Trip Summaries: Spatio-temporal Mobility Signatures for People Movement

IRL
IBM Research, India
@in.ibm.com

ABSTRACT

1. INTRODUCTION

- Movement summaries are good for many applications – Next-loc prediction, anomaly detection and for storing Customer Profiles
- Currently there is no good way to describe human movement summaries and no good way to store these summaries such they are beneficial to higher level applications
- Existing techniques to summarize trajectories do not abstract creating movement summaries. They do not account for what counts as a meaningful trip of a person, they do not accurately model the similarity metrics for trips of people.
- We propose to crate a layered-summary for user trajectories
- Contributions:
  1. Propose an abstraction to store human trajectories – not too sparse such as textual information, and not too fine that we store all the trajectories
  2. Propose algorithms and models for trip preprocessing, trip similarity and trip clustering such that a meaningful human trips are represented by representative trips
  3. Compare exhaustively with existing trajectory algorithms to identify the considerations for human trip summarization
  4. Evaluate across different data-sets and show that our technique are x% superior

2. MOTIVATION

Consider few toy examples

- Toy scenario to show DTW/LCSS gives wrong measurement (triangular inequality in a 3 traj clustering)
- Toy scenario to show DTW/LCSS gives wrong measurement (triangular inequality in a 3 traj clustering)

Here are the aspects of human trips that are not taken care by related work

- Similarity measures
  1. The similarity of human trips are significantly affected by the Origin and the Destination of the trips since in most of the cases *SHOW THIS* user intends to move between OD for a specific purpose. Hence, the similarity metric has to account explicitly for OD
  2. Similarity measures that are not metrics cannot be used for clustering since they invariably lead to inconsistent results when multiple pair of trajectories are compared against each other (such as in clustering) *SHOW THIS*
  3. Looking into spatio-temporal similarity accounting for the time of the user movement leads to ambigious definitions. Are simiarlities between two trajectories which go on the same path 10 mins part same as that of two trajectories which go 1 km apart at the same time? Hence, a hierarchical approach for solving this problem is suggested where temporal (based on time-of-the-day) and spatial similarities are separated.
  4. For human trip movement, denoising can be done prior to trajectory processing, intead of making the similarity measures resilient to noise; such measures are computationally expensive, and do not yield accurate results
- Summarization
  1. There may be many queries for asked about human trips
     - Customer Profile: Give summary of a person's trips in a region – both in space or time
     - Next-Location Profile: What are the most probable trajectories to find a person between time x to time y (optional: given that the person is at location L at time t)
     - Alerts: Alert when a customer is moving in an anomalous way

2. Some of them we do not account for (yet, but we can)
    - Insurance Usecase: Give summary of a person's trips that have good or bad speed profiles – immaterial of the space or time

3. FORMULATION

3.1 Preprocessing

Trajectory Representation.
We represent a trajectory as a function $f(t)$ that maps a time t to a space (say, latitude and longitude). For simplicity, we assume $t = [0, 1]$; $t = 0$ is the start, and $t = 1$ is the end of the trajectory.

Traj Segmentation.

Traj Denoising.
TODO

3.2 Similarity metric

A large number of trajectory similarity metrics have been proposed for various types of applications. A taxonomy of trajectory similarity metrics is discussed in Table 1. Similarity metric for summarizing human movement should consider the below aspects

- Importance to Origins and Destinations: Hurricanes and other physical effects (?) are generally governed by physical laws and computing similarity might have to account for different effects. However, movement of a people is generally associated with an intention (such as commuting to work) of moving from an origin point to a destination point. Hence, a reasonable summary of a person's movement accounts for end-to-end trips that she takes. Currently, there is no similarity metric that explicitly considers the end points of the trajectory. Ours does. For example, SWARM does not consider explicitly consider the end points of the trajectory. Hence, it might wrongly categorize movement on similar roads – even with varying end points – into the same cluster. *Show a toy scenario that shows how SWARM has mistook two different o-d clusters to be in the same cluster*
- Metric: Clustering trajectories using standard clustering algorithms require the distance function between two trajectories to be a mathematical metric. In the current literature, only a few functions are metrics [**?**, **?**]. Others, which mostly take care of artifacts of sampling, are shown to be non-metrics. Using such functions, which violate properties like triangular inequality, in clustering can lead to unforeseen results. *Show a figure where tri inequality is violated in actual sense*
- Sampling artifacts: Is time stretching and shrinking important? Are sampling points important? Ours is better because we consider human specific movement where direction and OD are important. While sampling frequency and alignment of samples is required, more or less all traces can be preprocessed after good samples have been taken. So, it really doesnt make sense to consider the effect of timing of samples (like DTW) while designing the clustering algorithm. *Vinay: Show a toy scenario where DTW goes wrong*

We now define a similarity metric that considers provides a measure of the distance between meaningful trips of a person.

Weighted LP-Norm.
We now propose an similarity metric, which is an extension to LP-Norm, for reasoning about the proximity of two clusters. Let $w(t)$ be a weighting function at time t such that

$$w(t) \geq 0, \qquad \forall t, \tag{1}$$

$$\int_t w(t)\,\mathrm{d}t = 1. \tag{2}$$

We describe the Weighted LP-Norm Similarity metric between two trips $f_1(t)$ and $f_2(t)$ as

$$\mathrm{WLP}(f_1(t), f_2(t)) = \left[\int_t w(t)\,(f_1(t) - f_2(t))^2\,\mathrm{d}t\right]^{\frac{1}{2}}. \tag{3}$$

It can be shown that $\mathrm{WLP}(.)$ since it a weighted combination of LP-Norm (which is a metric) *Vinay: Karthik, please comment* . We will now show that WLP is a powerful abstraction to get meaningful similarity metrics

Origin-Destination.
In human movement, similar trips are usually between similar origin and destination regions. For example, the summary of a vast majority of working population are their trajectories between home and office. Hence, the similarity function should provide greater importance to OD. We define *Weighted LP-Norm for OD* as

$$w_{\mathrm{OD}}(t, c, r) = \begin{cases} \frac{c}{r} & \text{if } t \leq \frac{r}{2} \text{ or } t > (1 - \frac{r}{2}), \\ \frac{1 - \frac{c}{r}}{(1-r)} & \text{otherwise,} \end{cases} \tag{4}$$

where r and c denote the parameters define the weight assigned to the origin and destination stertches when compared to the intermediate stretch. Here r defines the fraction of the stretch from origin or towards destination which has to be given a prominence ($r = [0, 1]$). And, c is the weight assignment factor at O and D stretches ($c = [0, 1]$). If $c > 1 - r$, then the origin and destination stretches of length $\frac{r}{2}$, will be given a higher weight than the intermediate stretch (of length $1 - r$)

Sim Measure	Is Metric	Type	Sen. to sample noise	OD Cognizant	Computatio
LP Norm	Yes	Sampling Sensitive	No	No	O(N
DTW/LCSS/EDW/EDW With real sequences	No	Sampling Sensitive	Yes	No	O(n^2
EDWP	Yes	Sampling Sensitive	Yes	No	??
LP Norm with Interpolation	Yes	Shape Sensitive	No	No	O(Num sa
ODSim (Ours)	Yes	Shape sensitive	No	No	O(Num sa

Table 1: Taxonomy of Similarity Measures

Vinay: Karthik, is there a better way to do the above? We need some heavy damping function which is thick at O and D, and bleak elsewhere

Let $x_i \geq 0$ represent the "color of grid i".

Let $N(i) = \{$All grids that are neighbors of i$\}$.

Let C be the "conflicting set". A tuple (i, j) is added to the conflicting set C if two grids i and j have a *LAC separating line* between them.

Let y_i be an indicator variable to say if the grid has same color as atleast one of its neighbor.

We formulate an optimization problem to find the color for each grid as below:

$$\text{Min} \sum_i y_i \tag{5}$$

such that:

$$x_i \neq x_j, \quad \forall (i, j) \in C \tag{6}$$

$$y_i = \begin{cases} 1 & \text{if } \forall j = N(i), x_i = x_j, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

3.3 Clustering

- Number of clusters in the summary: Since each person's movement is unique(?), the number of clusters and the number of trajectories expected in each clusters are not constant. Hence, standard mechanisms such as k-means clustering cannot be directly applied to summarize. Even after the clusters are found out, we need to see if this cluster represents points that have meaningful end-to-end trips. *This is not coming out good. Need to think* We need good OD + we need trajectories that are not far away in the middle. One way to design a sim metric that is congnizant of: (1) origins, destinations, directionality and (2) the maximum separation between the trajectories. However, such a sim metric will introduce other artifacts (say, by classifying far away trajectories into same cluster). Hence we consider an approach where we decouple by considering O/D and direction in the sim metric, and then design an algorithm to select optimal number of clusters by looking at the maximum intra-cluster separation.

3.4 Cluster Representative

3.5 Intra-cluster Movement Pattern

4. EVALUATION AND ANALYSIS

4.1 Datasets

- Beijing
- Xtify (?)
- Movebank/Starkey
- Singapore

4.2 Clustering Effectiveness

- Silhoutte is better in OD/LP vs anything else (Figure 2). Median Sil Coeff is x% better than in DTW, y% better than in Swarm, z% better than traj clus
- Num clusters with large number of trajs in it is better (Figure 13 and 11). Clearly, we summarize a human's movement better

Silhouette Coefficient.

We show that the proposed algorithm clusters significantly better by using Silhouette Coefficient (SC); a standard metric that shows the effectiveness of clustering. SC is based on the cohesion and the separation of clusters formed. The cohesion (*a(x)*) is defined as the average distance of x to all other vectors in the same cluster. The separation (*b(x)*) is defined as the minimum of the average distances of x to the vectors in other clusters. Further, the silhouette coefficient of a data point is defined as

$$s(x) = \frac{b(x) - a(x)}{max(a(x), b(x))} \tag{8}$$

The total silhouette coefficient of the dataset is the average over all the points given by

$$SC = \frac{1}{N} \sum_{i=1}^{N} s(x) \tag{9}$$

Ideally, SC is between [-1,1], where values closer to 1 representing better formed clusters. Figure **??** shows that our algorithm provides x%... improvement over SWARM. DTW performs the worst where the clustering .... Figure 2 shows the silhoutte coefficient for clusters for individual users. In more than *90%* of the scenarios, OD-based clustering outperforms DTW by *some percentage* , and outperforms SWARM by *some percentage* .

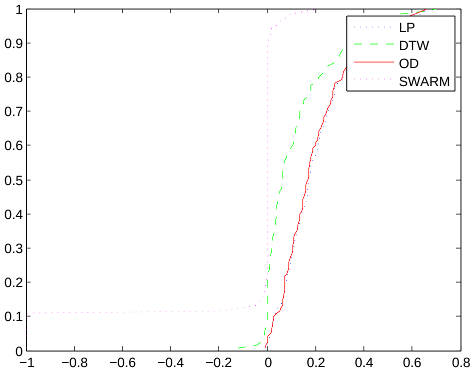


Figure 1: Comparison of silhouette coefficients for different schemes: OD- and LP-based clustering outperform existing mechanism. We outperform clusters found by DTW by order(s) of magnitude

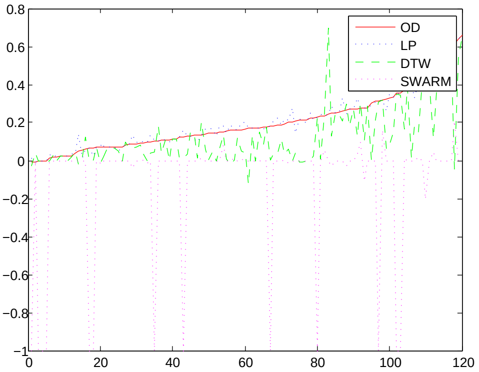


Figure 2: Comparison of silhouette coefficients for different schemes: OD- and LP-based clustering outperform existing mechanism. We outperform clusters found by DTW by order(s) of magnitude

4.3 Analysis of OD-clustering

- Interpolation: Spline vs linear on earth coordinate system. Illeffects of spline. *Show that figure where spline took a 2km path unnecessarily*

- Why hierarchical: We really dont know the number of clusters. We need to iterate over each k and then find out the optimal k. The time complexity of running k-means for 100's of ks and then finding out the optimal k is much more expensive than doing 1-shot hierarchical clustering and finding a good point to cut. *Show the time graph for running k-means vs hierarchical Show complexity*

Variance in origins and destinations.

Figure 4 and 6 show that OD- and LP-based clustering schemes have relatively low OD differences.However, DTW and SWARM have very less trajectories per cluster giving rise to low SSW values.

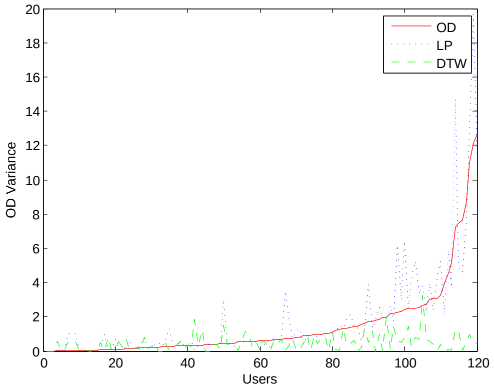
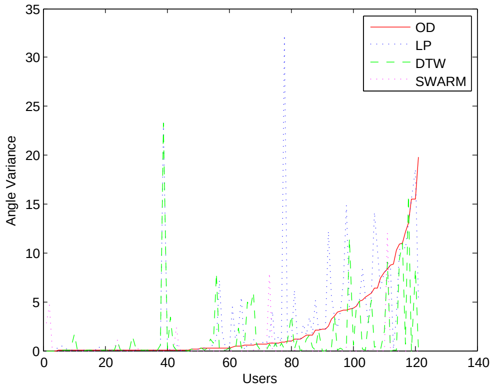


Figure 3: Comparison of OD variance for optimal cluster for LP-DTW-OD



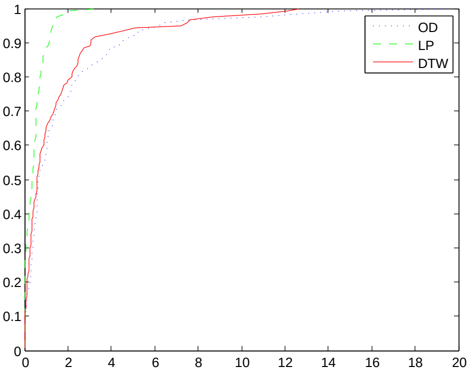
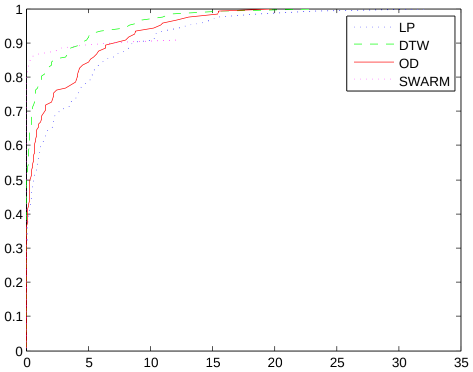


Figure 5: Comparison of angle variance for optimal cluster for LP-DTW-OD

Figure 4:



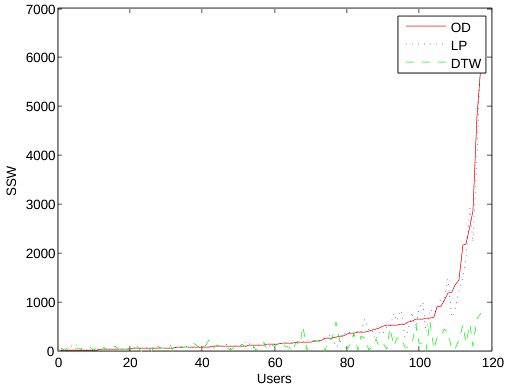


Figure 6: CDF of angle for optimal cluster for LP-DTW-OD

Figure 7: Comparison of SSW for optimal cluster for LP-DTW-OD

4.4 Evaluation of Similarity Effectiveness

4.5 Sum of Squares Within Clusters

(SWARM has very large SSW and OD variance values for some users, which skews the plots. Have Left it out for now, will have to change plots to logscale)

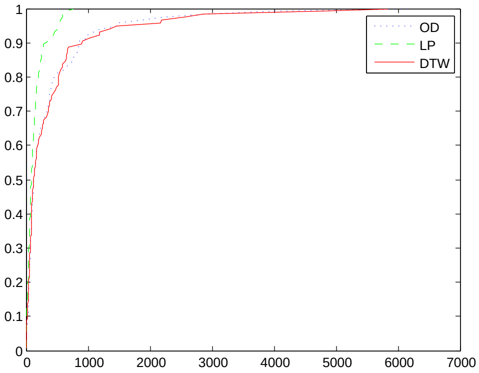


Figure 8: CDF of SSW for optimal cluster for LP-DTW-OD

Definition.

The sum of squares within clusters is defined as

$$SSW = \sum_{i=1}^{n} (|x_i - c_i|)^2 \tag{10}$$

where x_i is a data point and c_i is the mean of the cluster it belongs to. In the case of Trajectory analysis, we consider the pointwise summation of the Haversine distance between the trajectory and the mean trajectory of that cluster, over all the sample points.

Some Key Points for SSW Curves.

The reason behind the low SSW values for DTW method is that it does not discover all the trajectories in the final clusters. This is because the point in the dendrogram where the condition for reporting final clusters is satisfied, is very low when DTW is used as a similarity metric.

4.6 Origin-Destination Variance

4.7 Average Trajectories Per Cluster

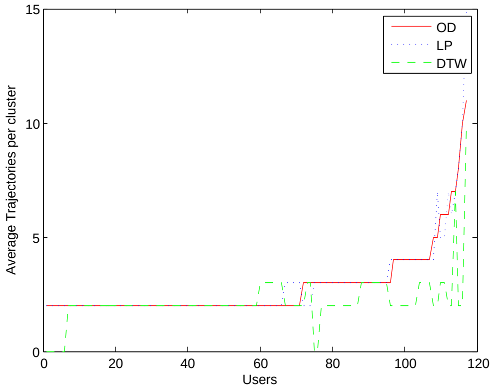
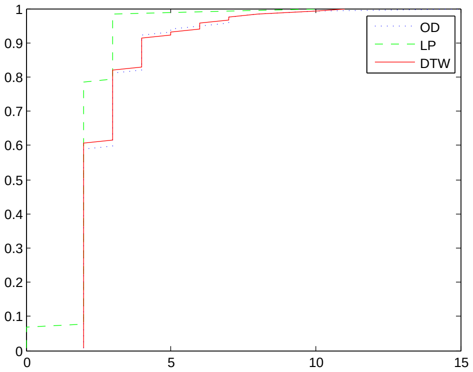


Figure 9: Average trajectories per cluster for LP-DTW-OD



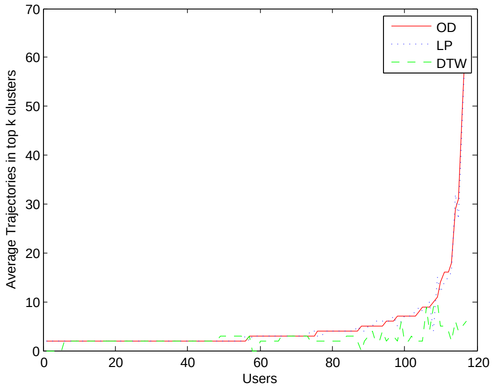
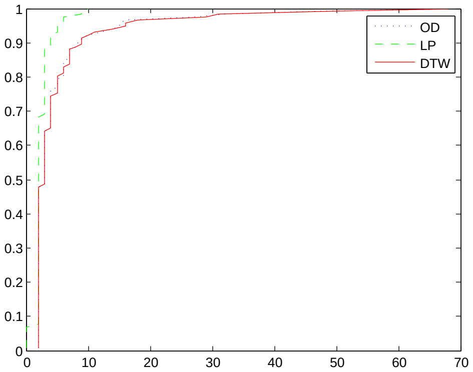


Figure 10: CDF of average trajectories per cluster for LP-DTW-OD

Figure 11: Average trajectories per top-k clusters for LP-DTW-OD

4.8 Average Trajectories Per Top-k Clusters



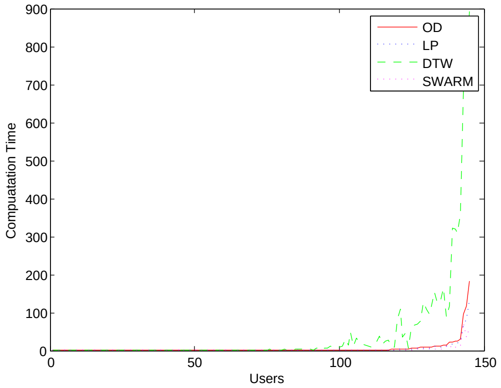


Figure 12: CDF of average trajectories per top-k clusters for LP-DTW-OD

Figure 13: Computation time comparison for LP-DTW-OD

4.9 Computation Time

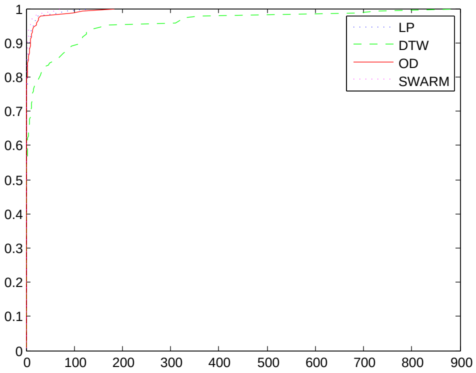


Figure 14: CDF of computation time for LP-DTW-OD