposes, such as, commuting to work or visiting friends) significantly influences travel behavior and travel mode. Therefore, both the edge label (e.g., trip-purpose, see Fig. 2) and node characteristics (crowd flow or activity performed at a POI) change based on time and day of a week). Most existing knowledge graph techniques leverage the embeddings learned from knowledge graphs as initial features ignoring the knowledge involved in the user trajectory trace [4]. For instance, using spatio-temporal features (e.g., time-duration spent, distance and time taken to reach next POI, frequency of visit etc.) we can extract trip-purposes and POI-types, which are essentially useful for a number of applications, such as, travel demand modelling, business settlement etc.

TABLE I
MOBILITY KNOWLEDGE TRANSFER LEARNING: APPLICATION SETUP

| Study Region | Source Domain | Target domain |
|---|---|---|
| Region-of-Interest | IITKGP (22.3145, 87.309) Area covered: 8.534 $Km^2$ | NITW (17.9837, 79.53) Area covered: 1.189 $Km^2$ |
| Aggregated movement pattern | Available | Available |
| Number of individuals' GPS log | 145 | 18 |
| Number of individuals' labelled GPS log | 145 | Not Available |
| Number and type of POIs | 236 | Not Available |

### A. Motivation

The real-life problems that we are attempting to solve are *identifying POI-type* (activity spots of a region) and *identifying trip-purposes of individual users*.

In the scenario depicted in Fig. 1 and Table I, we examine two regions of interest (ROIs): a source ROI (denoted as $SG$) and a target ROI (denoted as $TG$). We consider three primary data sources: a road network ($D_1$), labeled GPS traces of individuals ($D_2$), and crowd flow or aggregated GPS footprints ($D_3$) within different time scales of the ROIs. These data sources reveal correlations in movement semantics between points of interest

(POIs) across various time scales. Essentially, individuals move with specific intentions that are closely tied to the type or characteristics of the destination POI. The term "aggregated GPS footprint" refers to the density of footprints at POIs during different time intervals within a region, indicating the total counts of footprints at each time interval (e.g., 5-minute intervals throughout a day). This aggregated data aids in determining crowd flow at a POI by examining the intersections and overlaps of trajectories, as illustrated in Fig. 2 where the "visit" and "group" relations of the graph contribute to the aggregate count. Our primary data source ($D_2$) encompasses activities undertaken at a specific location (e.g., attending a lecture) or the intention behind a trip or movement (e.g., commuting for medical assistance). Acquiring such annotations or labels can be challenging, which is where "Mobilytics" comes into play. Mobilytics aims to learn from the source ROI ($SG$) through domain adaptation, enabling the mapping of this knowledge to another dispersed yet potentially connected target ROI ($TG$). This process allows for the annotation of users' unlabeled trajectories and the identification of POIs within the target ROI.

Let us consider a couple of real-world applications.

*(A) Urban Planning and Security:* For urban planning, Mobilytics uses D2 and D3 to create semantically enriched maps that inform the placement of new businesses or services, effectively predicting where a new restaurant or hospital might best serve the community. In security applications, the comparison of D2 against the backdrop of D1 and D3 aids in identifying anomalous behaviors–movements that do not correspond with the typical patterns associated with certain POIs, thus enhancing security measures.

*(B) Disaster Response and Management:* Mobilytics can play a crucial role in disaster-affected areas where the existing POI information may become outdated or irrelevant due to changes in the environment. For instance, after a natural disaster like a flood or earthquake, the road networks and local landmarks may be altered or rendered inaccessible. By analyzing real-time or recent GPS traces (D2) and crowd-flow data (D3), Mobilytics can quickly identify new navigational routes, temporary shelters, and resource distribution points. This information is critical for coordinating rescue and relief operations efficiently, ensuring resources are directed where they are most needed. Moreover, by comparing the pre-disaster and post-disaster mobility data, Mobilytics can assist in assessing the impact on infrastructure and community mobility, thereby facilitating a more informed recovery and rebuilding process.

Specifically, Mobilytics can be applied to (a) map POIs of a new city (where Google or other map services are not yet reliable) and identify new POIs from movement semantics, (b) understand trip-purposes and help identify specific areas that are not disclosed due to, e.g., defense activities (by identifying trips which do not have common mobility characteristics nor correlate with associated POI-type). Furthermore, if we deduce the usual trip-purposes and associated movement characteristics, then anomalous trajectory can be identified, e.g., enhancing security measures (such as the mobility behavioural difference between a user and a pickpocket suspect [5]) (c) identifying different activities at a multi-storied building (which are often not labelled by Map services), (d) Finally, Mobilytics may help in creating a theoretical foundation on extracting and analysing characteristics of different trip-purposes in a city region which may help in creating semantically enriched synthetic trajectory traces that can be used for research, simulation, and further tool-building. Annotating trip-purposes is also instrumental for allocating resources in advance or identifying suitable locations for new business (POI, say, restaurant, hospital, etc.) settlement.

### B. Our Contributions

The major contributions of this work are summarized below.

1) *Mobility Knowledge Graph:* We elaborately define and develop a novel time-dependent *mobility knowledge graph* (*MKG*) to model movement semantics among locations and temporal information of the region. This is the first work to develop a knowledge graph completion problem where massive volume of trajectory traces, temporal information and POIs are jointly modelled and represented to address mobility trip-purpose and POI identification problems where only limited labelled trajectory data is available. [See Section IV-B]
2) *User Mobility Semantics:* We describe a novel deep learning architecture for annotating GPS (movement) traces with *POI-tags* and *trip-purpose*.
3) *Mobility Knowledge Transfer:* Mobilytics presents a *transfer learning* technique for transferring mobility knowledge from source region to target region and *MKG completion*, which means the incomplete information of the entities and relations are updated.[1]
4) *Performance Evaluation:* The *Mobilytics* framework is evaluated with real-life datasets from two academic campuses in India, one campus in Australia, and GPS traces from China [See section VI-A].[2] The evaluation in terms of accuracy, precision and execution time demonstrate that our *Mobilytics* framework provides valuable insights regarding the correlations of peoples' moves, locations, contexts and finally transferring knowledge (i.e., using labelled MKG of source region to complete MKG in target region). Further, we have carried out an extensive study on simulated traces of 10,000 users to illustrate the scalability of the Mobilytics framework.

The rest of the paper is organized as follows. Section II summarizes related works. Preliminary concepts are introduced in Section III, while the *Mobilytics* framework consisting of mobility knowledge graph and deep learning modules are presented in Sections IV and V, respectively. Experimental performance evaluation is described in Section VI and conclusions are offered in the final section.

[1]The intuition is that the placement of POIs may be different, however the spatio-temporal characteristics of similar trip-purposes are similar in source and target regions.

[2]The datasets from India and Australian university have been collected voluntarily.

## II. Related Work

### A. Semantic Trajectory Data Mining

Most trajectory analysis works in the literature represent the trajectories as episodes of *stop* and *move* [1]. Nevertheless, *semantic* information, e.g., POI-type of stay-point, transportation mode, needs to be augmented with the raw GPS log to facilitate intelligent location based services (e.g., user-profiling, trip recommendation etc.) to capture dynamic user needs and provide meaningful suggestions. The formalization of a semantic-enriched knowledge discovery process is described in [6] for interpreting human movement behaviors. In [7], the authors studied the regional movement patterns and developed an efficient algorithm to find out such patterns in semantic trajectories. A collective embedding framework is presented in [8] to extract the community structure from spatio-temporal graphs of human mobility. The authors proposed a probabilistic propagation approach and used deep auto-encoder methods, and showed two use cases to evaluate the efficacy of their approach. In [9], a framework is introduced that can generate insightful embeddings for points of interest (POIs) by incorporating semantic and contextual information from diverse sources. Additionally, Wang et al. [10] propose "FROST", which focuses on optimizing facility placement by analyzing user movement patterns. However, while the aforementioned works primarily address semantic trajectory data mining, there remains a gap in the exploration of extracting trip purposes from historical movement logs.

### B. Knowledge Graph Embedding

The authors in [4] presented a spatial knowledge graph to profile users' mobility and predict the next location sequences. By introducing an imitation-based criterion for profiling accuracy, where the accuracy is highest when an autonomous agent can mimic a user's activity patterns, the authors propose a framework where an agent plans next visits based on a user's profile, improving profiling through interactions between users, spatial entities, and a spatial knowledge graph and demonstrates enhanced performance in predicting human mobility activities, showcasing its potential for incremental learning in user profiling and other applications. However, the authors only emphasized on users visiting specific locations without any semantic labelling of the trips. In [11], an approach (TransR) for entity and relation embedding is proposed for link prediction. This novel approach for knowledge graph completion constructs entity and relation embeddings in separate spaces, addressing the limitations of previous models like TransE and TransH which embed entities and relations in the same semantic space. TransR improves upon these models by projecting entities from an entity space to a corresponding relation space before performing translations, demonstrating significant improvements in link prediction, triple classification, and relational fact extraction tasks over state-of-the-art baselines. Our proposed mobility knowledge graph embedding is motivated by this work, however, we defined new mobility relations and proposed a new embedding technique to capture the movement semantics. Another work, named *StructRL* [12], formulates mobile user profiling by deep representation learning. The authors deploy an adversarial substructured learning framework for modelling activities of users and forecasting next activities. However, the method needs ample number of training data-instances to learn the parameters of their proposed model.

### C. Deep Learning Based Model

Deep learning has recently gained significant research attention in the trajectory data mining community [13]. An unsupervised neural network based framework is proposed in [14] to cluster similar mobility behaviours. A deep learning framework in [15] identifies living patterns in a population effectively. A semi-supervised deep learning model named, SECA, is proposed in [16] for transportation mode identification. Since the unavailability of labeled datasets is one of the major concerns in supervised learning techniques, the *transfer learning* paradigm [3] offers a way forward, thereby reducing the expensive data-labelling efforts. For instance, a transfer learning framework is proposed in [17] for mobile traffic prediction leveraging generative adversarial network based transfer learning over spatio-temporal data. A transfer learning technique presented in [18] generates spatial trajectories in the target city where no mobility data is present. This work mainly focuses on generating paths in the new city by learning path preferences in the source city, however, does not study the trip-purpose or POI annotation aspect. The authors in [19] estimate the *Home-to-Work* time for citizens from the survey data of the source and target cities. A transfer learning technique proposed in [20] detects parking hotspot by exploiting multi-source data for discriminative feature learning. The POI recommendation framework in [21] leverages transfer learning of users' movement behavior in the home country to the new city visited.

### D. LLM-Based Model

Recently, researchers have started to leverage large language models (LLMs) for time-series analysis. The authors in [22] introduces LLM-Mob, a method utilizing LLMs for predicting human mobility by incorporating historical and contextual data, showing superior prediction accuracy and interpretability, and suggesting a shift towards using general-purpose LLMs in mobility studies. GeoLLM [23] is a novel approach that leverages LLMs and OpenStreetMap data to improve geospatial prediction tasks. It demonstrated significant advancements in predicting population density and economic indicators, surpassing traditional methods and matching satellite-based benchmarks. However, despite these successes, Large Language Models (LLMs) require extensive pre-training on massive datasets, which may not be directly applicable to specific tasks like trip purpose detection that rely heavily on understanding spatial and temporal semantics. On the other hand, integrating mobility knowledge graphs (MKG) and transfer learning, as done in Mobilytics, can significantly enhance LLMs' performance in mobility analytics. The MKG provides a structured framework that enriches LLMs with domain-specific insights and contextual understanding necessary for accurate trip purpose detection and POI tagging.

Additionally, the transfer learning approach allows for the application of knowledge from one geographic area to another, compensating for LLMs' limitations in dealing with sparse or region-specific data.

To the best of our knowledge, there hasn't been prior research specifically dedicated to constructing a mobility knowledge graph (MKG) aimed at interpreting user trip purposes and transferring this mobility knowledge to another geographically dispersed region, particularly in cases where labeled data are limited. The concepts of MKG construction and knowledge transfer utilizing deep learning architecture represent novel contributions.

## III. Definitions and Terminologies

*GPS Trajectory* ($G$): It refers to time-stamped location information represented by $loc :< latitude, longitude >$ or (lat, lon). This term is used interchangeably with GPS trace to denote the path or movement recorded over time.

*POI-taxonomy*: Point-of-interest (POI) represents the activity-spots or socio-functional region (e.g., residential building, sports complex, etc.) of a city. The POIs of study region has been represented in a tree-structured taxonomy (POI-taxonomy). In this tree-based structure, the leaf nodes hold specific addresses while the internal nodes represent more generic POI-tags.

*Region-of-interest (ROI)*: The bounding-box captures the ROI which is divided into uniform square grids, of spatial resolution is $100\,m \times 100\,m$. This is determined by considering a balance between spatial resolution and computational efficiency. This size is particularly well-suited to urban and suburban environments where points of interest are densely packed and diverse in function.

*Stay-point* ($S$) : Stay-point of a user trajectory is defined by a set of GPS-points in spatial proximity and the time spent at this point is greater than a temporal threshold (see Section IV-A). Stay-point $S(B, pl, T)$ is represented by the *bounding box* ($B$), POI-tag ($pl$), and time-interval ($T$) of the visit. The bounding-box is represented by *polygon* geometry [24] and encloses the area covered by the user during $T$.

*Trajectory Segment* ($Traj\_Seg$): It is represented as:

$$Traj\_Seg = \{S_1(B_1, pl_1, T_1) \longrightarrow (l_1, t_1) \longrightarrow \cdots \longrightarrow (l_n, t_n) \longrightarrow S_n(B_n, pl_n, T_n)\} \quad (1)$$

where $S_1$ and $S_n$ are two stay-points. It's important to observe that each $Traj_Seg$ is composed of consecutive stay-points along with the GPS log between these stay-points. In cases where there are no preceding or succeeding GPS points available for a user, the starting and ending points of the trajectory are automatically considered as default stay-points.

*Movement Diary*: The movement-diary is the log of the semantic meaning or *intent* of a trip (e.g., visit from one stay-point to the next stay-point). In this work, we consider 56 such trip-purposes (see Appendix) and collect trip purposes from the users using a web-based app (see Section VI-A).

*Problem Definitions*

**Trajectory segmentation annotation**: Given the trajectory trace (timeseries data of latitude, longitude of an individual user), the model outputs the corresponding labels of the stay-point transitions[3] (e.g., commuting to workplace) and activity type (attending lecture) at a specific stay-point/ POI-type (e.g., academic department).

**Mobility Knowledge Transfer**: Given a source ROI (labelled movement traces and POIs) $SG$ and the classification task (trajectory annotation) $\mathcal{PC}_{\mathcal{SG}}$, a target ROI $TG$ (unlabelled movement traces) and classification task $\mathcal{PC}_{\mathcal{TG}}$, the proposed transfer learning module aims to improve the learning of target function $f_{\mathcal{PC}}$ in $TG$ utilizing the knowledge of $SG$ and $\mathcal{PC}_{\mathcal{SG}}$, where $SG \neq TG$ and $\mathcal{PC}_{\mathcal{SG}} = \mathcal{PC}_{\mathcal{TG}}$.

## IV. Trajectory Pre-Processing and MKG

### A. GPS Error Removal and Trajectory Segmentation

In the initial step, the GPS log is pre-processed to guarantee the increasing order of timestamp and remove duplicate points with same timestamp.

$$l_1(lat_1, lon_1, t_1) \rightarrow l_2(lat_2, lon_2, t_2) \rightarrow l_n(lat_n, lon_n, t_n), \quad where\ t_1 > t_2 > \ldots > t_n, \forall (l_1, l_2, \ldots, l_n) \in G \quad (2)$$

The *Kalman Filtering technique* [1] is deployed to remove the GPS errors due to low number of satellites in view or device error (large scale error) and small scale random error (e.g., GPS points out of the boundary of a POI or out of the road the vehicle was actually driven). The measurement noise ($v_i$) and process noise ($w_i$) are defined as:

$$w_i \sim N(0, Q_i); \quad v_i \sim N(0, R_i) \quad (3)$$

The process noise is represented by $Q$ and $R$ denotes measurement noise covariance matrix. In our experiment, GPS data log is collected from individuals' Google Map Timeline, where an accuracy value indicating the measurement accuracy is logged in the mobility data file (See Appendix A). Therefore, $R_i$ is the function of this accuracy and $Q_i$ is the amount of time elapsed between two measurements $i$ and $i-1$. We have followed the detailed process of using kalman filtering in trajectory pre-processing as proposed by Lee and Krumm [25]. The GPS points are smoothed using the above-mentioned filtering technique.

Trajectory segmentation is the process of fragmenting the complete path into different length of location sequences visited by the moving agent. In this paper, we segment the trajectories such that a segment holds consecutive stay-points and the path between the stay-points followed by the agent. To detect the stay-points from the trajectory, two scale parameters are required: distance ($dis_{th}$) and time ($t_{th}$) thresholds. Typically, stay-point occurs if an individual stays stationary for atleast $t_{th}$ time. This mainly happens while people enter in a POI (building) and the GPS signal is unavailable. In other cases, several GPS points are logged within a certain spatial region ($dis_{th}$) for a

TABLE II
MKG RELATION NAMES AND DESCRIPTION

| MKG relation name | MKG relation |
|---|---|
| $MR_{visit}$ | $\{(u_i, poi_j), [t_s, t_e], f_x\}$ |
| $MR_{group}$ | $\{(u_i, U' \setminus u_i, \exists(poi_1, \ldots, poi_m), [t_s, t_e], f_x\}$ s.t. $\forall_a^n MR_{visit}(u_a, poi_1, [t_s, t_s'])\wedge$ $MR_{visit}(u_a, poi_2, [t_2, t_2']) \cdots \wedge$ $MR_{visit}(u_a, poi_m, [t_m, t_e])$ |
| $MR_{flow}$ | $(poi_1, poi_2, [t_s, t_e], f_x)$ $\exists(u_1, \ldots, u_n)\forall_j^n$ $MR_{visit}(u_j, poi_1, [t_s, t_s^{'}], f_x)$ $\wedge MR_{visit}(u_j, poi_2, [t_e, t_e^{'}], f_x)\ \ n \geq \nu$ |

period which means the user is wandering outside and attracted by the surroundings. In this work, a spatio-temporal sequence of GPS points $GS = (loc_a, t_a); (loc_{a+1}, t_{a+1}); \ldots (loc_b, t_b)$ are converted into a single stay-point, if $t_b - t_a >= t_{th}$ and $\forall a < i <= b, Distance(loc_a, loc_i) <= dis_{th}$. We set $t_{th} = 08$ mins and $dis_{th} = 200$m in the stay-point detection process. In the first step, if a user stays more than 8mins within 200 m distance, then it is detected as a stay-point. Next, the geotagg information of the mean coordinate $(\sum_{i=a}^{b} loc_i.lat/|GS|, \sum_{i=a}^{b} loc_i.lon/|GS|)$ is extracted.

To label land use information, we employ a method known as *reverse geocoding*, which involves utilizing the Google Place API service to extract POI tags or geotags from Google Maps. We mention the "opening_hours" (POI's business hours) in the parameter-list of the reverse-geotagging procedure. The *. json* file returned by the API service consists of an array-list named *periods* (a pair of day and time objects describing POI's business hours). We eliminate the POIs that are not open in the time of stay-duration of the trajectory. If multiple landmarks or POIs (L: list of landmarks) are found that remain open (or active), then the nearest one from that GPS point is selected. Since the extraction of geotags or POI information is time consuming, we extract all the POI information within 200 m in the initial stage itself. It is important to clarify that our methodology utilizes land use/ geotagged information from Google Maps as an initial dataset for the source region, ensuring that the data is comprehensive and up-to-date. However, the core premise of Mobilytics is to leverage the movement semantics from well-mapped source regions to learn intricate mobility patterns and behaviors. Once these patterns are comprehensively understood, the learned model can then be applied or 'transferred' to target regions with potentially less precise data. Moreover, we have shown that the performance of Mobilytics can be further enhanced by combining data from two or more source regions (See Table V). This approach provides a robust fallback in situations where one region's data might not be as current or detailed, thereby ensuring the model's effectiveness and generalizability.

The OpenstreetMap (https://www.openstreetmap.org/) is used to extract the road network structure. This information provides geometric information of the road segments (length, width) and the connectivity and continuity (e.g., intersection) of the road network. The AntMapper map-matching algorithm, introduced in [26], is utilized to align the raw GPS log with the relevant road segments. This algorithm incorporates both topological information and a global similarity value during the mapping process. Subsequently, trajectory segments are established by connecting any two consecutive pairs of stay-points along with the path traversed by the individual. The trajectory-pattern of individual over days is modeled using Dynamic Bayesian network (probabilistic graphical model) named $Traj\_Window(V', E', \Upsilon)$ where $V'$ and $E'$ are the set of stay-points and the direction of visit among different stay-points respectively [27]; and $\Upsilon$ denotes the conditional probability tables of the nodes (stay-points) representing the degrees of dependence between variables. Here, *Traj_Window* is used to model and represent the movement patterns of an individual, rather than a single days' trajectory. The key reason to utilize probabilistic graphical model is the conditional dependency of stay-points in the movement path. Given that the user is present at a POI $(P_1)$ at time $(t_1)$, the probability of visiting the sequence of POIs (say $P_2$ at $t_2$ time) are captured in the Traj_Window. This *Traj_Window* is used to model the movement pattern of an individual over time by modelling the relationships between multiple time series and also different regimes of movement behaviors.

### B. Mobility Knowledge Graph (MKG) Construction

The *mobility knowledge graph (MKG)* captures the movement semantics (individual movement behavior and POI characteristics) of the region. Specifically, MKG is crucial for our framework for several reasons. (i) We need to model the movement semantics irrespective of the spatial (distance between two academic units or residential places) and temporal (time to travel from residential place to workplace) constraints. This helps extract the movement behavior and correlate with similar movement patterns at different region (where POI distribution is different); (ii) facilitate information retrieval and query-processing (see mobility relations such as group, visit, flow, and corresponding parameters defined later) by transforming the mobility semantics into machine readable format (Mobility traces can be easily stored and analysed as *stop* and *move* that can be represented by nodes and edges of a graph); and (iii) extracting previously unknown mobility patterns as well anomalies by merging more than one relation of the graph. (See mobility relations *visit*, *flow* from one node to another in Table II.)

*1) MKG: Entities and Relations/ Facts:* The mobility knowledge graph is represented as a directed graph that is composed of time-dependent mobility-relations (MR). Formally, MKG is defined by $< u, MR, p, t, f >$, where $u, p, t$ are *spatio-temporal entities* and $f$ denotes the strength (probability that each fact happens) of the correlation. The spatio-temporal entities consist of users $(u)$, places $(poi)$ and time-intervals $(t_s, t_e)$. The mobility relations are extracted based on the historical movement log. Table II denotes three different MRs used in this work: $MR_{visit}$: This relation captures the visits of users to points of interest (POIs) within a specific time interval. It is represented as a tuple containing the user identifier $(u_i)$, the point of interest $(poi_j)$, the start and end times of the visit $([t_s, t_e])$, and a function $f_x$ that represents probability of the visit. This relation allows the graph to capture individual user movements to specific locations. $MR_{group}$: This relation represents a group mobility pattern where multiple users visit a sequence of POIs together within a given time frame. The visit sequence is represented by a series of $MR_{visit}$ relations, each capturing a visit to one of the $m$ POIs in the sequence by user $u_a$ within specific time intervals. This relation is useful for identifying social or collective mobility patterns, such as a group of people moving from one venue to another together. $MR_{flow}$: This relation describes the movement flow between two POIs over a time interval. It is defined by the existence of visits from a set of users $(u_1, \ldots, u_n)$ to the first POI $(poi_1)$ starting within a given time frame $([t_s, t'_s])$, and then to the second POI $(poi_2)$ ending within a specific time frame $([t_e, t'_e])$. The $MR_{flow}$ relation is essential for understanding traffic patterns or migration between locations within the knowledge graph.

Fig. 2 illustrates a sample MKG in an academic-campus ROI. By constructing MKG, we attempt to model movement dynamics of the ROI. The entities and some relations among the entities namely *visit*, *group*, *flow* are shown in Fig. 2. In the knowledge graph, the entities are users and POIs. Each of the entities has some attributes, which are denoted by oval-shape object and dotted line. For instance, the attributes of a user are user-category (faculty/student/staff) and her permanent residence. Similarly, the attributes of a POI are POI-tag, opening/closing hours of the POI. The blue lines show the relations among the entities. Here, we have introduced time-dependent facts since movement dynamics can not be represented by static fact representation. The relation *trip-purpose* is not embedded in the movement log for all trips in the database. Therefore, this relation is marked with different color. The left side of the figure depicts the features of POIs, among which some of are directly extracted from the information provided (such as location of the POI, POI-type and opening/ closing hours from the geotagged information), and some fact like GPS footprint variation in different time-scale is observed by analyzing the aggregate movement trace in the study-period. The right side of the figure mainly deals with users' movement patterns and related facts such as visit-sequences, group and trip-purpose. The relation *connectedBy* represents the road-segments to connect any two POIs. This relation is extracted by the map-matching process defined in Section IV-A. The relation stores an arraylist containing the road-ids. The relation *boundingBox* represents the polygon coverage area of each POI. In the reverse geo-tagging procedure, we extract the POIs covering 200 m of any point. From this information, we find out the coverage of each POI and compute the bounding-box. The relation typically stores the lower left and upper right coordinates of the bounding-box.

To capture the variation in GPS footprints over different timescales, we divide a day into 15-minute time slots starting from 8:00 in the morning. For each point of interest (POI) in the knowledge graph, we represent the number of starting trips $(ST)$ and set-up trips $(SU)$ at different time slots of a day (e.g., $L_1$) using the vector notation:

$$ST_{L_1}^{p_a} = (ST_{t_0}^{L_1,p_a}, ST_{t_1}^{L_1,p_a}, \ldots, ST_{t_m}^{L_1,p_a});$$
$$SU_{L_1}^{p_a} = (SU_{t_0}^{L_1,p_a}, SU_{t_1}^{L_1,p_a}, \ldots, SU_{t_m}^{L_1,p_a}) \quad (4)$$

Next, we create two matrices for each POI representing the counts of start and set-up trips by analyzing GPS logs $(L_1, \ldots, L_d)$ of $d$ days. We then apply an autoregressive integrated moving average technique to determine the values of start and set-up trips in different time slots based on historical movement traces. This process enables us to capture the GPS footprint variation (denoted as "hasVariation" mobility fact) of different POI entities in the knowledge graph.

*How MKG relations can be useful in realistic scenarios:* MKG relations (See Table II) are crucial for understanding and analyzing mobility within a region by capturing individual and collective movement patterns and the flow of movement between different points of interest. They enable the translation of raw mobility data into structured, query-able information that can facilitate a range of applications from urban planning to traffic forecasting and personalized location-based services. For instance, in emergency response and disaster management, (a) $MR_{visit}$ can help emergency services predict which areas are likely to be most populated at certain times, allowing for more efficient evacuation route planning and resource allocation, (b) $MR_{group}$: understanding group movement patterns $(MR_{group})$ can be critical for coordinating evacuation efforts. If certain POIs are known to be gathering points for groups (e.g., schools or community centers), these can be prioritized in evacuation or emergency response planning, (c) $MR_{flow}$ relations can be analyzed to predict how people are likely to move between areas during an emergency. This information can be vital for managing congestion on evacuation routes and for planning emergency response logistics, such as where to place ambulances and first responders for the quickest access to those in need.

*2) MKG: Embedding Technique:* The MKG embedding has three steps: (a) represent *POIs* and *Users*; (b) propose a scoring function, and (c) learn the representations of POIs, users and mobility-facts at different time-scales. The entities (POIs and users) are usually represented as *vectors* and the relations (mobility-facts) are defined by operations in the vector space [28]. Next, a scoring function measures the plausibility of each mobility fact at different time intervals. In our MKG-embedding, we propose a new embedding by augmenting *Rotation and Translation Embeddings (RotatE)* [29] and *semantic matching model* [30], with some modifications to adapt to the movement semantics of a region. RotatE is known for its flexibility and effectiveness in capturing various relation patterns (e.g., symmetry/antisymmetry, inversion, and composition) in a knowledge graph where the semantic matching model aims to learn vector representations of entities and relations by capturing the similarity between the related entities and relations. We select TransE (Translation-based Embeddings), a semantic matching model that represents mobility relations as translations in the embedding space.

Given the fact $MF_i = < s_i, r_i, o_i, [t_1, t_2], f_x >$ of our MKG, we represent two entities $s_i$ and $o_i$ respectively by complex-valued vectors $\mathbf{s}_i \in \mathbb{C}^{dm}$ and $\mathbf{o}_i \in \mathbb{C}^{dm}$, where $dm$ is the dimensionality of the embeddings. The strength of the relation (probability that the fact happens in that time-interval) is denoted by $f_x$. The spatial information is embedded by $sp(s_i, o_i)$ which measures the spatial distance (Haversine distance) between the

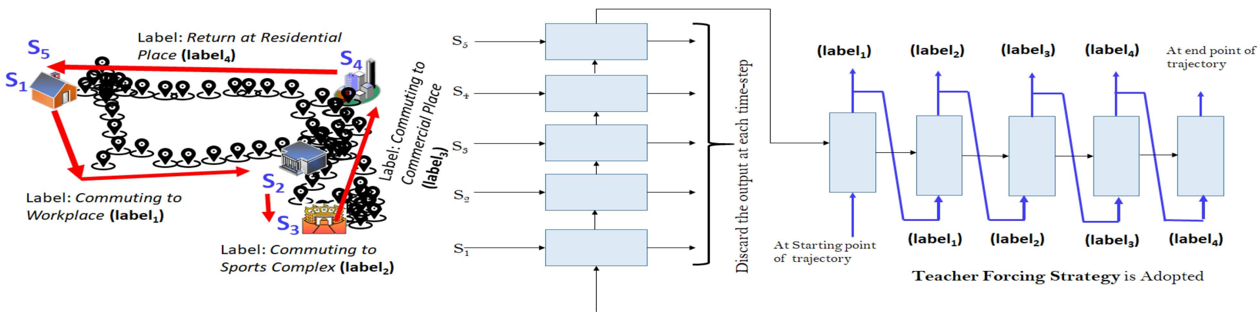


Fig. 3. Illustration of the training phase of Model I.

head entity (if $s_i$ is a user, it considers the user's present location) and the tail entity ($o_i$, a POI). The temporal vector ($\tau$), defined by the *time spent at the POI* ($T_{spend}$) and the *time taken to visit the POI* ($T_{visit}$), measures the time taken by a user to reach the POI from its previous location. These are combined as:

$$\tau(s_i, o_i, [t_1, t_2]) = \zeta \times (t_2 - t_1) + \phi \times T_{visit}(s_i, o_i) \qquad (5)$$

where $\zeta$ and $\phi$ are hyperparameters that control the importance of time spent at the POI and the time taken to visit the POI, respectively. Next, the scoring function ($score(s_i, r_i, o_i, [t_1, t_2])$) is defined as:

$$\begin{aligned} &= -\,(w_r * \|s_i * r_i - o_i\|_1 + \Psi * sp(s_i, o_i) \\ &\quad + \psi * \tau(s_i, o_i, [t_1, t_2])) \\ &= -\,(w_r * \|s_i * r_i - o_i\|_1 + \Psi * sp(s_i, o_i) \\ &\quad + \psi * (\zeta \times (t_2 - t_1) + \phi \times T_{visit}(s_i, o_i))) \qquad (6) \end{aligned}$$

where weight $w_r = exp(-\lambda * f_x)$ is introduced for each relation based on the strength feature ($0 \leq f_x \leq 1$) and $\lambda$ is the hyperparameter that controls the influence of the relation strength on the weight. MKG embedding captures varied relation patterns: (1) Symmetry/Antisymmetry: The "visit" relation is considered symmetric, as it implies that if a user visits a POI, the POI is also visited by the user, while the "flow" relation is considered antisymmetric, as the flow of users from one POI to another might not necessarily be the same in the opposite direction. (2) Inversion: The "meet" relation is an example of inverse relation, such as "part" (indicating that users part ways). When users $s_i$ and $s_j$ meet ($s_i, meet, s_j$), it implies that they were not together before ($s_j, part, s_i$). (3) Composition: The "group" relation is a composition of two other relations, such as "meet" and "visit". By capturing these relation patterns, MKG can represent the complexities of the relationships between users and POIs, leading to a more accurate representation of the movement semantics in the region.

The training objective in the MKG embedding is to minimize the ranking loss ($\mathcal{L}$) of the true triples ($s_i, r_i, o_i$) and their corrupted counterparts as follows:

$$\begin{aligned} &\mathcal{L}(s_i, r_i, o_i, [t_1, t_2], s'_i, r'_i, o'_i, [t_1, t_2]) = \max(0, \Lambda \\ &\quad + score(s_i, r_i, o_i, [t_1, t_2]) - score(s'_i, r'_i, o'_i, [t_1, t_2])) \qquad (7) \end{aligned}$$

where $\Lambda$ denotes non-negative hyperparameter that determines the desired separation between true and corrupted triples. The ranking loss function aims to ensure that the true triples have lower (better) scores than the corrupted triples. The training objective is to minimize the average ranking loss over all true triples and their corresponding corrupted triples in the training dataset using stochastic gradient descent (SGD). In summary, MKG captures the intricate dynamics of user mobility and be applied to various downstream tasks (such as mobility knowledge completion, and transfer learning) in the mobility context.

## V. Deep Learning Modules

### A. Model I: Annotation of the Trajectory Segments

Fig. 3 illustrates the problem statement and the adaptation of the basic encoder-decoder (many-to-many) model.

Here, we present the method of annotating (or *labeling with trip-purpose*) the trajectory-segments of an user by the proposed deep learning architecture (model I). To begin with, we define the *trip-purpose* as a semantic label for the transition from one stay-point to another (see Section VI-A for a few trip-purposes). To extract trip-purposes, we assume the following facts: (1) The stay-points of an individual are always either associated with *POIs* or represent some *group* activities. This assumption is realistic, since users spend a specific amount of time in the stay-point. The reason behind this stay may be some activities performed in the POIs (say, attending lectures in the lecture-hall) or any group-activity with other users. (2) The semantic labels of the trips provided by the individuals in their movement diaries are accurate. This assumption is convincing since the intuition behind any movement can be interpreted by the individuals correctly. Also, in the data collection process, we provided a fixed set (total 56) of trip-purposes as the labels. If an user finds her trip-purpose is not present within the list, the user can select "others" and provide additional label for that specific trajectory segment. We found only a very few trajectory segments ($\approx 2\%$) were labelled as "other". The synonymity problem is negligible. (3) The *day of the week*, *stay-duration*, *sequence of the visits*, *timestamp* of the visit directly influence the semantic-label or the trip-purpose. However, the trip-purpose labelling task is not straightforward. First, it's important to note that two trajectories, despite having significantly different spatial and temporal scales, may exhibit similar movement behaviors. For example, one's

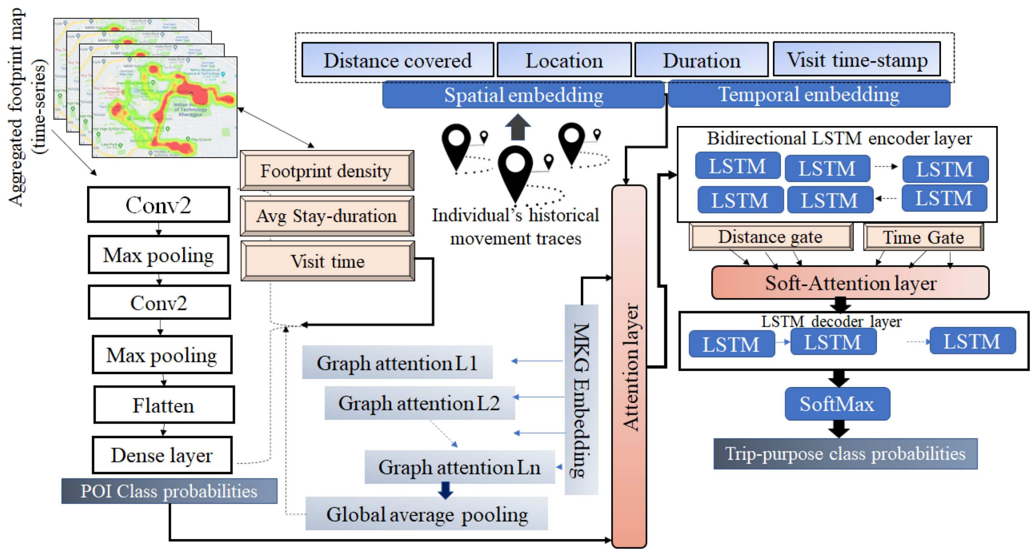


Fig. 4. Deep learning architecture for model I.

commute to work may take fifteen minutes and the other may take one hour. Moreover, the spatial and temporal scales of the trajectories representing similar moving behaviors may not be similar. For example, a one-hour commute may span only 5 miles (say with a bicycle) or 50 miles (say with a car). Even for the same mode of transportation, we may have similar commute times with different spatial scales due to traffic. Due to these challenges, any approach relying on similarity measures using raw spatial and/or temporal features of a trajectory, such as Dynamic Time Warping (DTW) or Longest Common Subsequence (LCSS) distance [1], cannot be used as a robust method for extracting similar movement behaviours that depend on temporal and geographic context of the trajectory. Again, a work commute happens on workdays, which usually starts in the morning from a residential area to a business area and vice-versa in the afternoon. There are also several external contexts which significantly influence the movement behaviours of the individuals and consequently the trip-purposes. For example, in the *examination-week* of the semester or in the *college festival time* the trip-purposes differ from the daily movement patterns. These *context-shifts* must be captured and incorporated in the model for an efficient intent mapping. Other existing trajectory distance metrics (DTW, LCSS etc.) fall short to incorporate all such contexts in the similarity measurements. To this end, deep hierarchical models is one of the feasible solutions [14] which can distinguish among several trip-purposes by learning the latent representations of mobility data along with other contexts.

Our deep learning architecture is depicted in Fig. 4. The *collective task layer*, implemented through a Convolutional Neural Network (CNN), processes combined movement data to categorize Points of Interest (POIs) within a given area. It incorporates the analysis of the spatial patterns of visitation intensity and the timing aspects (duration of stay and visit timing) to perform this categorization task effectively. The incorporation of MKG as graph attention layers serves to refine the accuracy of POI classification further. Meanwhile, the *individual task layer* processes annotated trajectory data (denoted as $TR, L$), with the model's decoder generating corresponding classifications. The approach adopts the teacher-forcing technique [31], which improves classification efficiency by utilizing both actual training sequences and the model's predicted outputs. To account for the influence of preceding and subsequent points in a trajectory, the model introduces the temporal-constraint gate ($TiG$) and spatial-constraint gate ($DG$). Additionally, to capture spatial and temporal relationships, the model employs the skip-gram and paragraph vector models for crafting spatial and temporal embedding modules, focusing on adjacent grid points denoted as $loc_{t+j}$ and $loc_t$.

$$\frac{1}{T}\sum_{t=1}^{T}\log[p(loc_{t-c},\ldots,loc_{t-1},loc_{t+1},\ldots,loc_{t+c}|loc_t)]$$
$$=\frac{1}{T}\sum_{t=1}^{T}\sum_{con}\log p(loc_{t+j}|loc_t)$$
$$where\ con: -c\leq j\leq c, j\neq 0 \tag{8}$$

$$x_t=\tanh([W_\Phi\Phi_t+b_\Phi]\oplus[W_\tau\tau_t+b_\tau]\oplus[W_\zeta\zeta_t+b_\zeta]) \tag{9}$$

Here, $\oplus$ denotes the concatenate operation. Using the embedding vectors, we deploy a bidirectional LSTM layer by adding two gates as follows:

$$TiG_t = sigmoid(x_t\mathbf{W}_{xt}+sigmoid(\Delta t\mathbf{W}_{tiG})+b_G);$$
$$s.t.\mathbf{W}_{tiG}\leq 0$$
$$TD_t = sigmoid(x_t\mathbf{W}_{xd}+sigmoid(\Delta d\mathbf{W}_{tD})+b_D);$$
$$s.t.\mathbf{W}_{tD}\leq 0 \tag{10}$$

In this context, the time and distance intervals are represented by $\Delta t$ and $\Delta d$, respectively. The conditions $\mathbf{W}_{tiG}\leq 0$ and $\mathbf{W}_{tD}\leq 0$ indicate that the influence is stronger when the time and distance intervals are relatively small. In our study, the network is trained jointly to minimize the cross-entropy loss between the predicted label and the ground truth label for both POI class and trip-purpose category.

### B. Model II: Transferring Mobility Knowledge

Transferring mobility knowledge from one region to another geographically dispersed region is a promising area [17], [18] as obtaining labeled mobility data is challenging. Table I represents different data modalities and learning task in the target domain. Note that although the task is similar (trip-annotation and POI identification) but the ROIs (IITKGP, NITW) are different. The mobility trace representation needs to be learnt followed by domain adaptation. The aggregated movement pattern over the underlying road network is initially mapped to find out the footprint deviation in different temporal scales in varied POIs.

The moving agents (people or vehicle) depart from a region (POI), arrive in the destination (POI) and spend time at stay-points. These trips comprise the overall aggregated movement flow of the complete study region (See "flow" mobility fact of MKG and embeddings in Section IV-B2). This mobility flow represent the semantics of the region and help to characterize different POIs in a region. Our proposed model utilizes *Transductive Transfer Learning* [32] to classify the POIs and label the trips of the users of the target domain.

A domain or Region of Interest (ROI) is defined by two primary elements: (a) the feature space ($\xi$), and (b) the marginal probability distribution $P(X)$, where $X = x_1, x_2, \ldots, x_n \in \xi$ [3].

In our set-up, we assume that the ROIs (IITKGP, NITW, UoM, MDC and GeoLife) have the same label space (the POI-tags and trip-purposes are same), but with different marginal probability distribution (spatial distribution of POIs, footprint density and temporal information are different).

The concept of adapting feature spaces from a source domain to a target Region of Interest (ROI) is known as *feature representation transfer*. This process involves acquiring spatio-temporal feature representations from both the source and target domains through deep learning frameworks, followed by dimensionality reduction via a multilayer perceptron. The subsequent phase focuses on knowledge transfer of instances by addressing the challenge of empirical risk minimization, formulated as:

$$\kappa^* = \arg\min_{\kappa} \sum_{(x,y)\in TG} P(G(TG)) \cdot lf(x, y, \kappa) \tag{11}$$

where $P(G(SG))$ represents the marginal probability distribution of the source region's aggregated GPS data, $lf(x, y, \kappa)$ is the loss function, and $\kappa$ denotes the optimal set of parameters in the learning framework. Considering the disparity between the target and source regions, thereby $P(G(SG)) \neq P(G(TG))$, the optimization equation is modified as:

$$\kappa^* = \arg\min_{\kappa} \sum_{(x,y)\in SG} \frac{P(G(TG))}{P(G(SG))} P(G(SG)) \cdot lf(x, y, \kappa) \tag{12}$$

To achieve this objective, it is necessary to estimate $\frac{P(x_{SG_i})}{P(x_{TG_i})}$ for each case. Initially, the Mobilytics framework employs a *domain adaptation* strategy due to the differing distributions of source and target data instances. The core idea is to leverage the labeled data from the source ROI to classify unlabeled trajectory data in the target ROI. Mobilytics proposes a reinforcement learning (RL)-based transfer learning method, emphasizing on instance weighting and adaptation through the learning of rewards and policies. Here, an *agent* aims to predict transitions between stay-points and the duration of stays at different POIs, with *actions* defined by a user's visits and durations at POIs. The *environment* is comprised of the users and POIs within the Mobility Knowledge Graph (MKG), denoted as $\langle MKG, u_i\rangle$. The *reward* is determined by the spatial and temporal accuracy between actual and predicted stay-point transitions. For the learning of policies, a variant of the Deep Q-Network and a potential-based reward shaping technique are employed, facilitating the application of transfer learning on the MKG. The key components of the framework are described as follows:

*Agent:* Within Mobilytics, the agent is conceptualized as the entity responsible for forecasting or planning the subsequent movement of a user. Based on the user's current location and the surrounding environment as inputs, the agent is tasked with determining the forthcoming transition, encompassing both the travel distance necessary to arrive at the next point of stay and the duration of time to be spent there.

*Actions ($\alpha$):* The framework delineates actions in a dual capacity: (i) $\alpha = (1, p_a, i)$ signifies a user's visit to point of interest $p_a$ after covering a distance $i$, and (ii) $\alpha = (0, p_a, t)$ represents a user's duration $t$ of stay at $p_a$. The domain of actions is constituted by the set of POIs, with the initial component of $\alpha$ indicating the choice between transitioning to or remaining at a POI.

*Environment ($En$) and State ($se$):* The system's environment is structured as $En = (MKG, U)$, encompassing the mobility knowledge graph and the user community within the specified region, including their mobility patterns. The interplay between user behaviors of visiting and staying and the mobility knowledge graph's structure influences both entities reciprocally. The state captures the mobility path of an individual user, encapsulating the count of visited and stayed POIs, the duration of stays, and the distances traversed between POIs.

*Reward ($Rd$):* Central to the reinforcement learning paradigm, the reward function guides the optimization process. In this framework, the reward is calculated as a weighted aggregate of several criteria: (i) $dtra$, the inverse of the distance between actual and forecasted POI visits; (ii) $durS$, the inverse of the discrepancy between actual and predicted stay durations at a POI; and (iii) $act$, reflecting the accuracy of predicting whether the user's action pertains to staying or transitioning at a POI. Assuming the trajectory comprises of $n$ stay-points, the reward for the complete trajectory trace is computed as:

$$Rd = wr_1 \times \sum_{j=1}^{(n-1)} dtra_j + wr_2 \times \sum_{j=1}^{(n)} durS_j + wr_3 \times \sum_{j=1}^{(n)} act \tag{13}$$

The reward function is designed to evaluate the predictions made by the agent regarding the user's mobility trace and is a weighted sum of three components: (1) $wr_1$ is the weight applied to $dtra_j$, which is the reciprocal of the distance traveled between the actual and predicted Point of Interest (POI) visit. This term emphasizes the accuracy of the location predictions. A higher weight for $wr_1$ will cause the agent to prioritize minimizing the distance error in its predictions, which could lead to more precise POI visit predictions. (2) $wr_2$ weights $durS_j$, the reciprocal of the difference in time duration spent at the POI between the real and predicted values. This weight controls how much the agent focuses on accurately predicting the time a user spends at a POI. If $wr_2$ is increased, the agent is incentivized to improve the accuracy of predicting the duration of stays at POIs. (3) $wr_3$ is the weight for the action accuracy component $act$, which represents whether the action of staying at or transitioning to a POI is correctly predicted. By adjusting $wr_3$, the reinforcement learning algorithm can be fine-tuned to either reward the prediction of user actions more or less, depending on which aspects of the user's mobility behavior are more critical to the application at hand.

Given the present state ($se$) of the user, the goal of the agent is to maximize the reward by correctly predicting the transitions and stay-points of the user trajectory trace. *Policy* is the core of the agent, which learns the mapping from state to an action from the available data-instances. As mentioned earlier, Deep Q-network is used for policy learning. Here, we propose two-phase learning simultaneously in a feedback loop: in the first phase, the agent attempts to learn the mapping from the source domain training dataset; in the second phase, the available historical records of the mobility data of target domain is used to refine the policy ($\pi$).

Now, we have to find the policy ($\pi$) in a way:

$$\begin{aligned} \pi &= \alpha_t \ such\ that\ maximize\ Rd_t \\ &= \sum_{t=0} \gamma^t rd_t;\ where\ 0 \leq \gamma < 1 \end{aligned} \tag{14}$$

where $\gamma$ is the discount factor. The deployed deep neural network approximates the optimal action-value function $Q^*(se, \alpha)$ as:

$$\begin{aligned} Q^*(se, \alpha) &= \max_{\pi} \mathbb{E}[Rd_t | se_t = se, \alpha_t = \alpha, \pi] \\ &= \max_{\pi} \mathbb{E}[rd_t + \gamma rd_{t+1} + \gamma^2 rd_{t+2} \\ &\quad + \ldots | se_t = se, \alpha_t = \alpha, \pi] \end{aligned} \tag{15}$$

After an observation $se$ and action $\alpha$, the maximum sum of rewards $rd_t$ at each time-step $t$ can be achieved by a policy $\pi = P(\alpha | se)$. Based on the well-known Bellman equation, we can re-write the optimal action-value function as:

$$Q^*(se, \alpha) = \max_{\pi} \mathbb{E}_{se'}[Rd + \gamma \max_{\alpha'} Q^*(se', \alpha') | se, \alpha] \tag{16}$$

The intuitive idea of any reinforcement learning algorithm is to find out the action-value function by *value iteration* method $Q_{j+1}(se, \alpha) = \mathbb{E}_{se'}[Rd + \gamma \max_{\alpha'} Q_i(se', \alpha') | se, \alpha]$. However, the basic approach is unrealistic as it converges at $i \rightarrow \infty$. Therefore, a function approximator is needed. We deploy a neural network such that $Q(se, \alpha; \omega) \approx Q * (se, \alpha)$, where $\omega$ denotes the weights, and the network is trained by the loss function $RLoss_i(\omega_i)$ at each iteration $i$ as:

$$\begin{aligned} RLoss_i(\omega_i) = \mathbb{E}_{se, \alpha \sim P(SG(.))}[(Rd + \gamma \max_{\alpha'} Q(se', \alpha'; \omega_{i-1}) \\ - Q(se, \alpha; \omega_i)^2] \end{aligned} \tag{17}$$

Since the source data instances are drawn as training samples, the gradient of the loss function is computed as:

$$\begin{aligned} \bigtriangledown_{\omega_i} RLoss_i(\omega_i) = \mathbb{E}(Rd + \gamma \max_{\alpha'} Q(se', \alpha'; \omega_{i-1}) \\ - Q(se, \alpha; \omega_i) \bigtriangledown_{\omega_i} Q(se, \alpha; \omega_i)] \end{aligned} \tag{18}$$

In the subsequent phase, our focus is to leverage *relational knowledge*. Given that the mobility knowledge graph (MKG) outlines the interconnections between mobility flows and Points of Interest (POIs) within a specific area, we leverage this framework to facilitate the transfer of relational insights from the source area to the target area, aiming to augment incomplete labels within the MKG of the target region. Drawing inspiration from the methodology proposed by Mihalkova et al. [33], which employs Markov logic networks [34] for mapping relational dynamics from one domain to another, our approach diverges by utilizing the inherent mobility relationships or facts encoded within the MKG, as opposed to relying on Markov logic networks.

To summarize, our approach employs Transductive Transfer Learning to assign labels to trajectory segments and Points of Interest (POIs) in the target domain based on the learned representation. For domain adaptation, we utilize a Reinforcement Learning (RL) agent to capture the movement behavior of users in the target region by leveraging knowledge from the source region. While prior works have made various attempts at transfer learning, the majority of them have focused on text or image classification. Our work, on the other hand, aims to predict missing labels and mobility behaviour relationships.

## VI. Performance Evaluation

This section presents the efficacy of the various components of the proposed architecture. Through comprehensive experimental analysis on real-world datasets, Mobilytics's performance is evaluated and benchmarked against a range of foundational methods across diverse scenarios. The implementation of these modules was carried out on a deep learning virtual machine (VM) configuration provided by *Google Cloud Platform (GCP)*, equipped with $2 \times NVIDIA\, Tesla\, T4\, GPUs$ and $2\, vCPUs\ +\ 13\, GB\, memory\, (n1 - highmem - 2)$. The development was undertaken using Python, with TensorFlow Enterprise 2.3 (CUDA 11.0) serving as the backbone for the deep learning framework.

### A. Datasets

We have used five trajectory datasets from different regions for validating Mobilytics.

- Indian dataset: The datasets are collected from residents of two indian academic campuses: Indian Institute of

Technology, Kharagpur (IITKGP) and National Institute of Technology, Warangal (NITW) where GPS traces of 145 and 72 volunteers are collected for 28 months. Total 56 such labels (*trip-purpose* or *activity*) are listed in the web-based survey, and the participants select the appropriate labels from the list. The mobility datasets contain continuous GPS logs of individuals for the studied time-period. Each individual's daily GPS log consists of average 11 trajectory-segments and 85% of the log capture mobility data in a very high-sampling rate of 60 secs. The POIs of these two campuses are initially mapped from Google Place API. Further, three residents of the campus refined the POI-tag database manually. It is discovered that a few buildings are utilized for both academic and administration purposes that are not captured by Google Place API. This process enriched POI-tags and built a more accurate POI-taxonomy. To the best of our knowledge, huge amount of labelled GPS traces are not available. There exist a few publicly available GPS trace datasets (e.g., *GeoLife*, *TDrive* [1]); however they do not contain any semantic information. We aim to collect and prepare the labeled GPS traces in this work.[4]

- Campus dataset (University of Melbourne): GPS logs collected from 25 users for 3 months.
- GeoLife dataset [1]: GPS trajectory of 182 users for one year in Beijing.
- Nokia MDC dataset [35]: GPS traces of 200 individuals for 9 months in the Lake Geneva region, Switzerland region.

### B. Semantic Mobility Knowledge Extraction

The Adam algorithm updates the network weights iteratively to optimize the parameters using cell size 64 and batch size 10.

Since the datasets have the "label" or "trip-purpose" information, we can testify whether Mobilytics can predict the trip-purposes accurately by comparing the output with ground-truth data. To the best of our knowledge, the literature presents a limited number of studies focused exclusively on deriving trip purposes from individuals' mobility logs. Nonetheless, our analysis includes a comparison of Mobilytics against the most closely related existing contributionsIn the implementation of baselines on next location prediction, we have slightly modified the output as next trip-purpose detection. and baselines as follows:

- LDA: Each trajectory segment is considered as a document where each POI is a word. Then, topic model LDA is used to learn the topic distribution of each trajectory segment followed by labelling to a specific class (trip-purpose).
- POI2Vec [36]: Mapping geographical and temporal impacts using POI embedding-based method to learn POI representations.
- Flashback [37]: RNN-based mobility model to search the periodic movement patterns from historical data followed by matching in the hidden states.

[4]Codebase and sample data available: https://drive.google.com/drive/folders/1BpM-K3clH6XYpSHkFe12aGsG8n1AclI4?usp=sharing

TABLE III
DIFFERENT IDENTIFIED POI-TYPES IN NITW CAMPUS AND NEIGHBOURING AREAS AFTER TRANSFER LEARNING (TC: TOTAL COUNT, NC: NUMBER OF CORRECTLY IDENTIFIED POI)

| POI-type | TC | NC | POI-type | TC | NC |
|---|---|---|---|---|---|
| Academic Building | 18 | 16 | Hospital | 1 | 1 |
| Student Hall | 23 | 21 | ATM | 3 | 2 |
| Residence (Staff) | 3 | 1 | Bank | 2 | 2 |
| Student Canteen | 6 | 5 | Guest House | 1 | 0 |
| Auditorium | 3 | 1 | Library | 1 | 1 |
| Department | 11 | 9 | Sports Complex | 2 | 1 |
| Restaurant | 6 | 5 | Medical Store/ Center | 3 | 2 |
| Cafeteria | 3 | 2 | Post-office | 1 | 1 |
| Parking Area | 25 | 23 | Shopping Complex | 3 | 3 |

- VANext [38]: Exploits individual's periodical mobility and recent movement paths using variational attention and a CNN to encode users' moving patterns.
- DeepMove [39]: Utilizes RNN and attention mechanism to encode human movement dynamics.
- Soares et al. [40]: Novel KDD method to detect the travel mode and predict the purpose (*home, work, education, shopping, leisure, other*) of a trip.
- STR [41]: Spatio-temporal regularity based model using Markov random field to find the best annotations maximizing the consistency of annotated trips.
- MoveSim [42]: Self-attention based sequential modeling network for encoding the temporal transitions in human movement patterns exploiting the prior knowledge and generative adversarial learning framework for pre-training.
- SML-TUL [43]: Self-supervised mobility learning framework to characterize the inherent movement correlations and classify trajectories.
- NCF [44]: Annotate the POIs associated with raw user-generated mobility records using neural context fusion approach considering POI-visiting behaviors and representation learning.
- Wheels [45]: Trip purpose prediction (9 categories) using vehicle GPS traces, public POI check-in data.

Evaluations of these methodologies were conducted utilizing popular metrics: $ACC@1$, $ACC@5$, macro-P, macro-R, and macro-F1. The performance outcomes of Mobilytics alongside other techniques are consolidated in Table IV. Among the contenders, NCF, which employs a neural context fusion and attention mechanism, alongside Wheel, demonstrate superior performance across various test cases. Yet, Mobilytics surpasses NCF by a notable margin of approximately 15%. It is also observed that NCF's performance drops to roughly 61% when trained with only 10% of the data, indicating its limitations under data scarcity. Despite the limited POI categories considered by the authors [44], their evaluation on public datasets from NYC and Beijing showcased a minimum of 32% higher accuracy over existing methods, underscoring the potential scalability of our framework. Additionally, SML-TUL, which leverages trajectory augmentation and a self-supervised mobility learning framework for trajectory classification, exhibited commendable performance. An ablation study was conducted to dissect the contributions of specific components within Mobilytics, such as MKG and Transfer Learning, revealing an enhancement in

TABLE IV
COMPARATIVE ANALYSIS OF SEMANTIC LABEL (TRIP-PURPOSE) ANNOTATION

| Method | IITKGP Dataset | | | | | NITW Dataset | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $ACC@1$ | $ACC@5$ | macro-P | macro-R | macro-F1 | $ACC@1$ | $ACC@5$ | macro-P | macro-R | macro-F1 |
| LDA | 62.91% | 73.53% | 61.77% | 59.45% | 60.52% | 61.37% | 72.08% | 60.55% | 58.48% | 59.25% |
| POI2Vec | 65.53% | 76.29% | 63.22% | 62.66% | 62.82% | 64.19% | 75.71% | 62.15% | 61.85% | 62.09% |
| Flashback | 66.21% | 77.81% | 64.90% | 64.12% | 64.55% | 65.67% | 77.13% | 63.21% | 63.14% | 63.46% |
| VANext | 68.78% | 80.02% | 66.72% | 65.30% | 66.21% | 67.12% | 79.82% | 65.24% | 64.11% | 64.70% |
| DeepMov | 66.98% | 78.20% | 65.12% | 64.90% | 64.77% | 63.08% | 74.86% | 64.27% | 62.16% | 63.19% |
| Soares et al. [40] | 66.48% | 79.91% | 65.02% | 63.87% | 64.43% | 66.48% | 78.25% | 61.99% | 60.02% | 60.98% |
| STR | 72.02% | 83.99% | 71.42% | 69.78% | 70.81% | 71.42% | 82.57% | 67.95% | 66.51% | 67.27% |
| MoveSim | 69.12% | 81.62% | 68.55% | 67.06% | 67.82% | 65.70% | 79.31% | 63.21% | 62.55% | 62.86% |
| SML-TUL | 72.88% | 85.14% | 71.45% | 69.70% | 70.46% | 71.22% | 82.62% | 68.53% | 67.41% | 67.85% |
| NCF | 73.54% | 84.29% | 71.60% | 70.21% | 70.29% | 72.86% | 84.15% | 69.57% | 68.12% | 68.53% |
| Wheel | 71.02% | 84.61% | 72.10% | 71.08% | 71.58% | 73.42% | 83.18% | 70.06% | 69.11% | 69.58% |
| **Mobilytics (w/o MKG)** | 84.06% | 91.01% | 76.56% | 79.20% | 77.85% | 80.62% | 88.71% | 76.12% | 78.02% | 77.05% |
| **Mobilytics (w/o MKG)** | 76.18% | 74.02% | 73.08% | 74.16% | 73.61% | 77.01% | 82.06% | 72.09% | 74.18% | 73.12% |
| (limited data at Target) | Transferred knowledge from NITW to IITKGP | | | | | Transferred knowledge from IITKGP to NITW | | | | |
| **Mobilytics (Full)** | 87.02% | 92.51% | 84.24% | 81.08% | 82.62% | 84.18% | 90.05% | 82.86% | 81.21% | 82.02% |
| **Mobilytics (Full)** | 83.20% | 87.18% | 80.61% | 78.11% | 79.34% | 83.66% | 88.10% | 80.21% | 80.52% | 80.36% |
| (limited data at Target) | Transferred knowledge from NITW to IITKGP | | | | | Transferred knowledge from IITKGP to NITW | | | | |
| Learning cost: ∼6 hr (IITKGP data), ∼2 hr (NITW data); Model size: 6.2 MB w/o Transfer learning module, 8.6 MB w/ Transfer learning model | | | | | | | | | | |

The evaluation encompasses all baseline methods using the full dataset. In contrast, the scenario labeled as limited data at target refers to the training process conducted with only 10% of the data accessible in the target region.

TABLE V
POI ANNOTATIONS COMPARISONS USING TRANSFER LEARNING FOR ALL FIVE REGIONS OF INTEREST

| Target domain | Source domain | ACC@5 | macro-F1 |
|---|---|---|---|
| NITW | IIT KGP | 88.08% | 78.02% |
| IIT KGP | GeoLife+NITW | 91.06% | 84.11% |
| Tsinghua | IIT KGP+UoM | 89.06% | 85.01% |
| UoM | IIT KGP | 85.81% | 80.05% |
| MDC | UoM | 83.21% | 76.02% |

TABLE VI
COMPARISON OF POI-CLASSIFICATION

| **Approach** | **Accuracy** |
|---|---|
| Transfer Learning with Naive Bayes | 51.05% |
| Transfer Learning with Hierarchical Bayesian network | 60.58% |
| *Transfer Learning in Mobilytics (Proposed)* | *83.58%* |

TABLE VII
COMPARISON OF POI-CLASSIFICATION (A: 72 PARTICIPANTS' LABELLED TRACE, B: 18 PARTICIPANTS' LABELLED TRACE)

| **Classifier** | **kNN** | **Naive Bayes** | **Decision Tree** | **SVM** | **BP** | **P-F** |
|---|---|---|---|---|---|---|
| Accuracy (A) (%) | 81.56 | 80.08 | 84.5 | 88.02 | 84.86 | **92.51** |
| Accuracy (B) (%) | 54.02 | 42.3 | 44.61 | 38.73 | 34.2 | **83.58** |

performance margins ranging from approximately 18% to 30% over the baseline methods.

We also carried out a comprehensive analysis on selection of domains. For all five different domains (different geographical regions), we evaluated for all 14 combinations of source domain selection. Table V represents the best scores amongst all source domain combinations. It is interesting to note that the integration of two domains produced better accuracy for a few of the cases.

To depict the efficacy of the transfer learning module, we carry out the POI-classification task with only 18 labelled GPS traces of NITW and using the proposed approach to identify and classify POIs by utilizing the mobility semantics of IITKGP campus. Table III shows the count of correctly identified POIs in NITW campus. The accuracy of the POI-identification is almost $83.58\%$ on average. To depict the significance of the proposed transfer learning technique, we implement five well-known classifiers namely *K-nearest neighbour(kNN)*, *naive Bayes*, *decision tree*, *support vector machine (SVM)* and *backward propagation neural network (BP)*.

To evaluate the classifiers, we used as features: (i) the number of trips starting in each 30mins ($n_1$) averaged over days. It is a vector of 96 dimensions, with 48 dimensions for weekdays and 48 dimensions for weekends (or holidays); (ii) the number of trips set-down (end) in each 30mins ($n_2$) averaged over days: It is a vector of 96 dimensions, where 48 dimensions for weekdays and 48 dimensions for weekends (or holidays); (iii) the ratio of starting trip and set-up trips in each 30mins ($n_3$); and (iv) the average stay-duration ($n_4$) in each day of a week.

The SVM classifier uses Gaussian kernel function. Since the problem is a multi-class classification, $m(m-2)/2$ binary classifiers are trained, where $m$ is the number of classes. The final classification result is obtained by voting through all binary classifiers, and the class with majority-vote is selected. We have implemented three-layer BP network.

To measure the dissimilarities, all samples (POIs) are represented by the vector input, each having four attributes as mentioned above. Then, attribute-wise euclidean distance is computed and the class label assigned to a test example is determined by the majority vote of its k nearest neighbors. The weights of all attributes are kept same. The k-values are checked from $1-10$, where the accuracy is maximized in $k=4$. It may be noted that all parameters of the experiments are optimized. The toolkit *Scikit-learn 0.19.2* of python is used to implement all these classical classifiers.

Table VII shows the POI-classification accuracy of our proposed framework (P-F) with five classifiers. The experiment is


(a) (b) (c)

Fig. 5. Sample illustration of Trajectory-traces and obtained clusters based on movement behaviors (a) Sample. json file (b) Trajectory traces of 10 users in 1000-2000 hrs of a day (c) Three similar movement behaviours shown in different colors.

carried out in two set-ups: (i) with all labelled GPS traces of 72 participants and (ii) with only 18 participants. It is observed that all other classifiers perform poorly with fewer labelled traces, however with the transfer learning module our proposed framework achieves more than 27% accuracy measure than all other baselines.

We implement other transfer learning approach namely, *transfer learning using Naive Bayes* and *transfer learning using Hierarchical Bayesian* model to map the mobility knowledge from source to target. The naive Bayes transfer learning (NBTL) classification algorithm for text categorization is presented in [46]. Raykar et al. [47] present the Bayesian multiple instance learning (MIL) algorithm, which is capable of feature selection and classifier construction parallelly. Table VI shows the accuracy measure of Mobilytics compared to others. It is observed that our transfer learning set-up has outperformed these two baselines. The key reason is the assumption of feature independence of *naive Bayes* does not hold for semantic label classification. On the other side, our proposed method achieves better result. The deep architecture has helped to produce the feature representation and map the knowledge more effectively.

Another interesting finding is the classification of a few POIs into multiple labels. For example, a few multi-storey buildings are utilized as *administrative work* and *lecture hall*, or *auditorium* and *administrative office*. The reverse geo-coding process using Google Place API fails to detect such cases. This issue is solved by Mobilytics framework, since it does not rely on the crowd-sourced data, rather it extracts underlying semantics of the place and classify them. Mobilytics was capable of extracting 12 such places in IITKGP and 5 places in NITW campus. Hence Mobilytics can also be used in activity-type annotations in building information modeling to predict activity-spots of multi-storey building.

### C. Visualization

We illustrate some real samples of the dataset and experimental results here. Fig. 5(a) represents a sample $.json$ file extracted from Google Map Timeline, where timestamp, latitude, longitude and accuracy value are shown. This sensor accuracy value is used in the Kalman filtering process. Fig. 5(b) shows trajectory traces of 10 users in a typical weekday in the time-interval 1000 - 2000 hrs. The trajectory traces of the individuals are converted into heatmap representation on Google Map surface. At a particular time-instance, the overlapping locations create the higher density and give each area a color value. It is used to depict the intensity (footprint density) of the location sequences, where areas of higher intensity are shown in red colour, and areas of lower intensity appear in green. The obtained clusters from these traces are shown in Fig. 5(c). These clusters represent similar moving behaviour, where three colors represent three different movement behaviours, such as, red-colored traces represent commuting to lecture-hall, blue-colored traces represent commuting to market and green-colored traces represent commuting to cafeteria. It may be noted that the trajectories span in different spatial and temporal scales, however our proposed algorithm is able to group the traces based on the trip-purposes. There are total 56 semantic labels of trajectory trips in the dataset and we have listed top 20 labels of the mobility traces and their respective counts in Appendix.

### D. Discussions and Simulation Study

This work aims to extract mobility semantics from the movement log of individuals. There are several challenges to capture this movement semantics. The proposed framework, Mobilytics, deals with these issues deploying a deep learning architecture, and is able to effectively extract the movement semantics for automatic annotations of trajectories. (refer Table IV, Sections V-A and V-B) The experimental result depicts promising accuracy and precision measures for identifying the movement semantics of users.

The key challenges to extract mobility semantics are two-folds: (i) first, it is not feasible to use conventional statistical analysis to model travel behaviours as this analysis may fall short to extract underlying complex dynamics of the mobility features.

(ii) And without labelled training data, it is not possible to carry out any supervised learning task at a new region. Mobilytics has shown that adapting a proper transfer learning technique can provision the mobility knowledge transfer to a geographically dispersed region. The proposed framework achieved promising accuracy for classifying POIs in the target region with very few labelled data compared to the baseline methods. [Refer Tables V–VII]

We collected real datasets consisting of 145 volunteers from IIT Kgp and 72 volunteers from NITW. However, due to privacy concerns, it is difficult to obtain large-scale mobility traces of individuals. To depict the scalability of the proposed framework, we have simulated a huge amount of mobility traces using MATSim (https://www.matsim.org/) simulator. The major observations are as follows: Mobilytics framework is scalable – the performance does not degrade with increase in data-load (simulated carried out up to 10 K users). It has been observed that there is an improvement of accuracy in semantic label annotation with more training data samples. It proves that the deep learning architecture and the transfer learning modules are capable to fine-tune the parameters when more training data is available. The precision and recall measures of annotation of few semantic labels (namely, $L3$, $L5$, $L7$, $L10$, $L16$ and $L17$) show better result with synthetic dataset. These are mostly the group-activities of users. (Detailed discussion in Appendix.)

## VII. Conclusion

This paper represents a conceptual framework *Mobilytics* conducive to extract mobility semantics of people and classify several points of interest (POIs). The contributions are manifold. First, the mobility knowledge graph (MKG) captures the underlying semantic correlations of movement patterns in different spatial and temporal scales. Our paper is the first effort to apply mobility knowledge transfer across geographically distinct regions for the purpose of classifying Points of Interest (POIs) in the context of scarce labeled data. Second, a deep architecture is deployed to represent users' mobility behaviours and extract the trip-purposes or movement semantics. Further, Mobilytics has outperformed the baseline methods in terms of *recall*, *accuracy*, and *precision*. The POI identification results have also shown promising outcomes. Third, through different domain selections, Mobilytics can be extended to a city region for annotating individuals' trips and POI classification. As future work, the proposed transfer learning module can be extended to contextual information such as Call Data Record and social network traces across geographical regions. At present, our work is limited to academic campus, and we plan to deploy the framework across cities and enhance the present architecture effectively leveraging heterogeneous data sources.

## References


[1] Y. Zheng, "Trajectory data mining: An overview," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 3, Art. no. 29, 2015.

[2] S. Ghosh and S. K. Ghosh, "Why did you go there? Semantic knowledge extraction from trajectory traces," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2023, pp. 6013–6016.

[3] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[4] P. Wang et al., "Incremental mobile user profiling: Reinforcement learning with spatial knowledge graph for modeling event streams," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 853–861.

[5] B. Du, C. Liu, W. Zhou, Z. Hou, and H. Xiong, "Detecting pickpocket suspects from large-scale public transit records," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 3, pp. 465–478, Mar. 2019.

[6] C. Renso et al., "How you move reveals who you are: Understanding human behavior by analyzing trajectory data," *Knowl. Inf. Syst.*, vol. 37, no. 2, pp. 331–362, 2013.

[7] D.-W. Choi, J. Pei, and T. Heinis, "Efficient mining of regional movement patterns in semantic trajectories," in *Proc. VLDB Endowment*, vol. 10, no. 13, pp. 2073–2084, 2017.

[8] P. Wang et al., "Learning urban community structures: A collective embedding perspective with periodic spatial-temporal mobility graphs," *ACM Trans. Intell. Syst. Technol.*, vol. 9, pp. 1–28, 2018.

[9] G. Giannopoulos and M. Meimaris, "Learning domain driven and semantically enriched embeddings for POI classification," in *Proc. 16th Int. Symp. Spatial Temporal Databases*, 2019, pp. 214–217.

[10] M. Wang et al., "Frost: Movement history–conscious facility relocation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, pp. 1–26, 2020.

[11] Y. Lin et al., "Learning entity and relation embeddings for knowledge graph completion," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 2181–2187.

[12] P. Wang, Y. Fu, H. Xiong, and X. Li, "Adversarial substructured representation learning for mobile user profiling," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 130–138.

[13] P. Kothari, S. Kreiss, and A. Alahi, "Human trajectory forecasting in crowds: A deep learning perspective," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7386–7400, Jul. 2022.

[14] M. Yue et al., "DETECT: Deep trajectory clustering for mobility-behavior analysis," in *Proc. IEEE Int. Conf. Big Data*, 2019, pp. 988–997.

[15] H. Cao, F. Xu, J. Sankaranarayanan, Y. Li, and H. Samet, "Habit2vec: Trajectory semantic embedding for living pattern recognition in population," *IEEE Trans. Mobile Comput.*, vol. 19, no. 5, pp. 1096–1108, May 2020.

[16] S. Dabiri, C.-T. Lu, K. Heaslip, and C. K. Reddy, "Semi-supervised deep learning approach for transportation mode identification using GPS trajectory data," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 5, pp. 1010–1023, May 2020.

[17] Q. Wu, K. He, X. Chen, S. Yu, and J. Zhang, "Deep transfer learning across cities for mobile traffic prediction," *IEEE/ACM Trans. Netw.*, vol. 30, no. 3, pp. 1255–1267, Jun. 2022.

[18] T. He et al., "What is the human mobility in a new city: Transfer mobility knowledge across cities," in *Proc. Web Conf.*, 2020, pp. 1355–1365.

[19] M. Katranji, E. Thuillier, S. Kraiem, L. Moalic, and F. H. Selem, "Mobility data disaggregation: A transfer learning approach," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst.*, 2016, pp. 1672–1677.

[20] Z. Liu, Y. Shen, and Y. Zhu, "Where will dockless shared bikes be stacked? —Parking hotspots detection in a new city," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 566–575.

[21] J. Ding et al., "Learning from hometown and current city: Cross-city POI recommendation via interest drift and transfer learning," in *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 3, no. 4, pp. 1–28, 2019.

[22] X. Wang, M. Fang, Z. Zeng, and T. Cheng, "Where would I go next? Large language models as human mobility predictors," 2023, *arXiv:2308.15197*.

[23] R. Manvi et al., "GeoLLM: Extracting geospatial knowledge from large language models," 2023, *arXiv:2310.06213*.

[24] S. Shekhar and S. Chawla, *Spatial Databases: A Tour*. London, England: Pearson, 2003.

[25] W.-C. Lee and J. Krumm, "Trajectory preprocessing," in *Computing with Spatial Trajectories*. Berlin, Germany: Springer, 2011, pp. 3–33.

[26] Y. -J. Gong, E. Chen, X. Zhang, L. M. Ni, and J. Zhang, "AntMapper: An ant colony-based map matching approach for trajectory-based applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 390–401, Feb. 2018.

[27] S. Ghosh, A. Mukherjee, S. K. Ghosh, and R. Buyya, "Mobi-IoST: Mobility-aware cloud-fog-edge-IoT collaborative framework for time-critical applications," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2271–2285, Oct.-Dec. 2020.

[28] H. Wang et al., "Spatio-temporal urban knowledge graph enabled mobility prediction," in *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 4, pp. 1–24, 2021.

[29] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "Rotate: Knowledge graph embedding by relational rotation in complex space," 2019, *arXiv: 1902.10197*.

[30] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 12, pp. 2724–2743, Dec. 2017.
[31] N. B. Toomarian and J. Barhen, "Learning a trajectory using adjoint functions and teacher forcing," *Neural Netw.*, vol. 5, no. 3, pp. 473–484, 1992.
[32] A. Moreo, A. Esuli, and F. Sebastiani, "Lost in transduction: Transductive transfer learning in text classification," *ACM Trans. Knowl. Discov. Data*, vol. 16, no. 1, pp. 1–21, 2021.
[33] L. Mihalkova, T. Huynh, and R. J. Mooney, "Mapping and revising Markov logic networks for transfer learning," in *Proc. AAAI Conf. Artif. Intell.*, 2007, pp. 608–614.
[34] M. Richardson and P. Domingos, "Markov logic networks," *Mach. Learn.*, vol. 62, no. 1/2, pp. 107–136, 2006.
[35] Y. Z. Y. S. Y. Wang, "Nokia mobile data challenge: Predicting semantic place and next place via mobile data," *Work*, vol. 80, no. 100, 2012, Art. no. 120.
[36] S. Feng et al., "POI2Vec: Geographical latent representation for predicting future visitors," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 102–108.
[37] D. Yang, B. Fankhauser, P. Rosso, and P. Cudre-Mauroux, "Location prediction over sparse user mobility traces using RNNs: Flashback in hidden states!," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, 2020, pp. 2184–2190.
[38] Q. Gao et al., "Predicting human mobility via variational attention," in *Proc. World Wide Web Conf.*, 2019, pp. 2750–2756.
[39] J. Feng et al., "DeepMove: Predicting human mobility with attentional recurrent networks," in *Proc. World Wide Web Conf.*, 2018, pp. 1459–1468.
[40] E. F. de S. Soares, K. Revoredo, F. Baião, C. A. de M. S. Quintella, and C. A. V. Campos, "A combined solution for real-time travel mode detection and trip purpose prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4655–4664, Dec. 2019.
[41] F. Wu and Z. Li, "Where did you go: Personalized annotation of mobility records," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, 2016, pp. 589–598.
[42] J. Feng et al., "Learning to simulate human mobility," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 3426–3433.
[43] F. Zhou et al., "Self-supervised human mobility learning for next location prediction and trajectory classification," *Knowl.-Based Syst.*, vol. 228, 2021, Art. no. 107214.
[44] R. Hu, J. Zhou, X. Lu, H. Zhu, S. Ma, and H. Xiong, "NCF: A neural context fusion approach to raw mobility annotation," *IEEE Trans. Mobile Comput.*, vol. 21, no. 1, pp. 226–238, Jan. 2022.
[45] C. Liao et al., "Wheels know why you travel: Predicting trip purpose via a dual-attention graph embedding network," in *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 1, pp. 1–22, 2022.
[46] S. Tan, X. Cheng, Y. Wang, and H. Xu, "Adapting naive bayes to domain adaptation for sentiment analysis," in *Proc. Eur. Conf. Inf. Retrieval*, 2009, pp. 337–349.
[47] V. C. Raykar et al., "Bayesian multiple instance learning: Automatic feature selection and inductive transfer," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 808–815.

**Shreya Ghosh** (Member, IEEE) is an assistant professor with the Department of Computer Science and Engineering, IIT Bhubaneswar, India. She was a postdoctoral fellow with the Pennsylvania State University, USA during 2021-2023. Her research interests include NLP, trajectory data mining, and social media data analytics.

**Soumya K. Ghosh** (Senior Member, IEEE) is a professor with the Department of Computer Science and Engineering, IIT Kharagpur, India. He was with the Indian Space Research Organization (ISRO), Bengaluru, India. His research interests include spatial data science, spatial web services, remote sensing, and cloud computing.

**Sajal K Das** (Fellow, IEEE) is a Curators' Distinguished Professor in the Department of Computer Science and the Daniel St. Clair Endowed Chair at Missouri University of Science and Technology, Rolla, MO, USA. His research interests include cyber-physical systems, smart environments, mobile and pervasive computing, and big data analytics.

**Prasenjit Mitra** (Senior Member, IEEE) is a professor with the College of Information Sciences and Technology, the Pennsylvania State University, USA. He is also a visiting professor with L3S Research Center, Leibniz University, Hannover, Germany. His research interests include information retrieval, big data analytics, and machine learning.