A Demonstration of Summit: a Scalable Data Management Framework for Massive Trajectory

Louai Alarabi
Department of Computer Science
Umm Al-Qura University, Mecca, KSA
lmarabi@uqu.edu.sa

Mohamed F Mokbel
Department of Computer Science and Engineering
University of Minnesota - Twin Cities, Minneapolis, USA
mokbel@cs.umn.edu

***Abstract*—Driven by the ubiquity of location-based services, that produces a massive amount of moving objects. Querying and analyzing these data become a must for a wide range of applications. This demonstration presents a scalable data management framework. The proposed system is well-suited to efficiently support several basic queries, such as range, *k*NN, and similarity queries. These queries and the architectural design of the proposed system are extendable, in a way that enables users to build various applications and operations.**

I. INTRODUCTION

Recent advances in mobile computing, sensor networks, GPS, and satellite technology have made it possible to produce massive amounts of moving object data. For example, New York Taxi & Limousine archives over a Billion of taxi trajectories [6]. NASA daily archives over 4TB of stars and asteroids movement activity [5]. This results in increasing interest from scientists and domain experts in performing analysis tasks [3].

Domain experts who analyze trajectory data either (a) use *Heterogeneous* multiple platforms [7], [4], [8], in which trajectory operations built on-top of generic platforms, such as Hadoop or Spark. Using these platforms as-is results in sub-performance for trajectory operations that require indexing, e.g., Marmaray [4] project from Uber uses Hadoop as a backbone for storing data as non-indexed heap files, or (b) use *Big Spatio-temporal Frameworks* [2], [9] that are efficient for processing spatio-temporal data on MapReduce platform, yet, with a limited support for trajectory operations. This is mainly due to the inability of the index structure to accommodate storing the entire topology of moving objects. This in turn, affects the performance of basic trajectory operations, e.g., finding similarity between trajectories.

This demo presents Summit; a full-fledged open-source trajectory library for MapReduce framework [1], shipped with the source code of ST-Hadoop [2]. Summit injects the trajectory data awareness inside each of ST-Hadoop layers, mainly, indexing, operation, and language layers. Programs that deal with trajectory data using Summit will have up to order(s) of magnitude better performance than ST-Hadoop. The main reason is that ST-Hadoop treats the spatio-temporal information of trajectories as a set of spatio-temporal points or lines that are not connected together. This means that performing a basic trajectory operation such as similarity queries might end up scanning the whole dataset to check for trajectory connectivity before computing the similarity.

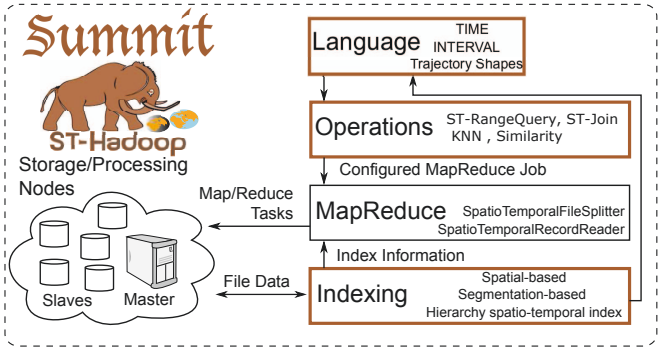


Fig. 1: Summit Architecture

During this demonstration, we will present to conference attendees a real prototype of Summit running on a remote cluster of 24 machines. We will be using NYC taxi [6] dataset, that contains over one billion records. The audience can submit any arbitrary queries on trajectories, and we will show the results on an interactive web-interface. We will also demonstrate the deep insights behind the performance gain of Summit with a live and dynamic comparison with other frameworks. Visually we will animate in-depth details on how Summit achieve its load balancing and performance by illustrating the system internals.

II. SUMMIT OVERVIEW

Figure 1 gives an overview of the Summit system architecture. Summit is a full-fledged open-source library on the ST-Hadoop MapReduce framework [2] with a *built-in* native support for trajectory data. The cluster in Summit contains one master node that breaks a map-reduce job into smaller tasks, carried out by slave nodes. Summit modifies three core layers of ST-Hadoop, namely, *Language*, *Indexing*, and *Operations*. Details of these layers briefly described as follow:

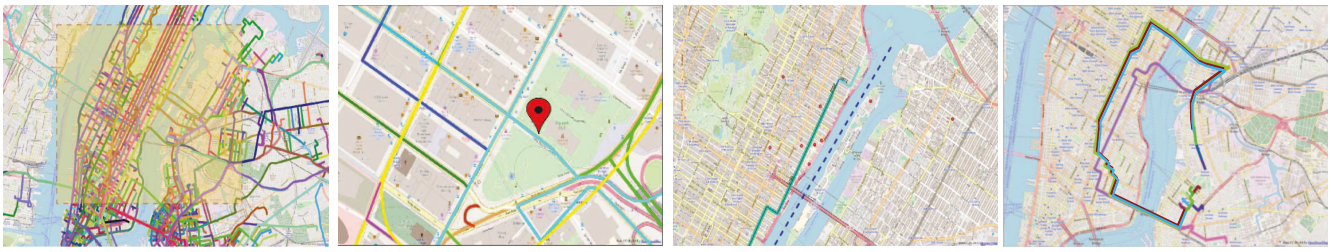
Language Layer: This layer provides a simple high-level SQL-like language that supports moving data objects (i.e., TRAJECTORY, which described as a consecutive sequence of spatio-temporal points `ST-Trajectory`.

Indexing Layer: Summit employs a two levels index structure of *global* and *local*. The *global* index partitions the data across the computation nodes, while the *local* index organizes the

2375-0324/20/$31.00 ©2020 IEEE
DOI 10.1109/MDM48529.2020.00046

(a) Interactive Interface (b) Spatial-based (c) Segmentation-based

Fig. 2: Summit Internal Demonstration



(a) Range (b) *k*NN point-based (c) *k*NN Trajectory-based (d) Similarity

Fig. 3: Summit Operations Demonstration

data inside each node in the Hadoop Distributed File System (HDFS blocks). Summit begins by temporally slice moving objects into disjoint time interval sets, then for each set of interval Summit introduces two spatial indexing techniques named spatial-based and segmentation-based. The spatial-based approach preserves the spatio-temporal locality closeness between sub-trajectories across multiple HDFS blocks. Meanwhile, a segmentation-based approach guarantees that the entire trajectory is mainly stored in a single HDFS block. The accommodation of those two approach highly depending on the target applications. For instance, segmentation-based partitioning is more in favor of operations that not only need to process the locality of trajectories but also their semantic or shapes over time, such as Similarity and join queries; in the meantime, spatial-based provide faster access to essential operation such as range and *k*NN queries.

Operations Layer: This layer encapsulates the implementation of three common trajectory operations, namely, range query, *k*NN, and similarity queries. More operations can be added to this layer.

III. Demonstration Scenario

Conference attendees can interact with the Summit system through a nicely designed web interface, where they can trigger the execution of their query and visualize results instantly. For genuine insight, the query will be submitted to Summit, ST-Hadoop, and Hadoop frameworks, respectively. The three frameworks installed on a separate cluster with the same configuration and dataset.

A. Index Demonstration

Conference audience can load datasets to Summit and visualize the boundaries of its index on the map, as shown in figure 2a. The different between Summit partitioning techniques of spatial-based and segmentation-based on a particular day are shown in figures 2b and 2c, respectively. Rectangles represent partitions, each of a size 128MB of data.

B. Trajectory Operations

Conference attendee can interact with Summit by submitting operations via the web-interface as shown in Figure 2a. Attendee can tune operation parameters, such as specifying the region of interest, time window, number of k, and similarity threshold. Figures 3a, 3b, 3c, and 3d depicts examples of users' results for the following operations: (1) Range query that "*finds all taxi in New York time square on 2016 new year's eve*", (2) *k*NN query that "*finds 8-nearest trajectories to New York city hall on 6 of JAN 2020*", (3) *k*NN query that "*finds 4-nearest trajectories their trip aligned with East River during the entire month of NOV 2019*", (4) Similarity operation that "*find 10 similar taxis to a given trajectory drawn by the user on the map on the 8 of JUN 2019*". Users can freely navigate through the result on the map. Conference attendee can alter these three scenarios of range, *k*NN, and similarity queries.

References

[1] L. Alarabi. Summit: a scalable system for massive trajectory data management. In *SIGSPATIAL*, 2018.
[2] L. Alarabi, M. F. Mokbel, and M. Musleh. ST-Hadoop: A MapReduce Framework for Spatio-temporal Data. *GeoInformatica*, 2018.
[3] X. Ding, L. Chen, Y. Gao, C. S. Jensen, and H. Bao. Ultraman: A unified platform for big trajectory data management and analytics. *PVLDB*, 2018.
[4] Marmaray. https://github.com/uber/marmaray.
[5] Data from NASA's Missions, Research, and Activities. http://www.nasa.gov/open/data.html.
[6] Data from NYC Taxi and Limosuine Commission. http://www.nyc.gov/html/tlc/.
[7] S. Ruan, R. Li, J. Bao, T. He, and Y. Zheng. Cloudtp: A cloud-based flexible trajectory preprocessing framework. In *ICDE*, 2018.
[8] Z. Shang, G. Li, and Z. Bao. DITA: distributed in-memory trajectory analytics. In *SIGMOD*, 2018.
[9] M. A. Whitby, R. Fecher, and C. Bennight. Geowave: Utilizing distributed key-value stores for multidimensional data. In *SSTD*, 2017.