From Mobile Data to Business Insights: A Scalable Analytics Platform for Urban Mobility Intelligence

Thiago Andrade 0000-0002-3210-2066[1*], Shazia Tabassum 0000-0003-0782-7054[1], Miguel E. P. Silva 0000-0003-4893-5625[1], Ricardo Dinis[2] and João Gama 0000-0002-3210-2066[1]

[1]LIAAD, INESC-TEC, Porto, Portugal.
[2]Mobile Network Analytics, NOS SGPS, Lisboa, Portugal.

*Corresponding author(s). E-mail(s): thiago.a.silva@inesctec.pt;

1 Abstract

Real-time location data derived from mobile applications is a powerful tool for addressing various urban challenges, including parking management, bus route optimization, tourism planning, and resource allocation. Besides, it offers invaluable insights for shaping strategic decisions in commercial domains such as location-based services, market share analysis, and behavioral profiling. In this expansive study, we aim to address all of the aforementioned challenges by investigating the behaviors and patterns of smartphone users within urban environments, particularly in the domains of tourism, transportation, and retail.

Our approach encompasses the development of a sophisticated data platform from inception to implementation, which includes the formulation of use cases, architectural design, and implementation of modules. We employ state-of-the-art techniques and technologies, including data anonymization, ETL pipelines, and utilizing Google BigQuery and Vertex AI for data processing and machine learning model development. Additionally, we apply interactive data visualization techniques via Power BI to facilitate a comprehensive interpretation of our findings.

The technical contributions of this work are significant, with the development of analytical models tailored to address massive data analysis of user behaviors and spatio-temporal pattern mining. These models cover diverse issues such as

mobility profiling, frequent trajectories, area of influence, anomaly detection, and origin-destination patterns. The results demonstrate a profound understanding of user dynamics at a granular level of both space and time, providing actionable insights for urban planning and business strategic decision-making.

2 Introduction

Mobile communications are an essential pillar of a digital society, with smartphones, in particular, as a mandatory accessory for most of the population, as both their main means of communication and point of access to digital content. Society's reliance on smartphones makes these devices an extension of the self, enabling user profiling from the data associated with the phone and its communications. Mobile phone operators are then in a privileged position to exploit this information through technology embedded in communication networks that fulfil services required by users. Their access to mobile communication data, specifically device location, enables mobile phone operators to capture, store and ubiquitously analyze massive amounts of data to generate insights on user patterns and profiles. These insights have a myriad of applications, from social good applications (in case of natural catastrophe, they can help civil authorities in planning evacuation routes or understanding tourist movement patterns to improve tourism routes and identify points of interest) to being sold as data monetization services (for commerce, banking or insurance companies to understand client movement patterns).

The pervasive reliance of society on mobile phone communications means that the quantity of mobile communications data generated per day is massive, imposing constraints on how it is stored, processed and analyzed as traditional methodologies are ill-equipped to handle such massive datasets. Another important aspect of mobile communication data is its streaming quality. New data is constantly being generated, and analytic models often have to answer questions in real time in this data stream. Over the past decade, there has been a large research effort to handle all facets of the ever-increasing volume of data generated worldwide. However, there is a constant need for novel methodologies for extracting knowledge from massive data streams, and existing solutions have often faced challenges to be adapted at scale for industry. It is within this context that the *City Analyser* project was proposed in Portugal, a consortium involving INESC-TEC – a research institute –, NOS – a Portuguese telecommunications and media company –, SONAE MC – a commerce company – and TUB – a public transportation company in Braga, Portugal.

The main objective of City Analyser was to develop a data platform to determine behaviours and habits of smartphone users, creating a client profile that enriches population-level analysis of movement patterns in urban environments. This data platform is the supporting environment for analytics focused on four main questions: area of influence of a particular location (where are the clients who visit that location coming from), mobility profiles (identification of city hot spots for social gatherings and mode of transportation when travelling), most frequent trajectories (identifying the path taken to travel between two locations) and anomaly detection (generate

alerts when user behaviour shifts from observed normal patterns). The contribution describes the methods developed to answer these four questions.

To achieve the outlined objectives and address the specified use cases, we have adopted a modular architecture known as "building blocks." Each building block represents a model designed to tackle a distinct task, such as user residency status identification or mode of transport classification. These building blocks are equipped to dynamically learn, predict, and populate temporary tables. They are subsequently queried or directly invoked by use case scripts to generate output tables, referred to as data products. These data products serve as the basis for generating insightful results and graphical representations.

While existing research in the field of mobility pattern mining often focuses on addressing specific or a limited set of related problems, typically narrowed down to specific issues. The research works enumerating various applications and methodologies are predominantly survey papers, often lacking practical implementations Atluri et al (2018). In contrast, our work is applied research, presenting a comprehensive exploration of diverse problems across three domains. Our contributions are multi-fold, as given below:

- The development of a sophisticated data platform with a holistic approach to data analysis, from use case formulation to module implementations, facilitates effective data processing and modeling.
- Utilized BigQuery for efficient and rapid data processing, retrieving model outputs and executing use cases.
- The models are adaptable to produce outputs tailored to various tessellations, including S2 cells [1] (grid system for indexing geographies developed by Google), sections (NUTS), and municipalities.
- Applied various machine learning techniques tailored to specific tasks, such as rule-based models for user profiles, transfer learning using Random Forest Classifier and Multilayer Perceptron for mode of transport classification, normalized Z-score for temporal anomaly detection of traffic flow, OD matrices, Ramer-Douglas-Peucker algorithm for simplifying trajectories, and DBSCAN with discrete Frechet Distance for clustering frequent trajectories.
- Addressed multiple challenges, including stay locations analysis, traffic flow analysis, catchment area analysis, market share of time analysis, and night stay area analysis, leveraging the models above.
- Created dashboards to visualise and analyse outputs resulting from data products.

The remainder of this paper is organized as follows: Section 3 reviews research works related to the issues addressed in this paper, divided into several topics of relevance given the multi-faceted nature of this study. Section 4 provides an overview of the architecture of the data platform, detailing the data flow, pipeline, and tools utilized, with Section 4.2 offering a comprehensive description of the dataset. Section 5 outlines the pre-processing steps and presents the fundamental definitions necessary for understanding the subsequent sections. All models developed and analyzed in this study are presented in Section 6, with detailed explanations of their structure

[1] http://s2geometry.io/

and function. Section 7 discusses various case studies and demonstrates the application of the building blocks in real-world scenarios. Finally, Section 8 summarizes the findings and conclusions of this research, highlighting the key contributions and potential areas for future work.

3 Related Work

The utilization of mobility data has proven invaluable in tackling numerous real-world challenges. It has facilitated in solving issues like predicting infectious disease outbreaks (Kraemer et al, 2020), investigating crimes (Gerber, 2014), protecting wildlife and environment (Cagnacci et al, 2010), modelling human movements in natural disasters (Han et al, 2019), and measuring the severity of air pollution (Shaji et al, 2022). Besides these, a vast range of literature focuses on leveraging mobility patterns for shaping the evolution of smart cities and urban development (Steenbruggen et al, 2015). Data-driven strategies have been explored for planning, managing and optimizing transportation systems (Zhu et al, 2018; Mahrez et al, 2021) fostering sustainable urban development. The nuanced interplay between human movement and urban dynamics has been extensively leveraged to transform cities into intelligent, responsive ecosystems. This focus aligns with the primary objective of our research. Therefore, we delve into relevant literature within these domains below, highlighting specific models and methodologies to tackle diverse tasks that are further investigated here. The classification is based on the nature of models used in trajectory analysis and predictive analytics.

3.1 Trajectory Analysis

Cluster mobility and profiling: The default approach can be simply distance-based algorithms (Ben-Gal et al, 2019), clustering-based models (Atluri et al, 2018) that focus on extending traditional clustering algorithms such as K-means, BIRCH, DBSCAN, OPTICS, and STING or collective motion arrest. These models are usually sensitive to variation in spatiotemporal scale and miss the length variant and irregular trajectories (Liao, 2005). (Yue et al, 2019) proposed DETECT that handles variable-length input by adapting an autoencoder trained using a large volume of unlabeled trajectories. Another class of algorithms deals with uncertain trajectory data where the object moves continuously. At the same time, the location is only recorded at discrete times (Yuan et al, 2017), which happens in the case of RAN Geolocation data.

Anomaly detection: Most anomaly detection works focus on clustering and point densities. More recently, tensor-based methods have been introduced as anomaly detection techniques. We have recently investigated the application of event detection tensor decomposition to dynamic O/D data using a hybrid tensor model called HTM (Fanaee-T and Gama, 2016). Sun et al (2017) detected non-recurring anomalies caused by real-time disruptions or events such as sports, weather, and accidents using Convolutional Neural Network (CNN). Similar strategies can be applied to avoid QoE (Quality of Experience) degradation in the mobile network.

Modeling catchment areas: Some techniques apply gravity models to analyze consumer behaviour and approximate the catchment area of a store or shopping centre by considering the spatial distribution of competition areas and assessing their attractiveness for different population groups (Dolega et al, 2016) based on some location factors such as the population of the city, the price and supply of the products offered. Some other models include entropy maximization (Daniotti et al, 2023), competitive destinations model (Cronjé and du Plessis, 2020), multi-purpose shopping model for estimating market share (Drezner et al, 2023), the travel-to-store model (Pratt et al, 2014), zone design problem (Kohání, 2012).

Characterizing users based on mobility patterns: Another application relevant to our case study is characterizing users into nationals (intra-region and inter-region), foreigners and tourists based on their mobility patterns. Xu et al (2021) introduced nine mobility indicators, such as duration of stay in a city, the spatial extent of activities, the location visited, and trips conducted, and mobility diversity to capture different facets of tourist travel behaviour. Further, they applied an eigen-decomposition approach and Principal Component Analysis to understand the variations and dependencies between the above indicators for individual travellers.

3.2 Predictive Analytics

Mobility prediction: This task encompasses problems like next location prediction, next trip, or trajectory prediction (Andrade and Gama, 2024; Ma and Zhang, 2022). Most works in the area are focused on predicting the mobility of a single user using different approaches such as Markov models (Andrade and Gama, 2024; Lv et al, 2016), non-linear time series analysis (Aljeri and Boukerche, 2020) and dynamic Bayes networks (Hou et al, 2016). However, the techniques mentioned above often fall short of accurately modelling multidimensional data and predicting the mobility patterns of multiple users concurrently. Advanced methods like tensor models and matrix factorization have been applied in multivariate spatiotemporal analysis and collaborative mobility prediction to address this complexity. For instance, the tensor-based prediction framework introduced by Bahadori et al (2014) can incorporate various properties in spatio-temporal data. However, tensor methods suffer from a cold start problem. Additionally, a crucial area of exploration involves real-time mobility prediction. An online CNN model is implemented for trajectory prediction in the work by Ouyang et al (2016). In another notable study by Fattore et al (2020), the authors propose a distributed structure for mobility prediction by using the Long Short-Term Memory model (LSTM).

Predicting traffic bursts and hotspots: Road-level traffic prediction is a similar problem as above that can be modelled as a time series of traffic flow data. Lately, RNN and LSTM are popularly used for time series spatiotemporal data prediction (Wang et al, 2020b). It is closely related to the problem of real-time event detection. It can be exploited for decision support in scenarios such as optimizing outdoor parking spaces and resources at supermarkets or opening new stores. Solé-Ribalta et al (2016) modelled traffic flow as a complex network and used the betweenness measure to analytically predict congestion hot-spots.

Transportation mode classification: Detecting the transportation mode of a trip is not a new problem, quite several works have investigated this problem and are still being benefited by the advancement in learning algorithms and quality of data. Huang et al (2019) presented a systematic review of transport mode detection based on mobile phone network data. Most of the traditional methods are rule-based where features like speed, duration and distance are used to categorize transport mode. However, Dabiri and Heaslip (2018) says hand-crafted features have drawbacks including vulnerability to traffic and environmental conditions, therefore, they take advantage of CNN architectures which automatically drive high-level features and predict travel modes based on only raw GPS trajectories, where the modes are labeled as walk, bike, bus, driving, and train.

Analysis and estimation of Origin/Destination flows: An OD matrix represents the flow of different entities from a set of origins to a set of destinations. The estimation of O/D matrices has been one of the primary concepts in transportation. Extensive research is carried out in this area to efficiently predict and estimate O/D matrices. A nuanced facet of this challenge involves the estimation of time-dependent O/D matrices. In Djukic et al (2012), the authors studied the use of Kalman filters to estimate O/D matrices. Krishnakumari et al (2019) explores a data-driven approach for this purpose. Moreira-Matias et al (2016) applied an incremental algorithm to discretize the target variable's historical values on each matrix cell. Ou et al (2019) employs a CNN-based model to learn the patterns from dynamic mappings between time-varying O-D flows. However, a significant challenge lies in the sheer size and high dimensionality of these matrices.

POI recommendation: Points of interest recommendation have emerged as one of the popular sub-problems in the area of recommendation with a plethora of works focused on this issue. Most of these works are concentrated on Location-based online social networks (Islam et al, 2022). However, they can be applied to mobility data gathered from any relevant source. This problem is very close to the above mobility prediction, except it focuses on the user level. The recommended POI or the so-called next place can be a known or unknown place to that user. Nevertheless, these personalized recommendations are essentially based on historical time series data such as the user's past visits, and location preferences and can leverage contextual information such as the user's current location, time of day, type of POIs, neighbourhood or explore collaborative approaches like friendship networks, similarity measures and clustering algorithms to recommend new places based on the trajectories of other users with similar mobility patterns. Wang et al (2020a) used the historical trajectories of a user to infer his future location. To that end, they employed a high-order Markov chain model to predict the most likely locations visited by each user. Kong and Wu (2018) applied HST-LSTM (Hierarchical Spatial-Temporal LSTM) to predict an individual's short-term next location. Liu et al (2021) proposed a category-aware gated recurrent unit (CA-GRU) model to decrease the impact of sparse check-in location data from social network services.

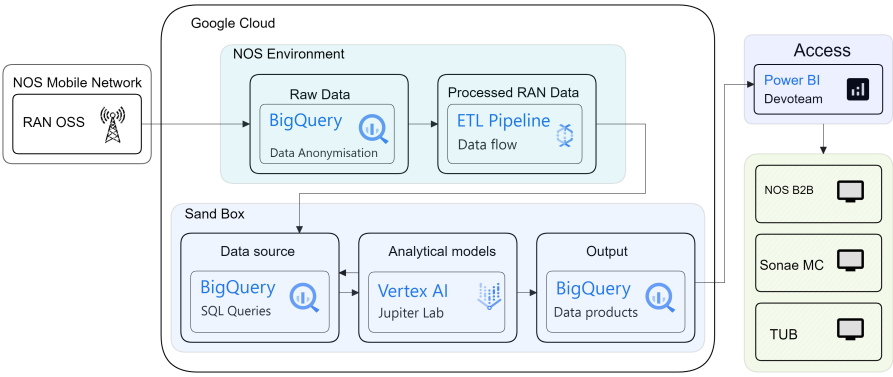


Fig. 1: Architecture of Sandbox in Google Cloud

4 Architecture of data platform

4.1 Development Environment

A working environment was provided by NOS for the research of analytical models by the INESC team. This environment is a sandbox that allows data exploration, model training and producing data products by harnessing the power of Python in Vertex AI, which is provided on GCP (Google Cloud Platform) (Google, Accessed on Mar 18, 2024a) as a complimentary next generation technology. A brief overview of the sandbox is given in Figure 1 below. RAW data from NOS Mobile Network's (another service area) Operation Support Systems (OSS) is fed into NOS project environment. This environment in Google cloud where the RAW data is processed and Anonymised (using techniques below) is fully isolated from the Sandbox provided to develop models. This strict isolation also ensures that there is no possibility of retrieving or downloading data, thereby safeguarding both privacy and security. Jupiter Lab integrated in Vertex AI facilitates querying and fetching data from only permitted Big Query tables using Python files. A range of these Python libraries are used to build machine learning models on the cloud console. Outputs will be accessed by all the project partners with specific permissions. Additionally, ingested by Power BI to generate reports and dashboards. Below we discuss in more detail the capabilities and functions of modules in this architecture.

4.1.1 Data Anonymisation

Any equipment connected to NOS's mobile network sends and receives data from the network, which depends on the equipment's action, from a telephone call to the sending of an email. This data contains technical information, such as the antennas to which the equipment is connected. Based on this data, and using mathematical models (such as trilateration) it is possible to calculate, among other things, the longitude/latitude of the equipment. Because the nature of the data is very sensitive, it is necessary to filter and anonymize the data before it is made available: Any information that allows users to be identified (number, type of equipment, etc.) is either

discarded or processed in such a way as to make the information unreadable; Users who are in rural areas, without at least X number of people nearby, are discarded, since, although anonymized, the date allows us to discern who the person is by their simple position; Users who are outside the areas of interest (areas A andB), or outside the period under study (between X and Y), are discarded. The final result was a source of anonymized atomic data (long/lat) of NOS customers, which are within the areas of interest, over time. In practice, for each instant of time (seconds), we can see the location of the aggregate of NOS customers that are within the areas of interest, but thanks to the anonymization made, it is impossible to see the location of a specific customer. With this data source, although it is impossible to identify an individual, it is possible to study population flows, main axes used, destination origins, population density, etc.

4.1.2 ETL Pipeline

Set up to run every day to use the dataset with the mobility information of mobile network users. Google Dataflow (Google, Accessed on Mar 18, 2024b) is a fully managed service that modifies and enhances data in both batch (historical) and stream (real-time) modes.

4.1.3 BigQuery

BigQuery (Google, Accessed on Mar 18, 2024c) is a serverless, highly scalable, and cost-effective cloud data warehouse that allows fast queries at petabyte scale. BigQuery's serverless architecture decouples storage and computing and allows them to scale independently on demand. This structure offers both immense flexibility and cost controls for customers because they don't need to keep their expensive computing resources up and running all the time. This is very different from traditional node-based cloud data warehouse solutions or on-premise massively parallel processing (MPP) systems. This approach also allows customers of any size to bring their data into the data warehouse and start analyzing their data using Standard SQL without worrying about database operations and system engineering.

4.1.4 Vertex AI

Vertex AI is a machine learning platform provided by Google Cloud (Google, Accessed on Mar 18, 2024d). It is designed to help users build, deploy, and manage machine learning models at scale. Vertex AI offers integration with BigQuery, a fully managed, serverless data warehouse provided by Google Cloud (read more above). This integration enables users to leverage the capabilities of both platforms for advanced data analytics and machine learning tasks. Once models are trained, Vertex AI allows users to deploy them to production environments for inference. It supports serving models through REST APIs and provides features for monitoring model performance and health. Google Cloud's Vertex AI platform offers integration with Jupyter Notebooks, providing with a flexible and interactive environment for developing, experimenting with, and deploying machine learning models. The models are developed in this environment using notebooks and python scripts.

4.1.5 Power BI

The interactive data visualisation software Power BI (Microsoft, Accessed on Mar 18, 2024) is used to visualise and interpret use case results and statistics. The access to output from the cloud to Power BI is restricted exclusively to authenticated service accounts, ensuring secure access. Subsequently, the reports and dashboards generated through this process are made available to partners through designated user accounts.

4.2 Dataset

This is a new dataset based on mobile phone data. The dataset contains 651.503 instances from 9 months or 250 days starting in March 2022 and finishing in November 2022, consisting of 466 different individuals. After cleaning and removing the duplicates it was reduced to 534.612 instances. The points were recorded in the Lisbon, Portugal area with a mixed granularity of sampling with more logs recorded as the users were using the network services such as streaming video, social networks, SMS or calls, and also some heartbeat for the telecom systems. No information about the users is derived from these data as the entire dataset is anonymized using personally identifiable information (PII) is any information connected to a specific individual that can be used to uncover that individual's identity, such as their social security number, full name, email address or phone number. Each point consists of a user sequential identification number, a pair of (latitude, and longitude), and a timestamp. A mobile country code was also provided but discarded as we did not use data other than the location points for this work. The data was delivered in two files and due to GDPR reasons, we can not make it public.

4.3 Data Monetization Models

The analytical models aimed to address the requirements of use cases defined in the project. These product models are indicated as building blocks are outlined in the next section. In the figure below we delineate the data analytics architecture. The data is updated in BigQuery tables from where the Data preprocessing step (Section 5.2 fetches data using Queries and then cleans and preprocesses. The data frames created from the data are used for model training and testing. Use cases call models or query the results stored in temporary Big Query tables which keep on updating on sliding window.

5 User trajectories from mobile communications

5.1 Definition of trajectory

Mobile communication data is provided by the mobile phone operator in the form of communication **logs** between the smartphone of a client in the operator's network and a Radio Access Network (RAN) tower and the location of the client is inferred using proprietary algorithms belonging to the mobile phone operator. Thus, a **log** is a data point of the form $L =$ (user, latitude, longitude, timestamp) that records the pseudo-anonymized user identifier, the geographic coordinates and time with precision to the

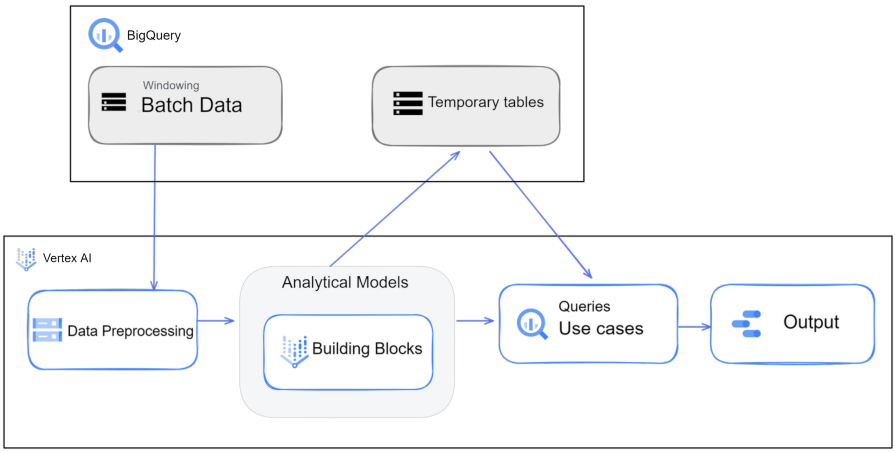


Fig. 2: Data Analytics Architecture

nearest second of the request to the communication antenna. Further information is given to inform the client profile, such as the Mobile Country Code (MCC) of the Subscriber Identity Module (SIM) card.

In accordance with the definition of (Feng and Zhu, 2016), we consider a **trajectory** to be a sequence of logs of a single user, $Traj = ((u, lat_1, lon_1, T_1), (u, lat_2, lon_2, T_2), \ldots, (u, lat_n, lon_n, T_n))$, with the following properties:

- $T_i > T_j \iff i > j$.
- $T_i - T_{i+1} < \Delta T$, where ΔT denotes the maximum period between consecutive logs in a trajectory. We define ΔT to be 30 minutes in this work.
- $\delta(lat_i, lon_i, lat_{i+1}, lon_{i+1}) < \Delta D$, where $\delta : \mathbb{R}\times\mathbb{R}\times\mathbb{R}\times\mathbb{R} \to \mathbb{R}$ is a function that given two geographic coordinates calculates the distance between them and ΔD is the maximum distance between consecutive logs in a trajectory. We abbreviate this notation to $\delta(P_i, P_j)$ to denote the distance between the coordinates of logs i and j of a given trajectory. As distances between points in our dataset are small enough that the curvature of the Earth can be ignored, we use the Euclidean distance as δ and define ΔD to be 1 kilometre.

A trajectory is called **stationary** if all logs are within a radius of the initial point of the trajectory, i.e., $\delta(P_1, P_i) < r_{SP},\ \forall i \leq n$. In turn, this means that the maximum distance between any points in a stationary trajectory is $2r_{SP}$. We define r_{SP} to be 200 meters.

A non-stationary trajectory is valid if the following properties are fulfilled:

- $\sum_{i=2}^{n} \delta(P_{i-1}, P_i)/(T_n - T_1) < \varepsilon_v$. The average speed of the user cannot exceed a maximum value (35 m/s), as exceeding this value is indicative of errors in data collection.

- Let V_{Traj} denote the average speed for the trajectory $Traj$, calculated as above. Then $V_{Traj}/(T_n - T_1) < \varepsilon_a$. The average acceleration also cannot exceed a maximum value (9.8 m/s^2), for the same reason.

A **stay point** in a trajectory is a significant segment of the trajectory where the client has little or no movement. Significance is determined by a minimum length of time ($\varepsilon_{SP} = 15$ minutes) the user must be in a certain radius (r_{SP}). Formally, a stay point is a sequence of logs $SP = (L_i, L_j)$, $1 \leq i \leq j \leq n$, such that $\delta(P_i, P_k) < r_{SP}$, $i \leq k \leq j$, and $T_j - T_i > \varepsilon_{SP}$.

5.2 Data Preprocessment

User communication logs reach the server for analytical processing daily as a big data stream, which we divide into chunks of one hour for ease of processing. Within this hourly data stream, we segment logs to identify trajectories that are **completed** instead of the remaining incomplete ones that could continue into the next hourly data chunk. This segmentation as a stream technique works by selecting users that have no data logs in the last ΔT minutes of a given hour or users that contain either a time (ΔT minutes) or distance (ΔD) gap between consecutive logs and considering the logs solely before this gap.

Upon determining which trajectories are completed, we apply trajectory preprocessing techniques, as suggested by Zheng (Zheng, 2015), namely noise filtering, compression, and stay point detection. For noise filtering, we use both a median filter (Andrade and Gama, 2018) with a window size of 5 and a heuristic-based outlier detection filter (Andrade et al, 2020a), with maximum values of instant velocity and acceleration set to 40 m/s and 15 m/s^2. It is worth noting that the analysis presented in this work is focused on mobility in urban environments with limited data collected on highways or other high-speed settings, hence the lower thresholds for these values when compared to other values suggested in related work.

Trajectory compression is particularly relevant to our problem, as people often use their mobile or smartphones when standing still, but the data logging process keeps on recording data, and due to the influence of GPS signal loss and data drift, several unwanted points show up. We use an online data reduction technique Douglas and Peucker (1973) for compression, in particular the sliding window algorithm Ramer (1972), discarding data points within a 25-meter radius of the anchor point. We modify the implementation of this algorithm within sci-kit-mobility Pedregosa et al (2011) to preserve the first and last points of the compressed sequence to avoid losing information on length-of-stay when users are at specific locations. This compression method reduced the amount of data stored between 75 and 80%.

Finally, we apply the stay point detection algorithm of Li et al. (Ye et al, 2009) to identify stay points with the characteristics detailed in the previous section.

6 Building blocks

As previously mentioned, the City Analyzer project involves a consortium of industrial partners from multiple sectors – tourism, commerce, and transportation. Despite the inherent diversity of these industrial domains, we identify recurring tasks every day to the business requirements of all partners and develop analytical models to solve them, which we refer to as "building blocks". In this section, we describe these building blocks.

6.1 Trajectories crossing polygons

Our analysis's most fundamental building block is matching trajectories to polygons of interest to know whether a trajectory intersects a geographical area. This building block is helpful to some questions posed by the use case owners, for instance, to know how many clients are in a certain area, how long they stay there or how many times clients visit the area over some time. This building block is also part of the remaining analytical models that require knowing whether a certain user is in a given geographical area. The input to this building block is an area of interest and an interval of time. The output is a list of users, their respective trajectories that cross the polygon and the times of entry and departure into/out of the polygon.

To calculate the time of entry, we search for pairs of consecutive points in a trajectory where a user was first outside the polygon, followed by inside the polygon. Similarly, to calculate the departure time, we search for pairs of consecutive points in the trajectory first inside and then outside the polygon. We also consider that a user's first data point may already be inside the polygon of interest, in which case we mark the time of entry as the time of this first data point and do the same in case a user's last data point is inside the polygon.

6.2 User Profile

Profiling is a technique used to identify users' behavioral patterns based on specific properties available in a particular context. In geospatial mobility data, existing literature predominantly focuses on crafting individual user profiles by leveraging features derived from their travel patterns. These features commonly include spatial distribution or visited locations, such as tourist spots or points of interest (POIs). However, our approach diverges by focusing on the locations of stay rather than movement to delineate user profiles. This shift in perspective enables a more nuanced understanding of user behaviour, emphasizing stationary contexts to extract valuable insights into users' places of stay and their residence and occupational statuses. For instance, we take a reverse perspective instead of adopting the common practice of classifying users staying in a hotel as tourists. We focus on identifying users' frequent stay locations and derive relevant features to ascertain their residency status. Subsequently, we examine whether these identified locations coincide with hotels or not.

Here, we discuss the five models we developed for building User profiles, which are also the basis for other building blocks. All the models presented below operate incrementally, processing data daily. The temporal dimension for each day is bifurcated into $daytime$ and $nighttime$ intervals. According to our domain experts,

the hours are defined as $08:00 \leq daytime < 20:00$ hours and $20:00 \leq nighttime < 08:00$ hours of the following day. After each day given the input of user locations (latitude and longitude) with timestamps, the following statistics are incremented for each user i from the evolving (users getting added or deleted) set of users $\mathcal{U}$ and area a from the finite set of areas $\mathcal{A}$, where $\mathcal{A}$ can be tessellated into different hierarchical levels of geography such as S2 cells, subsections, sections, parish, municipality, district and so on.

- $night\_count_i^a$: Number of nights spent by user i in the area a
- $day\_count_i^a$: Number of days spent by user i in the area a
- $stay\_time\_day_i^a$: mean time spent in the daytime by user i in the area a
- $stay\_time\_night_i^a$: mean time spent in the nighttime by user i in the area a
- $length\_of\_stay_i^a$: Duration of stay (nights $\cup$ days) spent by user i in the area a

We omit explicit mention of individual users (u) and areas (a) for all subsequent references to simplify notation.

6.2.1 User Night Stay Location Identification

We define a user's night stay location as the primary place where they spend most of their time during the $nighttime$, often synonymous with their home. The S2 cell corresponding to this night stay location is identified in this context. This determination is made within the framework of a given geographical area, which is tessellated into S2 cells. For each S2cell a and user u we calculate the statistics $nightcount_i^a$, $daycount_i^a$, and $staytimenight_i^a$. The night stay place is then labelled as `"yes"` or `"no"` based on the implemented rules:

- $night\_count > 0$
- $stay\_time\_night > \theta$
- The location with the highest $stay\_time\_night$ meeting these criteria is classified as the night stay place for the user.

6.2.2 User Day Stay Location Identification

The definition of a user's daytime stay location mirrors that of the nighttime stay location, with the distinction being the focus on daytime activities. This characterization is pivotal in identifying a user's work location. The corresponding S2 cell associated with the daytime stay location is pinpointed within a given geographical area, which undergoes tessellation into S2 cells for structured analysis. Similarly, for each S2cell a and user u we calculate the statistics $daycount_i^a$, $nightcount_i^a$, and $staytimeday_i^a$. The night stay place is then labelled as "yes" or "no" based on the implemented rules:

- $day\_count > 0$
- $stay\_time\_day > \theta$
- The location with the highest $stay\_time\_day$ meeting these criteria is classified as the day stay place for the user.

6.2.3 Residency Status Classification

This is one of the pivotal blocks in the framework and serves as a basis for many other blocks. A user's residency status is labelled as Resident, Tourist or Visitor. A user's residency status is calculated for different levels of hierarchy, basically S2cell, section, parish, municipality and district level. The user location data (latitude, longitude), timestamp and MCC code are inputted. For each user and the area in which all the features defined above $day\_count_i^a$, $night\_count_i^a$, $stay\_time\_day_i^a$, $stay\_time\_night_i^a$, $length\_of\_stay_i^a$ are updated per day by merging the previous day information with the current statistics. The labels of users are updated for every geography in the hierarchy based on the following conditions outlined in the algorithm below:

6.2.4 Residency area classification

The objective of this component is to classify the location where a user spends a certain amount of time, categorizing it as either their home, a hotel, undefined, or other type of location. We utilize information about amenities in a given geographical area to achieve this goal, specifically focusing on hotels. With access to this data, we employ map matching techniques to correlate user night-stay locations obtained from building block 3 with buffered* hotel areas. By doing so, we can determine whether a user stays in their residence, a hotel, or another (neither their home nor a hotel). In cases where there is insufficient information to ascertain the user's night-stay location, we label it as "undefined."

6.2.5 Professional status classification

In line with the preceding block, which used external data sources to furnish information regarding amenities within a locale, we extend our approach to incorporate data concerning the geographical boundaries of educational institutions. This data is then used to correlate with the user's daytime stay location, resulting from Building Block 2.2 (implicitly assuming these to be places of employment, excluding those who operate from home). The coordinates of daily stays are mapped against the buffered zones encompassing educational institutions. Those intersecting with such zones are categorized as 'students'. Individuals whose daily stays remain unidentified are designated as 'other', while the remainder are classified as 'workers'."

6.3 Mode of transport classification

Determining the mode of transport people use to move between locations is an important facet of urban mobility analysis, allowing outcomes such as public transportation route planning. This importance has led to the emergence of a rich methodological field aimed at automatically identifying the mode of transport given a trajectory Andrade and Gama (2022). However, most work in this area relies on GPS data, which is more regular and precise than location inferred from mobile communications data (Andrade, 2024). In addition to the challenges inherent to this task, our dataset is completely unlabeled, limiting the number of previous approaches applicable to our context.

Algorithm 1 Determining Residency Status

Require: A row of user data with attributes: *old_res_status, new_res_status, length_of_stay, night_count, stay_time_night, stay_time_day, mcc*
Ensure: Residential status of the user as 'Casual Visitor', 'Regular Visitor', 'Commuter', 'National Tourist', 'International Tourist', or 'Resident'

function DETERMINERESIDENTIALSTATUS(row)
  if $residential\_status\_x$ is null **then**
    return $residential\_status\_y$
  end if
  if $residential\_status\_y$ is null **then**
    if $length\_of\_stay$ $<$ min_length_of_stay and $stay\_time\_night$ $<$ min_night_stay_time and $stay\_time\_day$ $\geq$ max_commute_time **then**
      return 'casual visitor'
    else if $length\_of\_stay$ $\geq$ min_length_of_stay and $stay\_time\_night$ $<$ min_night_stay_time and $stay\_time\_day$ $\geq$ max_commute_time **then**
      return 'regular visitor'
    else
      return $residential\_status\_x$
    end if
  end if
  if $stay\_time\_night$ $<$ min_night_stay_time and $stay\_time\_day$ $<$ min_night_stay_time **then**
    return 'commuter'
  end if
  if $length\_of\_stay$ $<$ min_length_of_stay and min_night_stay_time $\leq$ $stay\_time\_night$ $<$ min_night_stay_time and $stay\_time\_day$ $<$ max_commute_time **then**
    return 'casual visitor'
  end if
  if $length\_of\_stay$ $\geq$ min_length_of_stay and min_night_stay_time $\leq$ $stay\_time\_night$ $<$ min_night_stay_time and $stay\_time\_day$ $<$ max_commute_time **then**
    return 'regular visitor'
  end if
  if $length\_of\_stay$ $\geq$ min_length_of_stay and $night\_count$ $\geq$ 0.0 and $stay\_time\_day$ $\geq$ max_commute_time and $stay\_time\_night$ $<$ min_night_stay_time **then**
    return 'regular visitor'
  end if
  if $length\_of\_stay$ $<$ min_length_of_stay and $night\_count$ $\geq$ 0.0 and $stay\_time\_day$ $\geq$ max_commute_time and $stay\_time\_night$ $<$ min_night_stay_time **then**
    return 'casual visitor'
  end if
  if $night\_count > 0.0$ and $night\_count <$ max_night_count and $stay\_time\_night \geq$ min_night_stay_time **then**
    if $mcc ==' 268'$ **then**
      return 'national tourist'
    else
      return 'international tourist'
    end if
  end if
  if $night\_count$ $\geq$ max_night_count and $stay\_time\_night$ $\geq$ min_night_stay_time **then**
    return 'resident'
  end if
end function

Fig. 3: Semi-supervised framework for mode of transport classification, based on pseudo-labeling and co-training.

Considering the challenges we have just outlined, we propose a semi-supervised learning model based on pseudo-labeling Cascante-Bonilla et al (2021) and co-training Chen et al (2022) for a mode of transport identification. Our approach works by considering two classifiers pre-trained on the popular trajectory mining dataset GeoLife Zheng et al (2010) and iteratively refining them until an agreement threshold is reached. See Figure 3 for a diagram of the models and training scheme, more details follow.

The GeoLife dataset (Zheng et al, 2010) is a dataset collected by Microsoft Research Asia, tracking the GPS location of a set of people over a long period. The most relevant aspect of this dataset for our development is that some trajectories are labelled with the mode of transportation, including different labels for different legs of a single journey (for example, someone using the train and reaching their final destination by foot). The available labels include train, taxi, walk, bus, subway, aeroplane, car, bike, boat, run and motorcycle. We drop trajectories labelled aeroplanes, boats, and motorcycles, merge trains with subways, cars with taxis, and walk with a run. After removing trajectories with less than 10 data points, the final dataset contained 8191 trajectories split among the five modes of transportation labels: 3416 on foot, 1692 by bus, 1250 by car, 1121 by bike and 712 by train. To become less susceptible to class imbalance issues, we randomly undersample the *on foot* class so that the training data has 2000 examples belonging to this class.

To train our mode of transport classifier, we start by training an initial classifier on the GeoLife dataset based on the work of Andrade and Gama (2022), who concluded that a Random Forest Classifier (RF) trained on trajectories described by their velocity and acceleration aggregated statistics (average, median, standard deviation and maximum value) can separate mode of transportation classes without external data. Our co-model for the co-training learning scheme is a Multilayer Perceptron (MLP), which sees a different view of the data by a normalization pre-processing step applied to the features. We used the scikit-learn (Pedregosa et al, 2011) implementation for both these models, tuning the maximum depth of each decision tree and the number of estimators for the RF and both the number of layers and the number of neurons per layer for the MLP using 10-fold cross-validation, keeping the rest of the hyperparameters equal to the default.

Upon training the two initial models, we transfer them into the domain of our mobile communications data and start the pseudo-labeling phase. We perform inference on the trajectories extracted from the mobile communications data and assume as true that the trajectories are assigned the same label by the two initial models. We gather this subset of trajectories as new training data and re-train an RF and an MLP using the same procedure (hyperparameter tuning with cross-validation). This process is repeated until the two models agree on 5% of the labels, after which the RF model is saved and used for inference.

6.4 Origin-destination - Traffic flows

We approach the problem of road traffic analysis from an anomaly detection perspective, attempting to find a city's normal spatial and temporal origin-destination traffic flow and creating a metric that highlights when the traffic flow deviates from normal behaviour. This approach is grounded on the requirements set out by this project, in contrast to related literature on traffic analysis that tackles the issue from a variety of approaches, allowing us to answer questions such as what are the optimal locations for advertisement or when should mobile network capacity be boosted to meet mobile traffic requirements that deviate from normal.

Our methodology is based on a tessellation of an input area, that is, dividing the input geography into smaller units that cover the whole initial area to map continuous geographical coordinates and the discrete regions. The anomaly detection module is agnostic to the tessellation of choice, as the tessellation method should vary according to the end application of the method. For example, an application that requires knowledge of traffic in a fine-grained area may consider a tessellation based on Google S2 cells; on the other hand, if such fine-grained detail is unnecessary or uninformative, we have also considered tessellations based on municipality and parish borders (Nomenclature of Territorial Units for Statistics (NUTS) levels 4 and 5).

The metric used to identify temporal anomalies is based on the traffic time series for a given unit of the chosen tessellation over one week. Given a tessellation with k units $\mathcal{T} = \{C_1, C_2, \ldots, C_k\}$, let ζ_i^t denote the traffic in cell C_i at time t, where t denotes one hour (for example, from 00:00 to 01:00). Then, we define the temporal anomaly score (TAS) as:

$$Z_i^t = \frac{\zeta_i^t - \overline{\zeta_i}}{\sigma_{\zeta_i}}, \qquad TAS_i^t = \frac{Z_i^t}{\sqrt{\sum_k Z_i^k}}, \tag{1}$$

where $\overline{\zeta_i}$ is the mean traffic in cell C_i over the week before time t (previous 168 hours) and $\sum_k Z_i^k$ indicates the sum of Z_i over the same time period. Thus, Z_i^t is the Z-scored traffic of a cell concerning its traffic over the previous week, and TAS is simply the normalized Z-score. Similarly, we define the spatial anomaly score (SAS) as the normalized Z-score for a certain time across all cells of a given tessellation:

$$Z_i'^t = \frac{\zeta_i^t - \overline{\zeta^t}}{\sigma_{\zeta^t}}, \qquad SAS_i^t = \frac{Z_i'^t}{\sqrt{\sum_k Z_k'^t}}, \tag{2}$$

, where $\overline{\zeta^t}$ is the mean traffic across the area of study at time t and $\sum_k Z_k'^t$ indicates the sum of Z^t over the same period

6.5 Frequent trajectories

With the development of location-based positioning devices and the advent of the Internet-of-Things (IoT), more and more moving objects are traced, and their trajectories are recorded, joining diversified information about their carriers and

equipment. These data comprise a rich source of spatial and temporal semantic information. Therefore, moving object trajectory clustering undoubtedly becomes the focus of the study in moving object data mining (Zheng, 2015).

Many areas can leverage the similarity of trajectory analysis, such as policy-makers/government, transportation companies/authorities, last-mile parcel carriers, biologists, and retail and marketing companies. In the public sector, the managers can analyze the moves of people at different hours of the day and week to promote changes in the infrastructure of a region, change the bus routes, increase the number of trains or metro cars, and take measures to diminish the bottlenecks of traffic hotspots and try to diminish the vehicle emissions. The private sector can also use these studies to target advertisements to specific groups of users that travel along some routes or visit some points of interest. Biologists can use these techniques to help understand the whereabouts of animals such as birds and fish, where they go, and which are the recurrent routes taken. Tourism in both public and private sectors can also make good use of trajectory analysis by recommending tourist routes or using these routes to improve or even deploy services along the path.

A common approach to performing trajectory analysis is by making use of clustering techniques where the process assigns a set of similar trajectories into groups (the clusters) having highly similar trajectories within each cluster and low similar trajectories among the different cluster sets (Zheng, 2015; Yuan et al, 2017). Among the clustering approaches, one, in particular, has shown to be more suitable for trajectory analysis due to its possibility of forming clusters of arbitrary shapes in Euclidean space: the density-based approach. One of the most popular algorithms in this group is DBSCAN (Ester et al, 1996). Still, one key component of good-quality trajectory analysis is how to calculate the similarity between trajectories in a group. Different similarity measures can be used, but not all consider the order of the data points in the trajectory set, which is paramount for a good-quality cluster of similar trajectories. Fréchet distance is one of the metrics that can be used to solve the problem.

Frequent Trajectory: A frequent trajectory is described as a regular route an individual tends to follow when travelling/moving between two locations (origin and destination) (Andrade et al, 2020a). A real-life example can be a street or highway to drive from home to work, a metro line used to commute, a sidewalk to walk to the user's preferred restaurant, a shopping mall, etc. In this study, we have focused on discovering the most frequent places that individuals visit and the common routes related to these displacements.

Other characteristics are also important to mention:

- Trajectories may have different lengths as individuals tend to move accordingly to their needs and singularities (e.g., N and M can be different for $Tr_i = (p_1, p_2, \ldots, p_N)$ and $Tr_j = (p_1, p_2, \ldots, p_M)$).
- Trajectories may have different directions. In the context of individuals' movement, the direction of each trajectory is an essential condition for the similarity of trajectories. As we propose the discovery of frequent routes, two trajectories moving in opposite directions should be considered different moves despite their proximity. They may represent different habits (e.g., going to work from home and going back home from work).

6.5.1 Trajectories Simplification with Ramer-Douglas-Peucker (RDP)

In some cases, GPS raw data can be very densely represented. The three datasets used in this work have different granularities, and we do not need much detail for the frequent trajectories discovery/clustering step. Many of these points can be removed as they are somehow redundant, whereas other key positions must be kept. An excellent way to avoid unnecessary processing is by using compression techniques. We use the Ramer-Douglas-Peucker algorithm (Ramer, 1972; Douglas and Peucker, 1973) to simplify the trajectories. The algorithm aims to produce a simplified poly-line with fewer points than the original but still keeps the original's characteristics/shape. The method takes one threshold parameter ϵ and connects the original line's first and last point with a reference OD pair. It then finds the point furthest away from that baseline reference and checks if it's greater than ϵ. If true, it keeps the point, and the function recursively splits the line into two segments, creating new reference points and repeating the procedure. If the point is nearer to the baseline reference than ϵ it discards all the points between these reference points simplifying the trajectory.

Figure 4 (a) shows an example of a trajectory split by the RDP algorithm.

6.5.2 Clustering Algorithm and Similarity Measures

Clustering is an efficient way to group data into different classes based on the internal and previously unknown schemes inherent in the data, and trajectory clustering is the most popular topic in current trajectory data mining. The aim is to discover the similarity (distance) in moving object databases, grouping similar trajectories into the same cluster, and finding the most common patterns (Yuan et al, 2017).

Density-based clustering techniques are very popular methods for location detection. They can detect clusters of arbitrary shapes without specifying the number of clusters in the data beforehand. Furthermore, they are tolerant of outliers (noise). Some recent studies have addressed the location detection techniques to improve the quality of the discoveries (Andrade and Gama, 2018; Andrade et al, 2019, 2020c,b,a).

In this study, we apply the clustering method proposed by (Andrade et al, 2020a), which is a variation of DBSCAN (Ester et al, 1996) to form the clusters of trajectories between the start (origin) and end (destination) points of all the trajectories.

One of the most important parts of a clustering algorithm is the similarity measure of two items. This is the step where the distance of two points is calculated before the algorithm decides whether to group these items. Different comparison strategies must be taken according to the purpose of the clustering task. Some of the most common distances are Euclidean, Hausdorff (Zheng, 2015), Longest Common Sub-Sequence (Vlachos et al, 2002), Dynamic Time Warping (Zheng, 2015), and Fréchet distance (Eiter and Mannila, 1994).

For the Euclidean distance (ED), the similarity between the two trajectories is simple and intuitive because it is parameter-free. In addition, its time complexity is linear, meaning it can handle a large dataset. However, noise existing in trajectory data will have a great influence on the result. Another main disadvantage of using the ED for measuring the similarity between trajectories is that the sampling points must be in corresponding positions (at the same time), and the trajectories must have

the same length. This is not true in real-world scenarios, even though the origin and destination are the same.

Hausdorff distance (HD) between trajectory segments A and B selects the maximum unidirectional HD from A to B and from B to A. It measures the maximum mismatching degree between two trajectory segments. HD tolerates the influence caused by point disturbance but is sensitive to noise data. This last point is also an issue in real-world scenarios when dealing with GPS data due to the signal interference caused by objects. For this reason, we avoid using this distance function.

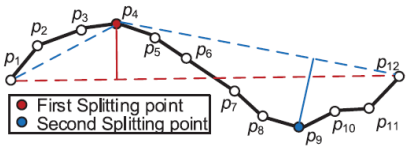
For the Longest Common Sub-Sequence (LCSS), as the name suggests, the idea is to get the longest list of common items in sequence between two trajectories. It uses a distance function (ED or any other) to compare if the combination of a pair of points is less than a threshold ϵ. Having the distance value less than the expected threshold, the value of LCSS is increased by 1. One advantage of LCSS is that it allows certain deviations in the sampling data (common in the real world). The advantages are the distance measure choice and parameter specification as well as the time complexity.

The Dynamic Time Warping (DTW) algorithm was proposed to find an optimal alignment between two given (time-dependent) sequences under certain restrictions. This method can match trajectories even if their lengths are different. The goal is to minimize the warping cost of finding similar paths between two trajectories. It is also sensitive to noise. The disadvantages are that when two trajectories are completely dissimilar in a small range, the DTW distance cannot be found, and the time cost and complexity are higher than the previous techniques.

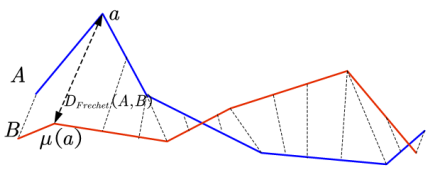
Finally, the discrete Fréchet Distance (DFD) considers the sequential relationship and the location of the points in the trajectories while measuring their similarity. It also relies on ED to calculate the distance in a point-wise fashion, as shown in equation 3.

$$DFD(x,y) = \max\left(\|p_{i(t)} - q_{i(t)}\|, \min\left(DFD(x-1,y), DFD(x,y-1)\right)\right) \quad (3)$$

where given two sequences of points p=(p_1, p_2, p_3, …, p_n) and q = (q_1, q_2, q_3, …, q_m), Fréchet distance represented by $DFD(x,y)$ is the maximum of the minimum distances between points p_i and q_i. Figure 4 (b) shows an example of the Fréchet distance between two trajectories.



(a) Douglas-Peucker algorithm (Zheng, 2015)



(b) Fréchet distance between trajectories

Fig. 4: RDP and Fréchet distance examples

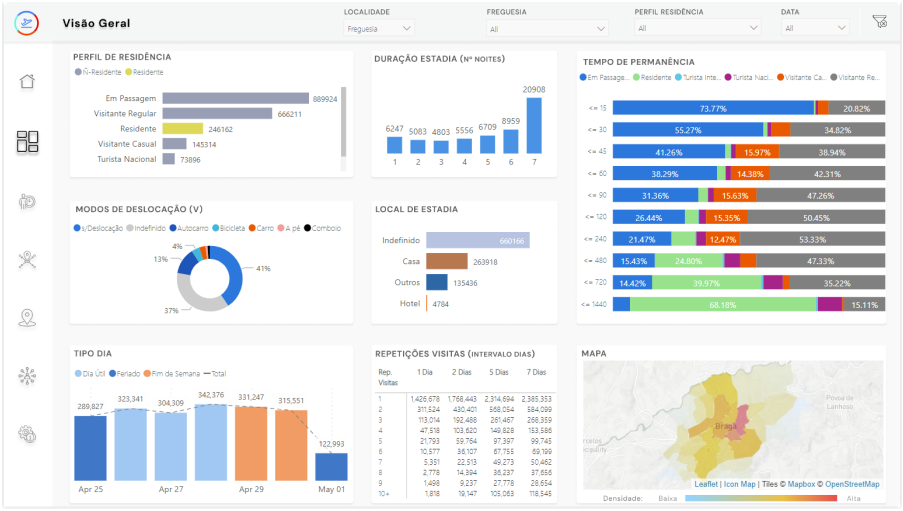


Fig. 5: Demonstration of dashboard_1 displaying the results of use cases in Power BI

One of the critical issues previous studies have shown is that the Fréchet distance contains the temporal relationship between the points. More particularly, the structure of the nodes inside the trajectory is considered in the computation process, which can more accurately describe the similarity between the trajectories, yelling better results (Magdy et al, 2015). In some scenarios with a backward direction, ring, or crisscross in a trajectory, the Fréchet distance value doesn't show more distortion than other distance measurements. Due to these characteristics, this metric is more descriptive and more suitable for measuring the similarity between trajectories. The time complexity is also similar to the other mentioned metrics.

For this work, we used the DFD to cluster the trajectories. We first construct a symmetric distance matrix of each pair in the trajectories connecting the given OxD using the DFD to obtain the clusters of frequent trajectories between the origin and destination pairs. We then fit the symmetric distance matrix to the DBMeans (Andrade et al, 2020a) method to obtain the different groups of trajectories.

7 Use cases

Below we delineate the use cases identified by stakeholders to guide the utilization of project resources and generate expected outputs. The building blocks discussed above were structured to be shared across multiple use cases. These use cases leverage the models by calling them directly or querying results stored in temporary tables (saved by models) within BigQuery, which continuously updates in real-time against incoming data streams. All the use cases utilize the output table generated by the residency status classification model to categorize and organize the results according to user profiles and other models. Queries for each use case are executed to produce outputs tailored to the needs defined by our partners. These results are generated for specific time intervals and geographic areas, which users can customize as parameters.

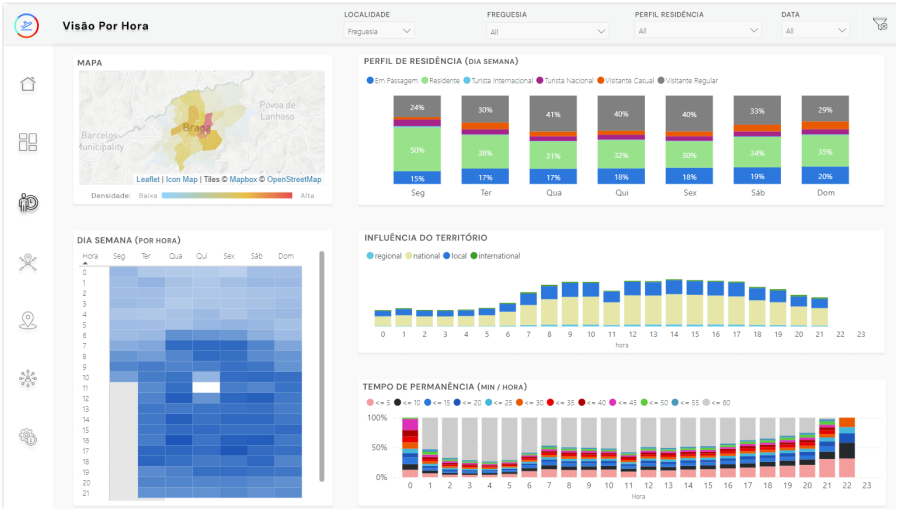


Fig. 6: Demonstration of dashboard_2 displaying the results of use cases in Power BI

The outputs, serving as data products aligned with the identified use cases, were stored in the cloud or designated buckets. Subsequently, the results were seamlessly transferred to Power BI to create comprehensive reports and interactive dashboards, enabling stakeholders to visualize and interpret the results effectively.

7.1 Area analysis

7.1.1 Area Population Count

This use case involves analyzing the population count within specific geographical areas over hourly or daily intervals while considering the distribution of people from different residency statuses. The geographical areas, represented by polygons such as Municipality, Parish, Section, Subsection, or s2cells, are defined by their unique polygon IDs. The analysis focuses on understanding the distribution and fluctuations of population counts within these areas over time.

7.1.2 Time spent Analysis

It aims to analyze the duration of time spent by individuals of different residency statuses within designated geographical areas/polygons. The analysis was conducted hourly and daily at various intervals, such as 5, 10, 15 minutes, and so on. It provides insights on how long the individuals stay in different locations and in which areas people tend to spend more time.

7.1.3 Different visits

This use case aims to track the frequency of visits by individuals to particular geographic regions/polygons of interest over a defined time frame, considering their

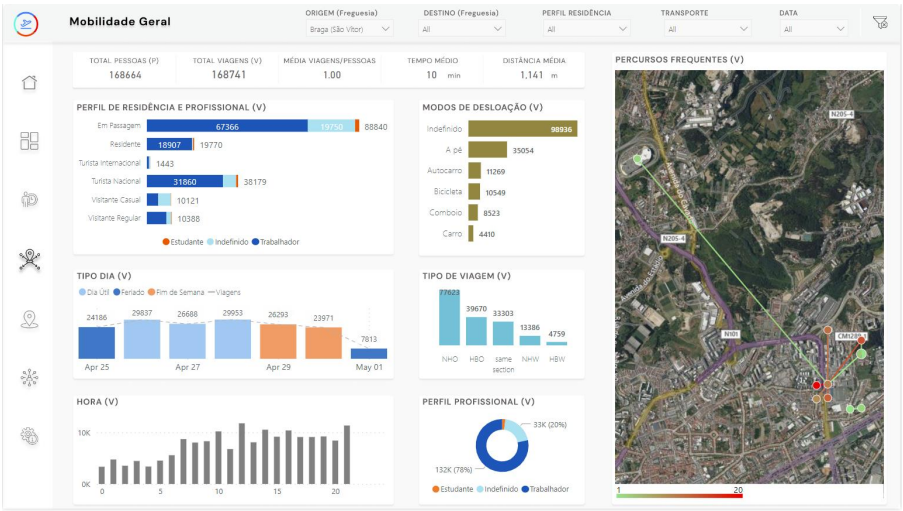


Fig. 7: Demonstration of dashboard_3 displaying the results of use cases in Power BI

residency status. This involves counting the number of repeat visits each person makes within the designated area.

7.2 Catchment area analysis

The catchment area represents the region or territory people travel to access a particular service, facility, or business. In this use case, we analyze the number of visitors in different categories from residency status (international tourists, national tourists, casual visitors, and regular visitors) within specific geographic areas (such as municipalities, parishes, sections, or subsections) during each hour. The analysis distinguishes individuals as local, regional, national, or international. Where ”local” and ”regional” means the users are residents in the same municipality or district as the geographic area being analyzed. Conversely, if the label is ”national” or ”international,” it indicates that the visitors come from a district or country other than Portugal. This use case provides valuable insights into visitor demographics and their distribution across locations.

7.3 Night stay analysis

7.3.1 Night stay classification

This use case focuses on understanding where people stay overnight within specific geographic areas during a given period. It tracks the number of individuals staying in different accommodations, such as hotels, homes, or other establishments, based on their night stay location and residency status given by models in section 3. The analysis is conducted within polygons representing various geographical divisions like counties, parishes, sections, or points of interest (POIs).

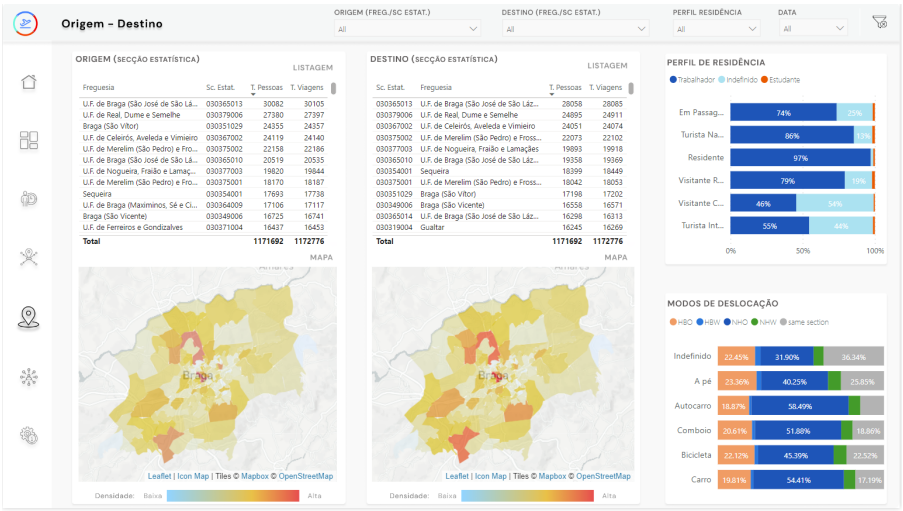


Fig. 8: Demonstration of dashboard_4 displaying the results of use cases in Power BI

7.3.2 Understanding night stay patterns

The problem involves quantifying the number of individuals who spend nights within various geographic areas (S2 cells, subsections, sections, parish, municipality, district) over specific time intervals classified by their residency status. For instance, within a municipality, the aim is to determine the distribution of residents, tourists, visitors, etc., among the individuals staying there overnight.

7.4 Stay locations analysis

This use case involves analyzing the locations where people stay based on their professional status (student, worker, or other) and residency status. It focuses on counting the number of individuals either studying or working within specific geographic areas during a specified period between the start and end dates and their residency status in that area. This information was intended to help transportation planners tailor bus routes and schedules to better meet the needs of students and employees. By optimizing transportation services, the solution aims to improve overall efficiency, reduce congestion, and enhance the commuting experience for residents or visitors.

7.5 Traffic flow Analysis

7.5.1 Hourly Traffic flow

Here, we analyze the traffic flows hourly within specific geographic cells. For each hour and cell, the number of people present is recorded. Additionally, the significance_spatial metric compares the number of people in each cell against the average for all cells, indicating over- or under-representation. Similarly, the significance_temporal metric compares the count of people in a cell over time, providing

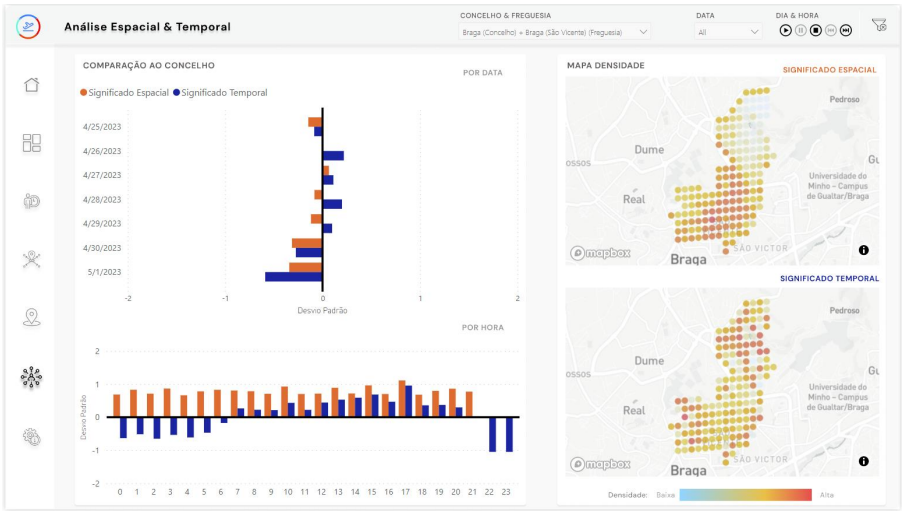


Fig. 9: Demonstration of dashboard_5 displaying the results of use cases in Power BI

insights into temporal trends in traffic flow. This analysis helps understand patterns of human activity and movement within different geographic areas over time.

7.5.2 Origin-destination patterns

This use case involves analyzing travel patterns between different locations. For each hour, we track the number of trips and unique individuals traveling between specific areas, identified by their origin and destination. The analysis includes details such as the mode of transportation, the residency status of travelers, and the purpose of each trip. Additionally, we provide insights into the average duration and distance traveled for each trip category.

7.6 Market share of time

This use case involves analyzing the percentage of time individuals spend inside a specific business/service area compared to other business areas within a defined geographic area (e.g., Municipality, Parish, Section). The analysis is conducted hourly to understand the distribution of time individuals spend across different supermarkets in the area.

7.7 Frequent trajectories

The context where the frequent trajectories were applied in the project is related to a public transportation agency where the player wanted to know the mobility profiles and the most common routes used to commute in a given region. The main aim is to optimize service delivery, adapt offers, and identify users' necessities.

8 Conclusions

In this work, we present a comprehensive large-scale study and implementation of analytics across various components of our project. We demonstrate the analysis of real-world spatiotemporal data to address practical problems, improve services, and drive advancements in urban environments. Our architecture is designed to meet industry needs and can handle vast amounts of data swiftly and effectively. We detail the extraction of trajectories, stay points, and preprocessing techniques essential for accurate data analysis.

Our research employs a range of machine learning models and algorithms, including rule-based models and transfer learning, showcasing the adaptability and reusability of these models across multiple use cases. We extract features such as average night and day stay times and length of stay, which can be incrementally updated to reflect changes in a user's residency or professional status. For the mode of transport identification, we propose a semi-supervised learning model based on pseudo-labeling and co-training, addressing the challenge of limited labeled data.

We identify anomalies in a city's normal spatial and temporal traffic flow by developing a metric that highlights deviations from typical traffic patterns. Additionally, we explore various techniques and distance measures to calculate frequent trajectories and discuss the suitability of different approaches in this context.

The practical applications derived from our models are encapsulated in several use cases, which include detailed statistics on geographical areas, such as user counts, visit frequency, time spent, and traffic flow between origins and destinations. The analysis also enabled the identification of different user profiles based on residency, profession, modes of transport, influence in catchment areas, and market share. These insights are visualized through interactive dashboards, providing stakeholders with valuable decision-making and strategic planning information for the selected geographical areas.

The authors acknowledge the project AI-BOOST funded by the European Union under GA No 101135737.

References

Aljeri N, Boukerche A (2020) A performance evaluation of time-series mobility prediction for connected vehicular networks. In: Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks, pp 127–131

Andrade T (2024) Mobility patterns from data

Andrade T, Gama J (2018) Identifying points of interest and similar individuals from raw gps data. In: EAI International Conference on Smart Cities within SmartCity360° Summit, Springer, pp 293–305

Andrade T, Gama J (2022) How are you riding? transportation mode identification from raw gps data. In: EPIA Conference on Artificial Intelligence, Springer, pp 648–659

Andrade T, Gama J (2024) Where do we go from here? location prediction from time-evolving markov models. In: Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing, pp 365–367

Andrade T, Cancela B, Gama J (2019) Discovering common pathways across users' habits in mobility data. In: EPIA Conference on Artificial Intelligence, Springer, pp 410–421

Andrade T, Cancela B, Gama J (2020a) Discovering locations and habits from human mobility data. Annals of Telecommunications 75(9):505–521

Andrade T, Cancela B, Gama J (2020b) From mobility data to habits and common pathways. Expert Systems 37(6):e12,627

Andrade T, Cancela B, Gama J (2020c) Mining human mobility data to discover locations and habits. In: Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Proceedings Part II, Springer, pp 390–401

Atluri G, Karpatne A, Kumar V (2018) Spatio-temporal data mining: A survey of problems and methods. ACM Computing Surveys (CSUR) 51(4):1–41

Bahadori MT, Yu QR, Liu Y (2014) Fast multivariate spatio-temporal analysis via low rank tensor learning. Advances in neural information processing systems 27

Ben-Gal I, Weinstock S, Singer G, et al (2019) Clustering users by their mobility behavioral patterns. ACM Transactions on Knowledge Discovery from Data (TKDD) 13(4):1–28

Cagnacci F, Boitani L, Powell RA, et al (2010) Animal ecology meets gps-based radiotelemetry: a perfect storm of opportunities and challenges. Philosophical Transactions of the Royal Society B: Biological Sciences 365(1550):2157–2162

Cascante-Bonilla P, Tan F, Qi Y, et al (2021) Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. In: Proceedings of the AAAI conference on artificial intelligence, pp 6912–6920

Chen M, Du Y, Zhang Y, et al (2022) Semi-supervised learning with multi-head co-training. In: Proceedings of the AAAI conference on artificial intelligence, pp 6278–6286

Cronjé DF, du Plessis E (2020) A review on tourism destination competitiveness. Journal of Hospitality and Tourism Management 45:256–265

Dabiri S, Heaslip K (2018) Inferring transportation modes from gps trajectories using a convolutional neural network. Transportation research part C: emerging technologies 86:360–371

Daniotti S, Monechi B, Ubaldi E (2023) A maximum entropy approach for the modelling of car-sharing parking dynamics. Scientific Reports 13(1):2993

Djukic T, Flötteröd G, Van Lint H, et al (2012) Efficient real time od matrix estimation based on principal component analysis. In: 2012 15th International IEEE Conference on Intelligent Transportation Systems, IEEE, pp 115–121

Dolega L, Pavlis M, Singleton A (2016) Estimating attractiveness, hierarchy and catchment area extents for a national set of retail centre agglomerations. Journal of Retailing and Consumer Services 28:78–90

Douglas DH, Peucker TK (1973) Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica: the international journal for geographic information and geovisualization 10(2):112–122

Drezner T, O'Kelly M, Drezner Z (2023) Multipurpose shopping trips and location. Annals of Operations Research 321(1-2):191–208

Eiter T, Mannila H (1994) Computing discrete fréchet distance. Technical Report CD-TR 94/64

Ester M, Kriegel HP, Sander J, et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd, pp 226–231

Fanaee-T H, Gama J (2016) Event detection from traffic tensors: A hybrid model. Neurocomputing 203:22–33

Fattore U, Liebsch M, Brik B, et al (2020) Automec: Lstm-based user mobility prediction for service management in distributed mec resources. In: Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp 155–159

Feng Z, Zhu Y (2016) A survey on trajectory data mining: Techniques and applications. IEEE Access 4:2056–2067

Gerber MS (2014) Predicting crime using twitter and kernel density estimation. Decision Support Systems 61:115–125

Google (Accessed on Mar 18, 2024a) Google cloud services. URL https://cloud.google.com/?hl=en

Google (Accessed on Mar 18, 2024b) Google cloud services. URL https://cloud.google.com/dataflow/

Google (Accessed on Mar 18, 2024c) Google cloud services. URL https://cloud.google.com/bigquery/

Google (Accessed on Mar 18, 2024d) Google cloud services. URL https://cloud.google.com/vertex-ai/

Han SY, Tsou MH, Knaap E, et al (2019) How do cities flow in an emergency? tracing human mobility patterns during a natural disaster with big data and geospatial data science. Urban Science 3(2):51

Hou J, Zhao H, Zhao X, et al (2016) Predicting mobile users' behaviors and locations using dynamic bayesian networks. Journal of Management Analytics 3(3):191–205

Huang H, Cheng Y, Weibel R (2019) Transport mode detection based on mobile phone network data: A systematic review. Transportation Research Part C: Emerging Technologies 101:297–312

Islam MA, Mohammad MM, Das SSS, et al (2022) A survey on deep learning based point-of-interest (poi) recommendations. Neurocomputing 472:306–325

Kohání M (2012) Exact approach to the tariff zones design problem in public transport. In: Proceedings of the International Conference Mathematical Methods in Economics

Kong D, Wu F (2018) Hst-lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In: IJCAI, pp 2341–2347

Kraemer MU, Yang CH, Gutierrez B, et al (2020) The effect of human mobility and control measures on the covid-19 epidemic in china. Science 368(6490):493–497

Krishnakumari P, van Lint H, Djukic T, et al (2019) A data driven method for od matrix estimation. Transportation Research Procedia 38:139–159

Liao TW (2005) Clustering of time series data—a survey. Pattern recognition 38(11):1857–1874

Liu Y, Pei A, Wang F, et al (2021) An attention-based category-aware gru model for the next poi recommendation. International Journal of Intelligent Systems 36(7):3174–3189

Lv Q, Qiao Y, Ansari N, et al (2016) Big data driven hidden markov model based individual mobility prediction at points of interest. IEEE Transactions on Vehicular Technology 66(6):5204–5216

Ma Z, Zhang P (2022) Individual mobility prediction review: Data, problem, method and application. Multimodal transportation 1(1):100,002

Magdy N, Sakr MA, Mostafa T, et al (2015) Review on trajectory similarity measures. In: 2015 IEEE seventh international conference on Intelligent Computing and Information Systems (ICICIS), IEEE, pp 613–619

Mahrez Z, Sabir E, Badidi E, et al (2021) Smart urban mobility: When mobility systems meet smart data. IEEE Transactions on Intelligent Transportation Systems 23(7):6222–6239

Microsoft (Accessed on Mar 18, 2024) Microsoft power bi. URL https://www.microsoft.com/en-us/power-platform/products/power-bi/

Moreira-Matias L, Gama J, Ferreira M, et al (2016) Time-evolving od matrix estimation using high-speed gps data streams. Expert systems with Applications 44:275–288

Ou J, Lu J, Xia J, et al (2019) Learn, assign, and search: real-time estimation of dynamic origin-destination flows using machine learning algorithms. IEEE Access 7:26,967–26,983

Ouyang X, Zhang C, Zhou P, et al (2016) Deepspace: An online deep learning framework for mobile big data to understand human mobility patterns. arXiv preprint arXiv:161007009

Pedregosa F, Varoquaux G, Gramfort A, et al (2011) Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12:2825–2830

Pratt MD, Wright JA, Cockings S, et al (2014) Delineating retail conurbations: A rules-based algorithmic approach. Journal of Retailing and Consumer Services 21(5):667–675

Ramer U (1972) An iterative procedure for the polygonal approximation of plane curves. Computer graphics and image processing 1(3):244–256

Shaji N, Andrade T, Ribeiro RP, et al (2022) Study on correlation between vehicle emissions and air quality in porto. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, pp 181–196

Solé-Ribalta A, Gómez S, Arenas A (2016) A model to identify urban traffic congestion hotspots in complex networks. Royal Society open science 3(10):160,098

Steenbruggen J, Tranos E, Nijkamp P (2015) Data from mobile phone operators: A tool for smarter cities? Telecommunications Policy 39(3-4):335–346

Sun F, Dubey A, White J (2017) Dxnat—deep neural networks for explaining non-recurring traffic congestion. In: 2017 IEEE International Conference on Big Data (Big Data), IEEE, pp 2141–2150

Vlachos M, Gunopoulos D, Kollios G (2002) Discovering similar multidimensional trajectories. In: Proceedings of the 18th International Conference on Data Engineering. IEEE Computer Society, Washington, DC, USA, ICDE '02, pp 673–, URL http://dl.acm.org/citation.cfm?id=876875.878994

Wang H, Zeng S, Li Y, et al (2020a) Predictability and prediction of human mobility based on application-collected location data. IEEE Transactions on Mobile Computing 20(7):2457–2472

Wang S, Cao J, Philip SY (2020b) Deep learning for spatio-temporal data mining: A survey. IEEE transactions on knowledge and data engineering 34(8):3681–3700

Xu Y, Xue J, Park S, et al (2021) Towards a multidimensional view of tourist mobility patterns in cities: A mobile phone data perspective. Computers, Environment and urban systems 86:101,593

Ye Y, Zheng Y, Chen Y, et al (2009) Mining individual life pattern based on location history. In: Mobile Data Management: Systems, Services and Middleware, 2009. MDM'09. Tenth International Conference on, IEEE, pp 1–10

Yuan G, Sun P, Zhao J, et al (2017) A review of moving object trajectory clustering algorithms. Artificial Intelligence Review 47:123–144

Yue M, Li Y, Yang H, et al (2019) Detect: Deep trajectory clustering for mobility-behavior analysis. In: 2019 IEEE International Conference on Big Data (Big Data), IEEE, pp 988–997

Zheng Y (2015) Trajectory data mining: an overview. ACM Transactions on Intelligent Systems and Technology (TIST) 6(3):1–41

Zheng Y, Xie X, Ma WY (2010) Geolife: A collaborative social networking service among user, location and trajectory. IEEE Data Eng Bull 33(2):32–39

Zhu L, Yu FR, Wang Y, et al (2018) Big data analytics in intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems 20(1):383–398