Self-Adapting Trajectory Segmentation

Agnese Bonavita
Scuola Normale Superiore
Pisa, Italy
agnese.bonavita@sns.it

Riccardo Guidotti
University of Pisa
Pisa, Italy
riccardo.guidotti@di.unipi.it

Mirco Nanni
ISTI-CNR, Pisa
Pisa, Italy
mirco.nanni@isti.cnr.it

ABSTRACT

Identifying the portions of trajectory data where movement ends and a significant stop starts is a basic, yet fundamental task that can affect the quality of any mobility analytics process. Most of the many existing solutions adopted by researchers and practitioners are simply based on fixed spatial and temporal thresholds stating when the moving object remained still for a significant amount of time, yet such thresholds remain as static parameters for the user to guess. In this work we study the trajectory segmentation from a multi-granularity perspective, looking for a better understanding of the problem and for an automatic, parameter-free and user-adaptive solution that flexibly adjusts the segmentation criteria to the specific user under study. Experiments over real data and comparison against simple competitors show that the flexibility of the proposed method has a positive impact on results.

KEYWORDS

Mobility Data Mining, Segmentation, User Modeling

1 INTRODUCTION

Thanks to the wide diffusion of localization technologies and mobile services based on the positioning of users and devices, the availability of mobility traces is increasing fast in several application domains. Location-based services provided through smartphones are nowadays extremely popular, from nearby restaurant suggestions to travel assistants, and in the near future all circulating vehicles will be equipped with localization capabilities for their continuous monitoring. The vast amounts of data that this trend leads to produce and collect open the door to several opportunities of converting them into better services, economical returns, more sustainable cities, improved living conditions, etc. All this starts from appropriate mobility analysis operations able to extract from raw data usable and useful information, such as deeper domain knowledge, patterns, models and forecasts.

In mobility analytics one of the fundamental concepts is *movement*, meaning with that the part of mobility data that describes a transfer from one place where the individual (or the object) was staying, to another one were the user will stop. Identifying movements in the raw stream of positions, for instance the continuous flow of GPS traces of a vehicle, is essential yet non-trivial. While it is simple to define a *stop* in geometrical terms, it is much less clear how to define *significant stops*, i.e. stops that might have some meaning for the user (for instance, stopping to do some activity before leaving), as opposed to stops that are simply incidental (for instance, due to a small traffic jam).

Practitioners in the mobility analytics domain defined several simple strategies to select stops in the mobility data stream (a brief account of literature on this topic is provided in the next section), each of them apparently capturing well some specific concept or some application-specific idea of stop. For instance, some solutions simply identify the moments where the object did not move, based on some thresholds, while others select the stops that have a duration compatible with some specific task, for instance discarding stops at a supermarket if their duration is physically too short to be able to enter, buy and exit. However, most existing solutions suffer from two important limitations: (i) they are based on critical thresholds that the user needs to choose accurately, and in most cases it is difficult to understand what value is the best; (ii) such thresholds are global, i.e. the same threshold value applies to all the moving individuals, irrespective of any distinctive characteristics they might have. The reason of the latter is that, while an overall evaluation might be performed to guide the choice of a global threshold, doing that separately for each individual might be impossible if their number is huge.

In this work we try to overcome the limitations highlighted above, providing a general methodology that inspects the mobility of the individual, and identifies segmentation thresholds that apparently match her mobility features. The process allows to get rid of any input parameter, adapts thresholds to each single individual and, most importantly, is completely automatic, thus applicable to large pools of users.

The paper is organized as follows: Section 2 discusses the related works and how our proposal differs from existing solutions; Section 3 provides some preliminary definitions; Section 4 defines the problem we want to tackle; Section 5 introduces our proposed method to solve the problem; Section 6 defines some evaluation measures to quantify the quality of a segmentation; Section 7 provides empirical quantitative and qualitative evaluations of results, also comparing against a few baselines; finally, Section 8 closes the paper with some conclusions and pointers towards future developments.

2 RELATED WORK

Segmentation is a technique for decomposing a given sequence into homogeneous and possibly meaningful pieces, or *segments* such that the data in each segment describe a simple event or structure. Segmentation methods are widely used for extracting structures from sequences, and are applied in a large variety of contexts [22]: time series [4, 9], genomic sequences [15, 17, 18], and text [12], to cite a few.

The segmentation of human trajectories is a very valuable task as it enables the development of mobility data models [7, 19] and applications like carpooling [6], or trajectory prediction [24]. Various simple approaches are currently adopted in practice. In [23] human trajectories are extracted adopting a predefined rule based on a pair of spatio-temporal parameters regulating the end of a trajectory and the start of the subsequent one. Similarly, in [8] the trajectory is divided into subsequent trips if the time interval of "nonmovement" exceeds a certain threshold. In [26] it is described a change-point-based segmentation approach for GPS

trajectories according to the transportation means adopting a universal threshold for determining whether a segment is "walk" or "nonwalk". The work in [3] presents a theoretical framework that computes an optimal segmentation by using several criteria (e.g., speed, direction, location disk) that are satisfied in each partition, thus making the approach local, and applied computational geometry methods. However, their methods are general and not clearly applicable to the human trajectory context, where a trip can be complex and not show the geometrical/movement uniformity the methods look for. Finally, each criterion corresponds to thresholds that the user must set, without clear guidelines on how to choose them.

The authors of [25] segment the trajectories in two steps. The first segmentation is performed by means of simple policies with respect to temporal and/or spatial predefined constraints. Then, the trajectories are divided into *stops* and *moves* observing variations of the speed of the object. If the variations of the speed is below a speed threshold and there is a sufficient number of observations, then the portion of trajectory is annotated as a stop. The speed threshold is not general but changes according to the user behavior and also to the surrounding of the stop. In [20] is defined a measure of the density of the points in the neighbourhood of each trajectory point, the Spatio-Temporal Kernel Window (STKW) statistics. To determine the start and end points of segments, the algorithm looks for maximal changes in STKW values. The focus of the approach is on capturing changes of transportation mode, including stops, which are simply points with low speed.

In addition to those mentioned above, several other solutions to the trajectory segmentation problem have been proposed in literature, yet with objectives different from ours. For example, cost-function based strategies were presented in [11][10], while clustering-based ones are introduced in [13] [14]. All these approaches are focused on splitting a movement into homogeneous parts, rather than discovering significant stops, which is the purpose of this paper.

In this work we provide a segmentation method that, opposed to most of the approaches mentioned above, is not based on fixed space and/or time thresholds to be fixed by the user – this is the case, for instance, of [8, 23, 25, 26]. Instead, we aim to make the segmentation parameter-free and also adaptive to the single user's data, giving the opportunity to have different kinds of segmentation over different users. Also, our approach is complementary to the STKW-based one [20], as the latter aims to differentiate movements with different speed profiles, including stops as a particular example, while we focus on stop timing and try to understand which stops are actually significant (e.g. not too short) for the user. A similar work was proposed in [5]. Here the authors proposed a new approach called Octal Window Segmentation(OWS) for unsupervised trajectory segmentation. The intuition behind their approach is that when a moving object changes behavior, this shift may be detected using only its geolocation over time. So the work focuses on finding these changes only from the object's coordinates using interpolation methods to generate an error signal. This error signal is then used as a criterion to split the trajectories into sub-trajectories.

3 SETTING THE STAGE

We start by defining trajectory segmentation based on a spatial and a temporal threshold, in a way similar to standard approaches in literature.

Definition 3.1 (Individual trajectory). Given a user u, her *Individual Trajectory* T_u is the sequence of n points $T_u = \langle p_1, \ldots, p_n \rangle$ that describes her position in time, where each point $p \in T_u$ is defined as a triple $p = (p.x, p.y, p.t)$, representing its spatial coordinates x and y and the corresponding timestamp t. Moreover, points are in chronological order, i.e. $\forall 1 < i \leq n . p_{i-1}.t < p_i.t$.

Definition 3.2 (Pseudo-stop duration). Given an individual trajectory $T = \langle p_1, \ldots, p_n \rangle$ and a spatial threshold σ, the Pseudo-stop duration associated to point p_i is defined as $SD(T, i) = \min\{p_j.t - p_i.t | i < j \leq n \wedge d(p_i, p_j) > \sigma\}$, where d represents the geometrical Euclidean or geographical distance.

Notice that the last point p_n will have $SD(T, n) = \min \emptyset = \infty$.

Definition 3.3 (Segmented trajectory). Given a trajectory $T = \langle p_1, \ldots, p_n \rangle$, a spatial threshold σ and a temporal threshold τ, we define the (σ, τ)-segmentation of T as $T^{\sigma,\tau} = \langle S_1, \ldots, S_m \rangle$, such that:

(i) $\forall 1 \leq i \leq m . \exists 1 \leq s < e \leq n : S_i = \langle p_s, p_{s+1}, \ldots, p_e \rangle$
(ii) $\bigcup_{i=1}^{m} set(S_i) = set(T)$
(iii) $\forall 1 \leq i \leq m . \forall 1 \leq j \leq |S_i| : SD(S_i, j) > \tau \Leftrightarrow j = |S_i|$
(iv) $\forall 1 \leq i \leq m . S_i$ is maximal

where $set(I) = \{p \in I\}$.

Conditions (i) and (ii) imply that the segments of the segmented trajectory of T form a partitioning of the elements of T in the strictly mathematical sense. Moreover, conditions (iii) and (iv) state that all the points in a segment are movement points, i.e. their pseudo-stop duration is smaller than the given threshold, excepted the last point. Therefore, each point in T that has a high pseudo-stop duration will act as a split point, and corresponds to a distinct partition in $T^{\sigma,\tau}$.

4 PROBLEM FORMULATION

Existing trajectory segmentation methods assume that the same rules and the same parameters should apply to all moving objects. Since different objects can show very different movement characteristics, the above assumption leads to make choices that on average fit best the dataset, yet potentially making sub-optimal choices on single individuals.

Our objective is to overcome this limitation, making the segmentation process adaptive to the individual and taking into consideration her overall mobility. Our problem statement extends the traditional formulation of segmentation as a threshold-based operation, thus the core issue is to find good parameter values for each user.

Definition 4.1 (Individual cut threshold problem). Given an Individual Trajectory T_u, and a global spatial threshold σ, the problem is to identify the temporal threshold τ that yields the optimal segmentation $T^{\sigma,\tau}$.

Since the number of moving objects can be very large, the process must be completely automatized and require no human intervention. In Section 5 we will introduce a simple and effective approach to solve the problem and thus find a suitable value of τ for each user. In addition, some basic guidelines to choose a value for the global spatial parameter will be provided.

5 PROPOSED METHOD

The proposed solution to the individual cut threshold problem consists in fixing the spatial threshold to a global value (i.e. to be used for all users) and then in studying the segmentations that we

would obtain by applying different temporal thresholds. We will start describing the process for choosing the temporal threshold, which is the central part of the solution, and later discuss how the spatial one can be chosen.

5.1 Self-Adaptive Trajectory Segmentation

When very small values of τ are used, the segmentation obtained will contain a huge number of very short segments, till the extreme case where each point forms its own segment. As the threshold is increased, more and more segments will merge together, since some of the former splitting points will fall below τ. The process is expected to gradually enlarge the trajectory segments by first including simple slowdowns (i.e. not really stop points), then temporary stops (e.g. at traffic lights), and so on.

Our approach consists in (virtually) monitoring such progression, and detect the moment where an anomalous increase in the number of segments is observed, which represents a sort of *change of state*. This follows the same kind of idea adopted in various unsupervised classification contexts, such as the *knee method* for deciding the number k of clusters for the k-means algorithm, or analogous solutions to choose the radius for density-based clustering (e.g. DBScan).

In our solution, rather than relying on visual or similar heuristic criteria, we will base the threshold selection on a statistical test. In particular, we will adopt the Modified Thompson Tau Test [2] which, basically, checks whether a given value fits the distribution of the rest of the data or not. Since we look for anomalous values in a sequence, we apply the test iteratively, comparing each value $n(t)$ (the number of segments obtained with $\tau = t$) against the values $n(t')$ obtained for larger thresholds t'.

This process yields a set of thresholds that have an anomalous number of partitions as compared to the successive thresholds. Among them, we simply choose the smallest one, thus deciding to select the segments that emerge at the first *change of state*, also representing shorter and finer granularity movements.

The procedure, named ATS (self-Adaptive Trajectory Segmentation) is summarized in Algorithm 1. Step 3 collects the pseudo-stop durations SD of all the points i that make up the segment, and step 4 computes the frequency F of each value, basically representing the number of new segments obtained using that value as τ w.r.t. the previous smaller thresholds. In our implementation such frequency distribution is computed through smoothed histograms, grouping values into bins of 1-minute width. Figure 1(left) shows the frequency distribution of a sample trajectory, the vertical line corresponding to a possible cut point. The resulting set of segments obtained is described in Figure 1(right) in terms of segments duration. Finally, step 5 selects the frequency values that appear to be anomalous (based on the Modified Thompson Tau Test) w.r.t. the frequency of larger thresholds, and step 6 returns the earliest time threshold that has an anomalous frequency.

Computational complexity. The cost of Algorithm 1 is dominated by step 3, since the computation of each pseudo-stop duration (SD) could in principle require to scan all the remaining points of the individual trajectory, thus yielding a $O(n^2)$ cost, where n is the size of the individual trajectory. However, in practical applications the trajectory portion needed for each SD is relatively small, leading to a quasi-linear cost. The remaining parts of the algorithm can be realized a linear time, including the Modified Thompson Tau Test which can be computed for each points through incremental updates.

Algorithm 1: ATS(T, σ)

1 **Input:** Individual trajectory T, spatial threshold σ
2 **Output:** Cut threshold τ
3 $S = \langle\, SD(T, i) \mid 1 \le i \le |T| \,\rangle$;
4 F = frequency distribution of S values ($F(a) = |\{a \in S\}|$);
5 $C = \{t | t \in range(F) \wedge TT(F(t), \langle F(t') | t' > t \rangle) = true\}$;
  // $TT(a, B)$ = Modified Thompson Tau Test of a vs. set B
6 **return** min C

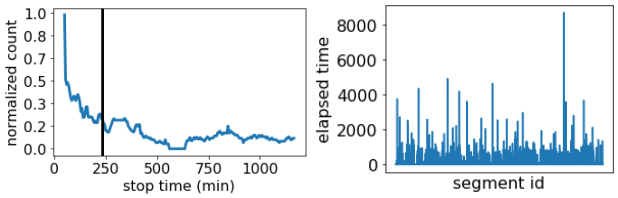

Figure 1: Frequency distribution of pseudo-stop durations for a user trajetory (left), and the durations of the segments obtained using a specific threshold to cut the trajectory (right). The threshold used corresponds to the vertical line on the left image.

5.2 Fixing the spatial threshold

In our approach, the threshold σ represents the minimum distance between two (consecutive) points that can be considered as a movement, and the temporal parameter is indeed measured as the time needed to make a movement. A simple way to fix its value is to adopt the minimum value that, according to the accuracy of our dataset, cannot be mistaken for a positioning error, for instance due to GPS uncertainty. In our experiments we adopt road vehicle GPS traces that are expected to have errors not larger than 10 meters, therefore we could fix $\sigma = 20$ (the worst case distance between two points that have the maximal error in opposite directions). We decided to slightly increase it to 50 in order to stay on the safe side, also to take into account that errors are slightly higher than average in urban centers, which is the application context where our experiments are performed. Since we do not have data source from other kind of transport (ships, planes, etc.) the selected threshold seems to meet our purposes. However, empirical results confirm that the value of the global parameter σ is not critical, as our approach shows a low sensitivity to it. For this reason, the value we chose in our experiments (50 meters) can be considered a good guess for generic vehicle GPS data. Other data sources with a higher spatial uncertainty might require larger values.

6 EVALUATION MEASURES

The reconstruction error generally used for evaluating segmentation problems [1] just measures how well each segment can be approximated with one value, and thus seems not to fit with trajectory segmentation. Therefore, similarly to clustering evaluation, we propose three internal evaluation measures [21]. Let T be the sequence of n points and $T_S = \langle S_1, \ldots, S_m \rangle$ its segmentation. We denote with $A_t = duration(T) = p_n.t - p_1.t$ the total elapsed time from the first point of $p_1 \in T$ to the last point $p_n \in T$, and $A_d = length(T) = \sum_{i=1}^{n-1} d(p_i, p_{i+1})$ the total distance covered by the trajectory, computed by considering every couple of subsequent points in T. Let $M_t = \sum_{S_i \in T_S} duration(S_i)$ be the

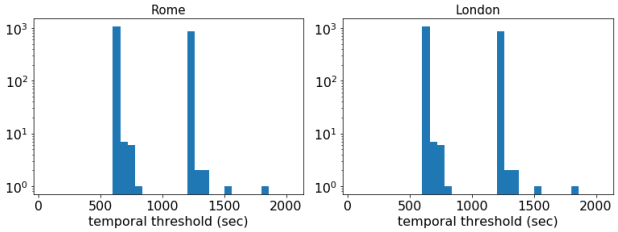

Figure 2: Time threshold distributions for trajectories in Rome and London. The peaks show the ideal thresholds to be set to build the trajectories.

sum of the segments' duration, i.e., the time spent driving, and $M_d = \sum_{S_i \in T_S} length(S_i)$ be the sum of the segments' length, i.e., the distance traveled. Then, we define the following measures:

- *time precision*: $TP = 1 - M_t/A_t$
- *distance coverage*: $DC = M_d/A_d$
- *mobility f-measure*: $MF_\beta = (1+\beta^2) \cdot TP \cdot DC / ((\beta^2 \cdot TP) + DC)$

All measures range from zero to one. The higher the value the better the result. The objective of these measures is to promote segmentations capturing long stops (*time precision*) yet also covering most of the overall distance (*distance coverage*). These two objectives are conflictual, since making stops longer reduces the number of points that contribute to the distance covered. The *mobility f-measure* accounts for both aspects simultaneously. In the experiments we adopt $\beta = 0.25$, which weighs *time precision* much higher than *distance coverage* by augmenting the relevance of missing precision in stop detection. The reason is that *i)* it is relatively easy to guarantee an high distance coverage, and *ii)* the main focus of the paper is on the temporal aspects of trajectory partitioning.

7 EXPERIMENTS

We experimented the proposed self-adaptive trajectory segmentation approach (ATS) described above over a real dataset of GPS vehicle traces. The results commented in the following refer to 2000 users of the area of Rome (Italy), and London (UK). The means and standard deviations of the sampling rate for the users analyzed are 12194.67 ± 22575.66 and 4385.76 ± 9359.14, for Rome and London respectively. The high values and their high variability is due to the presence of several long time gaps, typically due to parking periods.

In the following we first analyze the personal temporal thresholds returned by ATS, then we propose a quantitative and qualitative evaluation of the results for understanding the benefits of the novel method with respect to existing ones. We compare ATS against the trajectory segmentation method with fixed parameters proposed in [23] ($\text{FTS}_{\text{temp-thr}}$). Moreover, we adopt as baseline a random trajectory segmentation method that segments the sequence of points $T = \langle p_1, \ldots, p_n \rangle$ into m equal-length segments *(i)* with m randomly extracted between 2 and $n/2$ (RTS_1), or *(ii)* with m set to the number of segments returned by the proposed ATS method (RTS_2).

7.1 Self-Adaptive Temporal Threshold

We observe in Figure 2 the distribution of the time thresholds selected by ATS for each user (vertical axis represents value frequencies in log-scale).

method	$MF_{.25}$	TP	DC	$ratio_{\text{sr}}$	*#segms (avg ± std)*
ATS	**.951**	.951	.981	0.049	837.34±854.52
FTS_{120}	.925	**.996**	.456	**0.015**	592.26 ±652.78
FTS_{1200}	.948	.947	**.997**	0.053	746.28 ± 733.96
RTS_1	.279	.268	.722	0.700	2094.85± 2472.36
RTS_2	.124	.118	.877	0.883	899.59 ± 926.03

Table 1: Evaluation on Rome data. The first three columns show the measures adopted to test our new approach. The fourth one reports the ratio between the average sampling period of non-stop points over that of all points, and the last column is the number of segments.

method	MF_{25}	TP	DC	$ratio_{\text{sr}}$	*#segms (avg ± std)*
ATS	.955	.953	**.999**	0.047	433.915±513.715
FTS_{120}	**.958**	**.961**	.944	**0.040**	1131.829± 1431.810
FTS_{1200}	.952	.950	**.999**	0.050	359.545 ± 410.606
RTS_1	.267	.256	.695	1.007	2833.718 ± 4203.049
RTS_2	.035	.033	.958	1.008	445.645 ±527.969

Table 2: Evaluation on London data. The first three columns show the measures adopted to test our new approach. The fourth one reports the ratio between the average sampling period of non-stop points over that of all points, and the last column is the number of trajectories.

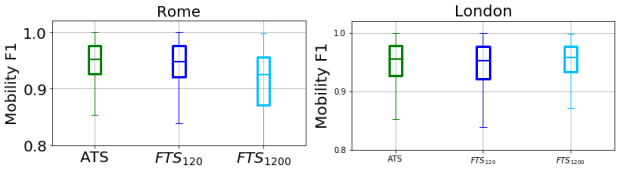

Figure 3: Boxplots for the $MF_{.25}$ results. On the Rome data ATS yields better results than the FTS solutions, while in London all three produce almost the same results. The variability of ATS results is consistently smaller than the other methods, which is a sign of robustness.

Although every user has her own mobility behavior with its own mix of regular and more erratic behaviours [16], we observe two clear peaks in the distributions for both Rome and London. This means that with respect to ATS we mainly recognize two different types of users regarding to the minimum duration of the stops. This supports the intuition behind our approach, namely to have a self-adaptive procedure selecting a personalized best temporal threshold for each user. Selecting one single threshold value for all the data might negatively affect the segmentation of some users, partitioning their trajectories either too much or too little. The first peak is at about 600 seconds (~ 10 minutes), while the second peak at 1200 seconds (~ 20 minutes). These values correspond to the temporal thresholds that the ATS procedure uses to cut each trajectory. There is also a minority of users having values outside the two peeks.

7.2 Comparison of Evaluation Measures

In this section we compare the proposed self-adaptive trajectory segmentation approach with the other methods taken into account. In Tables 1 and 2 we report the results obtained with all the

methods. The first three columns show the evaluation measures described above. The fourth column shows the ratio between the average sampling period of movement points (thus discarding the stop portions of the user's trajectory) and the average sampling period of the full trajectory, while in the last one the average number of segments with its standard deviation is given. In general, we can observe that the best results were obtained with the ATS and FTS methods, both for Rome and London. Analyzing the ratio (fourth column) we can see that values are low for both ATS and the FTS ones, meaning that the long stops are ignored (i.e. they are recognized as real stops) and just the short ones are considered. On the contrary, with the random approaches the ratio is bigger because the algorithm function evaluates all stops in the same way. Looking at the number of segments it is possible to note that FTS and ATS methods produce different quantities, especially the FTS_{120} result produces less segments in the Rome case and much more in London. About the last two approaches, the RTS_1 method works with a random number of segments, so it is normal that the final result differs from the others, while the RTS_2 takes as number of segments the same of the ATS approach so we aspect to achieve similar results.

For the evaluation measures we can see that our new approach reached the goal we expected, i.e., yielding a quality of results which is always comparable or higher than fixed-threshold approaches and more robust. Indeed, for both Rome and London the values obtained by ATS are compatible with the FTS results, even better in the $MF_{.25}$ for Rome and in the distance coverage for London. In particular, in the Rome example, having a high $MF_{.25}$ values means that also the time precision and the distance coverage are well correlated in a way that produce a satisfying result. If we see the FTS_{120} result we can note that the time precision is high but the distance coverage is very low because the algorithm builds short trajectories with few points. An analogous reasoning can be done analyzing the FTS_{1200} method which produces an excellent distance coverage score but a lower time precision. Our solution reaches a good balance, thanks to its self-adaptive characteristic that allows to control and correct the trajectory fragmentation, and all its evaluation measures are always either the best or the second best of the group.

To have a better understanding of the quality of our new approach, the distribution of $MF_{.25}$ values for the different approaches on the two datasets is shown in Figure 3 through a boxplot visualization. For the Rome case we can observe that with the ATS approach the median value is the highest (closest to 1) and the inter-quartile range is smaller than the other two, meaning that we have a smaller variabiliy and thus more robust results. The London case appears to be different, and the best $MF_{.25}$ values are obtained with the FTS_{1200}, with a median similar to ATS and a slightly narrower box. This leaves room for future improvements of our methodology.

7.3 Comparison of Segmentation Statistics

In the following we analyze other statistical indicators on the trajectory segments extracted by the various methods. The next plots want to show other significant features for the segmentation problem in order to compare their distribution and try to infer something more about the segmentation. In addiction discovering some hidden correlations between trajectory features and the segmentation approach could lead to a better understanding of the problem and highlight other relevant aspects. In Figure 4 we report the distributions of the average number of points per

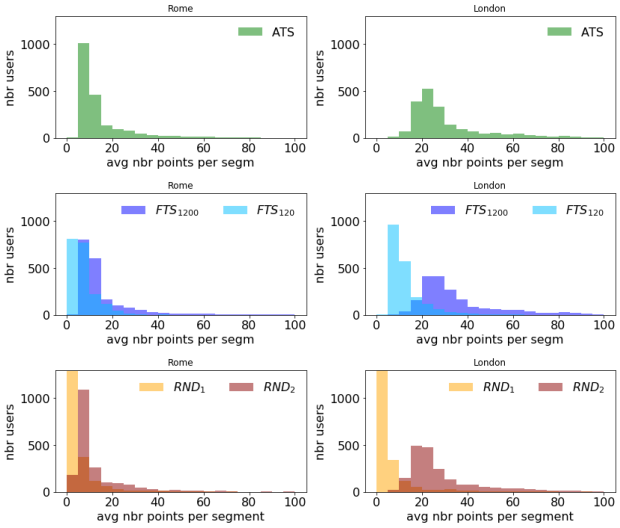

Figure 4: Distributions of average number of points per segment in Rome (left) and London (right).

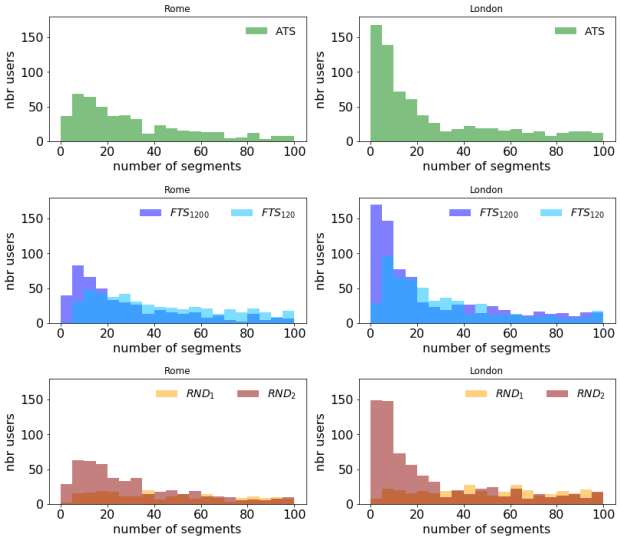

Figure 5: Distribution of the number of trajectory segments over Rome (left column) and London (right column) with each segmentation method (on the rows, grouped by family).

segment for Rome and London. For all methods, the majority of segments have less than 20 points, probably meaning that most of the trips take place within the city. However, in the distribution tails some long trajectories with more points emerge. We observe that the distribution peaks of ATS place somehow in between the peaks of the two FTS variants (though closer to FTS_{1200}, especially in London) thus finding a trade-off between them. Moreover we can see that London and Rome distributions are different: London has a wider distribution than Rome, meaning that the variety of trips is greater in London.

In Figure 5 are displayed the distributions of the average number of segments per user. In London most of the users have less than 20 trajectory segments. The peak of the distribution is between 5 and 10 segments. Between 30 and 100 segments the

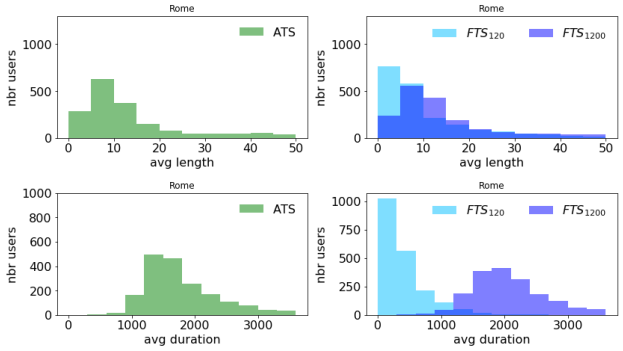

Figure 6: Distributions of the average length (top) and duration (bottom) for the trajectory segments returned by ATS (left) and FTS (right) for the area of Rome.

distribution remains stable at a small value larger than zero. In Rome we observe a similar result with a peak between 15 and 20 trajectories. Also in this case, the peak of ATS distribution tends to stay in the middle of the FTS ones.

In Figure 6 we compare the distribution of average length and average duration of the segments returned by ATS (left) and FTS (right) for the area of Rome. With the ATS method the peak value is around 10km, thus confirming that most of the trips are short, and likely to take place around the city. With the FTS methods the peak position depends on the temporal threshold imposed: with a threshold of 1200 seconds the average distance is similar to ATS, while with 120 seconds it becomes lower and close to 5 km. The results for the RTS methods are omitted, since their plots are very similar to the FTS ones. Also, the plots in London show exactly the same kind of behaviour observed on Rome.

In terms of segment duration, ATS yields a distribution with a peak around 1200 – 1500 seconds (~ 20 – 25 minutes). With the FTS methods the peaks change: for FTS_{120} the peak is around 500 seconds while for FTS_{1200} the peak is centered in 1800 seconds. Also in this case, the results on London are very similar and omitted here.

7.4 Case Study

In this section we show qualitatively on a case study the effectiveness of ATS with respect to FTS. In Figure 7 we report the segmentation returned by FTS_{1200} [23] (left) and by ATS (right), the user is travelling from south to north. FTS_{1200} [23] returns two trajectories (green and blue), while ATS returns three trajectories (green, orange and blue). The second line of plots report the inter-leaving time between consecutive GPS points. The colors match the trajectory segments, while stops are highlighted in red. We observe how ATS identifies the short stop of less than 15 minutes at the service area similarly to the subsequent longer stop. On the other hand, FTS_{1200} considers the first stop as part of the green trajectory. The map in the bottom line of Figure 7 shows the service area which is very close to the GPS points reported on the bottom right corner of the map. This case study highlights how various existing stops under a certain predefined threshold can be missed with a segmentation approach like FTS, while a more data-driven and self-adaptive method like ATS is able to take into account specific user behavior and return a better result.

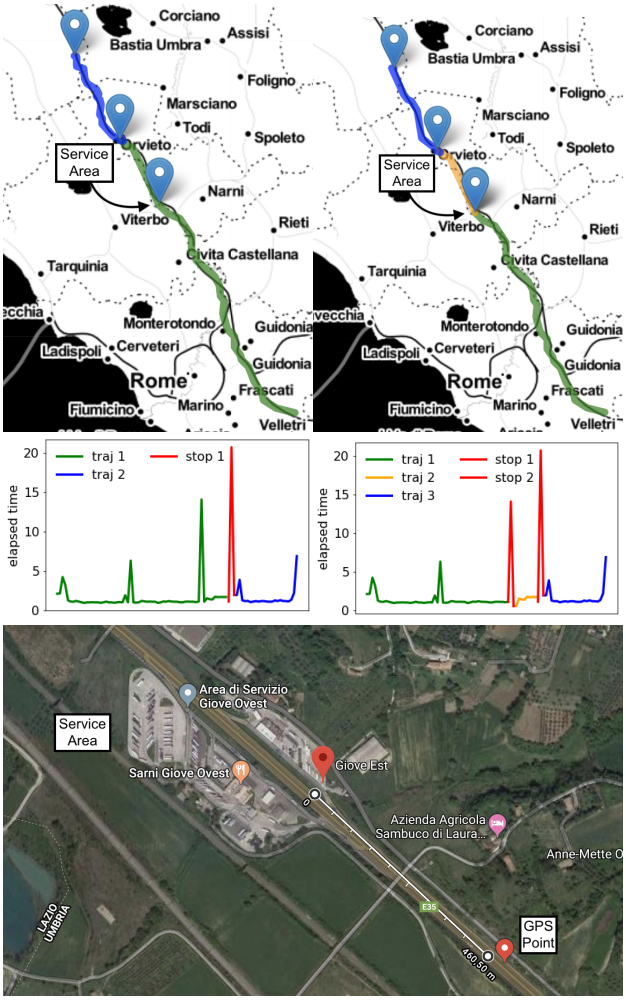

Figure 7: Trajectory segmentation returned by FTS_{1200} (left) and ATS (right). The user is travelling from South to North. *Top*: spatial representation showing the trajectory segments. *Center*: temporal segmentation showing the inter-leaving time between GPS points. *Bottom*: zoom on the service area highlighted in the top maps where the user probably stops for ~ 15 minutes. Best view in color.

8 CONCLUSION

The paper presented a user adaptive method for solving the trajectory segmentation problem, a very common and useful task in mobility data mining, especially in preprocessing phases. Though preliminary, the experiments show that it is possible to derive user-adaptive cut thresholds, improving the performances of the segmentation over less flexible solutions. This is an ongoing work, and several improvements are being explored. Among them, the future lines of research will aim to derive thresholds for trajectory segmentation which are not only user-adaptive, but also location-adaptive, thus considering the fact that a stop at different places might require time intervals of different duration to be considered a significant stay – and thus a trajectory cut point. Also, we will study the possibility of exploiting the context around the (moving) user, such as the mobility of other users and the geographical area surrounding the candidate stops.

ACKNOWLEDGMENTS

This work is partially supported by the European Community H2020 programme under the funding scheme *Track &Know* (Big Data for Mobility Tracking Knowledge Extraction in Urban Areas), G.A. 780754, https://trackandknowproject.eu/.

REFERENCES

[1] Ella Bingham. 2010. Finding segmentations of sequences. In *Inductive Databases and Constraint-Based Data Mining*. Springer, 177–197.
[2] Ronald Bremer. 1995. *Outliers in statistical data*. Taylor & Francis.
[3] Maike Buchin et al. 2010. An algorithmic framework for segmenting trajectories based on spatio-temporal criteria. In *SIGSPATIAL*. ACM, 202–211.
[4] Harmen J Bussemaker et al. 2000. Regulatory element detection using a probabilistic segmentation model. In *Proceedings of the International Conference on Intelligent Systems for Molecular Biology*. 67–74.
[5] Mohammad Etemad et al. 2019. A Trajectory Segmentation Algorithm Based on Interpolation-based Change Detection Strategies. In *EDBT/ICDT Workshops*.
[6] Riccardo Guidotti et al. 2017. Never drive alone: Boosting carpooling with network analysis. *IS* 64 (2017), 237–257.
[7] Riccardo Guidotti et al. 2017. There's a path for everyone: A data-driven personal model reproducing mobility agendas. In *DSAA*. IEEE, 303–312.
[8] Sini Guo et al. 2018. GPS trajectory data segmentation based on probabilistic logic. *International Journal of Approximate Reasoning* 103 (2018), 227–247.
[9] Johan Himberg et al. 2001. Time series segmentation for context recognition in mobile devices. In *ICDM*. IEEE, 203–210.
[10] Amílcar Soares Júnior et al. 2015. GRASP-UTS: an algorithm for unsupervised trajectory segmentation. *International Journal of Geographical Information Science* 29, 1 (2015), 46–68.
[11] Amílcar Soares Júnior et al. 2018. A semi-supervised approach for the semantic segmentation of trajectories. In *19th IEEE International Conference on Mobile Data Management (MDM)*. 145–154.
[12] Victor Lavrenko et al. 2000. Mining of concurrent text and time series. In *KDD Workshop on Text Mining*, Vol. 2000. 37–44.
[13] Jae-Gil Lee et al. 2007. Trajectory Clustering: A Partition-and-Group Framework. In *ACM SIGMOD*. ACM, 593–604.
[14] Luis Leiva and Enrique Vidal. 2013. Warped K-Means: An algorithm to cluster sequentially-distributed data. *Information Sciences* 237 (07 2013), 196–210.
[15] Wentian Li. 2001. DNA Segmentation as a Model Selection Process. In *Proceedings of the Fifth Annual International Conference on Computational Biology (RECOMB '01)*. ACM, 204–210.
[16] Luca Pappalardo et al. 2015. Returners and explorers dichotomy in human mobility. *Nature communications* 6 (2015), 8166.
[17] Adam Pavlíček et al. 2002. A compact view of isochores in the draft human genome sequence. *FEBS letters* 511, 1-3 (2002), 165–169.
[18] Vasily E Ramensky et al. 2000. DNA segmentation through the Bayesian approach. *Journal of Computational Biology* 7, 1-2 (2000), 215–231.
[19] Salvatore Rinzivillo et al. 2014. The purpose of motion: Learning activities from individual mobility networks. In *DSAA*. IEEE, 312–318.
[20] Katarzyna Siła-Nowicka et al. 2016. Analysis of human mobility patterns from GPS trajectories and contextual information. *IJGIS* 30, 5 (2016), 881–906.
[21] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2018. *Introduction to data mining*. Pearson Education India.
[22] Evimaria Terzi and Panayiotis Tsaparas. 2006. Efficient algorithms for sequence segmentation. In *SDM*. SIAM, 316–327.
[23] Roberto Trasarti et al. 2011. Mining mobility user profiles for car pooling. In *KDD*. ACM, 1190–1198.
[24] Roberto Trasarti et al. 2017. Myway: Location prediction via mobility profiling. *IS* 64 (2017), 350–367.
[25] Zhixian Yan et al. 2013. Semantic trajectories: Mobility data computation and annotation. *ACM TIST* 4, 3 (2013), 49.
[26] Yu Zheng et al. 2011. Recommending friends and locations based on individual location history. *ACM Transactions on the Web* 5, 1 (2011), 5.