

TrajViz: A Tool for Visualizing Patterns and Anomalies in Trajectory

Yifeng Gao¹, Qingzhe Li¹, Xiaosheng Li¹, Jessica Lin¹, and Huzefa Rangwala¹

George Mason University

{ygao12,qli10,xli22,jessica}@gmu.edu,rangwala@cs.gmu.edu

Abstract. Visualizing frequently occurring patterns and potentially unusual behaviors in trajectory can provide valuable insights into activities behind the data. In this paper, we introduce TrajViz, a motif (frequently repeated subsequences) based visualization software that detects patterns and anomalies by inducing “grammars” from discretized spatial trajectories. We consider patterns as a set of sub-trajectories with unknown lengths that are spatially similar to each other. We demonstrate that TrajViz has the capacity to help users visualize anomalies and patterns effectively.

1 Introduction

With the rapid growth of tracking technology, a large amount of trajectory data are generated from users’ daily activities. Discovering frequently occurring patterns (motifs) and potentially unusual behaviors can be used to summarize the overwhelming amount of trajectories data and obtain meaningful knowledge. In this paper, we present TrajViz, a software that visualizes patterns and anomalies in trajectory datasets. TrajViz extends our previous work in time series motif discovery [?] to sub-trajectory pattern visualization. We consider patterns as a set of sub-trajectories with unknown lengths that are spatially similar to each other. We use a grid-based discretization approach to remove the speed information and adapt a grammar-based motif discovery algorithm, Iterative Sequitur (ItrSequitur), to discover the patterns. We design a user-friendly interface to allow visualization of repeated, as well as unusual sub-trajectories within the datasets.

2 Relate Work and Overview of TrajViz

Previously, we introduced a grammar-based motif discovery framework [?], which uses Sequitur[?], a grammar induction algorithm, to find approximate motifs of variable lengths in time series. However, the unique characteristics and challenges associated with spatial trajectory data make it unsuitable and difficult to apply the algorithms directly on trajectory data. In [?], the authors introduced STAVIS, a trajectory analytical system that uses grammar induction to infer variable-length patterns. However, its definition of “pattern” is based on

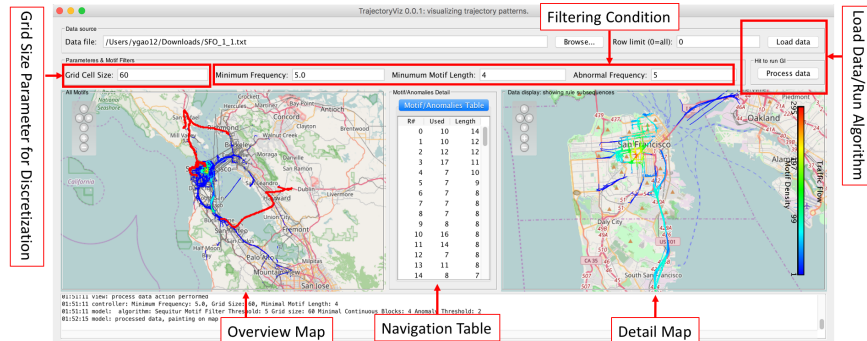


Fig. 1: Screenshot of TrajViz and Default View for San Francisco Taxi data [?]

time series motifs. Therefore, speed variation will significantly affect the quality of patterns discovered. Other work such as [?][?] focuses on either sequential pattern mining based on important locations, or trajectory clustering, both of which are different from the goal of our software.

A screenshot of TrajViz is shown in Fig. 1. TrajViz follows the Visual Information-Seeking Mantra [?]. After processing the data, an overview heat map of pattern density is displayed. User can zoom in to see the detailed map and use domain knowledge to filter out unwanted patterns by setting minimum frequency, minimum continuous blocks length (Minimal Motif Length) and maximum frequency for anomaly detection (Anomaly Frequency). Adjusting these thresholds does not require re-running the discretization and grammar induction steps (introduced in the next subsection). Further details on TrajViz can be found in goo.gl/cKCeDt.

3 Our Approach

3.1 Discretization

Before we can induce grammars on trajectory data, it is necessary to pre-process the data. We first convert the trajectory data to speed-insensitive symbolic sequences after removing noises from the trajectory dataset. To prepare for discretization, we divide the entire region into an $(\alpha \times \alpha)$ equal-frequency grid, where α is the grid size. We assign each grid cell a block ID sequentially from left to right and from top to bottom.

After block IDs are assigned, we use a four-step procedure to convert raw trajectory to a block ID sequence S_{block} . First, we up-sample the raw trajectory by using linear interpolation to ensure that the consecutive blocks in S_{block} are spatially adjacent. Then trajectories are converted into block ID sequences based on the order of traversal. Next, we perform further noise removal by removing blocks that are barely covered by the trajectory. Finally, numerosity reduction [?] is adopted to compress the sequence by only recording the first occurrence of

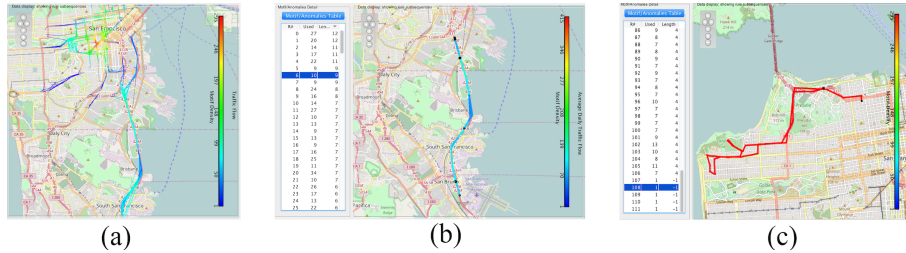


Fig. 2: Example of patterns detected in San Francisco Taxi Dataset [?] (a) Motif Heatmap (b) A pattern indicates a frequently visited route from the city to airport (c) An unusual (infrequent) round trip route

consecutively repeating symbol. S_{block} is insensitive to speed variation. This is an important property that allows us to detect spatially-similar sub-trajectories.

3.2 Grammar Induction with ItrSequitur

As demonstrated in previous work [?], a context-free grammar summarizes the structure of an input sequence. Intuitively, repeated substrings in S_{block} represent a set of similar sub-trajectories. Therefore, learning a set of grammar rules to identify repeating substrings from S_{block} can discover frequently occurring patterns (sub-trajectories) in trajectory data. Previous work [?] utilizes Sequitur [?], a linear complexity grammar induction approach, to learn the grammar rules. However, Sequitur can only detect patterns if they have identical symbolic representation. In TrajViz, we adapt an iterative version of Sequitur, called ItrSequitur [?], for more robust grammar induction. ItrSequitur iteratively rewrites the input sequence based on the output of Sequitur and re-induces the grammar on the revised sequence until no new grammar can be found. Different from Sequitur, ItrSequitur allows small variation in matching substrings. Therefore, it is robust to noise in the dataset.

3.3 Patterns/Anomalies Discovery and Motif Heatmap

TrajViz consolidates the patterns detected by merging patterns that have similar symbolic representations. Top-ranked frequent patterns that satisfy user-defined filtering conditions are listed in the motifs/anomalies table. User can navigate the patterns by clicking through the items in the table; a zoom-in of the selected pattern is then shown on the right panel. Fig. 2 shows screenshots of a motif and an anomaly detected. To show the direction of the trajectories, the start points are marked by black circles, and the end points are denoted by black squares.

For each point in a motif, we compute the point density by counting the number of points from other motifs within some distance threshold, and create a motif heatmap. A five-color gradient (blue-cyan-green-yellow-red) is built to linearly map the densities to their specific colors. The most dense points have the red colors while the least dense ones are in blue.

To find anomalies, we create a trajectory rule-density curve by counting the number of grammar rules covering each consecutive pair of block IDs (we consider a pair at a time in order to preserve the direction of the trajectory). The intuition is that, an anomalous subsequence would have zero or very few repetitions, hence low rule-density. TrajViz finds low-density subsequences within a trajectory and marks them as unusual routes (Fig. 2(c)).

4 Target Audience

TrajViz provides an efficient, interpretable, and user-interactive mechanism to understand functional activities behind massive trajectory data. TrajViz targets a diverse audience including researchers, practitioners, and scientists who are interested in discovering patterns in trajectory data.

ACKNOWLEDGEMENTS : We would like to thank Ranjeev Mittu at the Naval Research Lab (NRL) for the support and valuable suggestions on our work.

References

1. Y. Gao, J. Lin, and H. Rangwala. Iterative grammar-based framework for discovering variable-length time series motifs. In *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*, pages 7–12. IEEE, 2016.
2. J.-G. Lee, J. Han, X. Li, and H. Gonzalez. Traiclass: trajectory classification using hierarchical region-based and trajectory-based clustering. *Proceedings of the VLDB Endowment*, 1(1):1081–1094, 2008.
3. J. Lin, E. Keogh, L. Wei, and S. Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107–144, 2007.
4. C. G. Nevill-Manning and I. H. Witten. Identifying hierarchical structure in sequences: A linear-time algorithm. *J. Artif. Intell. Res. (JAIR)*, 7:67–82, 1997.
5. T. Oates, A. P. Boedihardjo, J. Lin, C. Chen, S. Frankenstein, and S. Gandhi. Motif discovery in spatial trajectories using grammar inference. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 1465–1468. ACM, 2013.
6. M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser. A parsimonious model of mobile partitioned networks with clustering. In *2009 First International Communication Systems and Networks and Workshops*, pages 1–10. IEEE, 2009.
7. P. Senin, J. Lin, X. Wang, T. Oates, S. Gandhi, A. P. Boedihardjo, C. Chen, S. Frankenstein, and M. Lerner. Grammarviz 2.0: a tool for grammar-based pattern discovery in time series. In *Machine Learning and Knowledge Discovery in Databases*, pages 468–472. Springer, 2014.
8. B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343. IEEE, 1996.
9. Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th international conference on World wide web*, pages 791–800. ACM, 2009.