Sipresk: A Big Data Analytic Platform for Smart Transportation

Hamzeh Khazaei(✉), Saeed Zareian, Rodrigo Veleda, and Marin Litoiu

School of Information Technology, York University, Toronto, ON, Canada
{hkh,zareian,rveleda,litoiu}@yorku.ca

Abstract. In this paper, we propose a platform for performing analytics on urban transportation data to gain insights into traffic patterns. The platform consists of data, analytics and management layers and it can be leveraged by overlay traffic-related applications or directly by researchers, traffic engineers and planners. The platform is cluster-based and leverages the cloud to achieve reliability, scalability and adaptivity to the changing operating conditions. It can be leveraged for both on-line and retrospective analysis. We validated several use cases such as finding average speed and congested segments in the major highways in Greater Toronto Area (GTA).

Keywords: Smart transportation · Data analytics platform · Traffic data · Adaptive systems

1 Introduction

The effective movement of people and vehicles has long been critical to economies and qualities of life worldwide. Inefficiencies cost money, increase pollution and take time away from peoples lives. The problem is, the supply of transportation infrastructure grows more slowly than demand. Cars can be built more quickly than roads. Cities grow faster than highways can be expanded. Even if there were a limitless supply of money and personnel for road construction, many areas are already built out. That is why the transportation industry is turning to data analytics to find smarter ways to use the resources that exist, reduce congestion, and improve the travel experience [4].

Researchers have modelled different aspects of transportation using travel surveys, fluid flow model or game theory. With the emergence of new data sources, such as traffic sensors, cameras, GPS-devices and cell phones, opportunities have emerged for near real-time and at-rest data analytics [15].

The velocity and magnitude of data varies across sources. For example, loop detector sensors, embedded in the Highways of Greater Toronto Area (GTA) collect data at every 20-s intervals. Meanwhile, the social activity varies during the day. The data exists in a variety of forms including numerical, textual and visual either in structured or un-structured fashion. The data is collected over

A. Leon-Garcia et al. (Eds.): Smart City 2015, LNICST 166, pp. 419–430, 2016.
DOI: 10.1007/978-3-319-33681-7_35

years, and is voluminous in size. Managing and mining this data is truly a big data problem [10].

The analytics, e.g. average daily traffic and congested segments, are of immediate interest to ministry of transportation, various municipalities and transportation planners. Meanwhile, data mining might benefit a wider audience by predicting traffic congestion, uncovering timings of various hot spots and/or suggesting fastest route [19].

In this paper, we present a big data analytics architecture, Sipresk, that is tailored to transportation data and adaptation. Sipresk is not an acronym and it means "Swallow" (the bird) in the old Persian language. The architecture is an instantiation of the conceptual architecture presented by some of the current authors in [19]. More specifically, Sipresk has a multi-tier architecture including, data, analytics and management components. Data layer ingests data from multiple sources, performs en route data processing with user-specified plug-ins, and normalizes it for analytical jobs. The analytical layer supports three types of job analytics: (a) interactive processing, (b) batch processing, and (c) graph processing. Sipresk requires to handle large magnitude of data and number of users. Therefore, a MAPE-K loop based solution [23] is employed to keep the Sipresks performance at an optimal level, for example, scaling out in times of high load. We realize an instance of Sipresk to shed some lights on congested segments and average daily speed in Ontario's highways.

The rest of the paper is organized as follows. In Sect. 2 we highlight the functional and non-functional requirements of Sipresk. Section 3 specifies the characteristics of available traffic data and the data management component. In Sects. 4 and 5, we describe our analytic engine and the management system. We present a case study on loop detectors data by leveraging Sipresk platform in the Sect. 6. Section 7 surveys related research and Sect. 8 concludes the paper.

2 Functional and Non-functional Requirements of Sipresk

There are different types of users that pose different questions to a transportation analytics platform such as Sipresk; Shtern et al. [19] classify them into four categories:

1. Transportation Manager
   - How was the traffic on the highways yesterday?
   - Which regions saw the worst traffic yesterday?
2. Traffic Engineer
   - Which loop detectors are malfunctioning?
   - Which locations do congestion occur and what time?
   - How do congestions start and spread?
3. Planner Researcher
   - What will be the traffic volume in future?
   - Where will the future bottlenecks be? How can they be addressed?
   - How will the hybrid cars effect the environment?
4. Policy Maker

- What are the suitable toll charges on the highways?
- How much more should heavy vehicles pay relative to cars?

In order to answer above type of questions, we design Sipresk in a way that can support spatiotemporal, graph, periodic, statistical, prediction and fusion queries as highlighted in [19]. These queries can be mapped into 4 classes of workloads, namely batch, interactive, stream and graph processing that must be supported by Sipresk. The characteristics of traffic data and the functional requirements impose the following non-functional requirements on Sipresk:

- **Scalability and Elasticity**: handle constantly increasing size of traffic data, and varying number of users.
- **Efficient range scans**: provide efficient range scans to support data aggregations.
- **Low-latency of storage and access**: offer low-latency between storage of a data source sample, and availability for analysis through specialized interfaces.
- **Autonomic management**: adapting to unpredictable changes and optimizing its performance by self-awareness, auto configuration, recovering from failures, and protecting itself from malicious users.
- **High Availability**: provide high availability to support real-time data ingestion and online statistics.

3 Data Management Subsystem

Table 1 presents the available traffic data in GTA, Ontario, Canada. Sipresk provides storage and analytics capabilities on all available data. In this work we leverage the loop detectors data to answer the interested questions. For a detailed description of data refer to [15,19].

The data management layer pools the data from CVST platform [3,20] that collects traffic data directly from multiple sources. The acquired data is then processed according to user specified plug-ins and is stored in HBase or HDFS (as the data warehouse) depending on the data type and size. Data with small size is stored in HBase while large payloads go directly into HDFS. For example, speed metrics can be stored in HBase. Meanwhile, videos can be stored directly in HDFS, while any meta-data extracted during the pipeline processing is also stored in HBase.

We chose HBase for our warehouse because HBase supports high access throughput, strictly consistent reads and writes, and efficient range scans. It also provides low latency of storage and access [2]. Hsu et al. [8] have used HBase to create spatial indexes, a main requirement for efficient spatial queries. However, HBase supports only one index. To overcome this shortcoming, we use Solr[1] to generated additional indexes on data which are then stored back into HBase.

We provide on demand analytic datastores, e.g., key-value, document, wide column or graph stores for research projects on top of the warehouse. The type of

[1] http://lucene.apache.org/solr.

Table 1. Available traffic data in GTA

Data source	Data format	Data type	Description
Loop detector sensors	Structured	Numerical	Average speed and traffic flow per 20-s
Traffic cameras	Unstructured	Video	Blob of video in stream format
Mobile devices (GPS/Bluetooth)	Structured	Numerical	Location and speed via cellular network
Toronto traffic survey	Structured	Text/Numerical	This survey has a very large sample size resulting in interviews with hundreds of thousands of households
Incident reports	Structured	Text/Numerical	Witness reported issues
Public transportation	Structured	Numerical	Tabular schedules of public transportation
Media outlets	Semi-structured	Text/Numerical	e.g., radio stations reports, CP24
Social nedia	Unstructured	Text	Crowd reported information

analytic datastore is dependent on target queries and the nature of the research project data. For example we may create a key-value storage for loop detectors data for Winter of 2014. This layered architecture is mainly adopted for the sake of isolation, performance, scalability and availability. By doing so, each project has the full access to the data in the most appropriate datastore technology. Figure 1 shows the concept of our adopted layered architecture.

4 Analytic Subsystem

The analytic engine in Sipresk is based on Sahara project, which is the data processing component in OpenStack[2] foundation. It can deliver different types of data processing clusters based on Apache Spark or Hadoop ecosystems. The analytic engine consists of modellers, graph processing, real-time processing, batch processing and machine Learning algorithms at large scale. The analytic engine provides high-level interfaces to analyze traffic related problems.

For instance, in case of a Spark[3] cluster deployment, R[4] gives the user the capability to construct statistical and prediction models from the traffic data; MLlib allows analysts to detect patterns and build classters over data; GraphX provides the ability to perform iterative graph processing at large scale such as calculate travel time on a route; Spark SQL and Spark Streaming provide fast real-time and batch processing in memory. In case of a hadoop-based clusters, corresponding tools will be available for above mentioned type of analytics. Figure 2 shows the high level architecture of Sipresk.

[2] http://www.openstack.org.
[3] https://spark.apache.org.
[4] http://www.r-project.org.

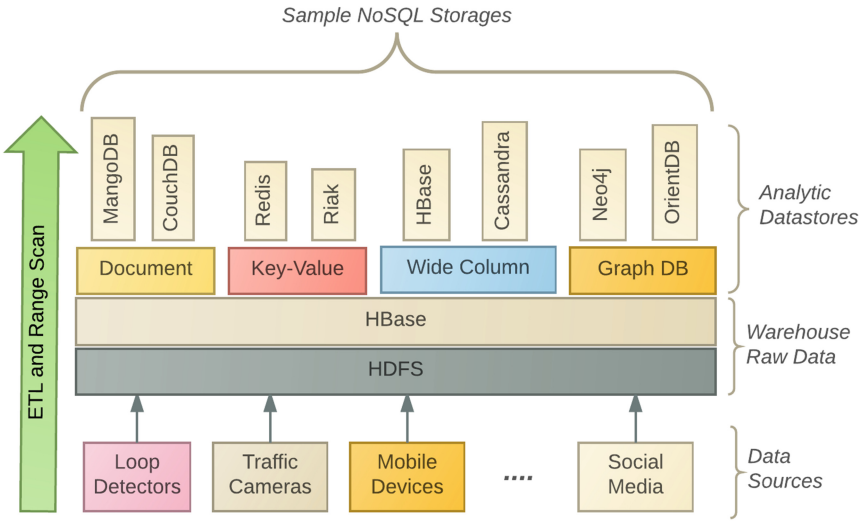

Fig. 1. Layered data management platform.

5 Management Subsystem

We expect very large data sizes and varying number of users, and the need of platform to adjust to changing demands. Therefore, a management system based on the MAPE-k loop [9] is an integral part of the platform. The manager monitors the analytic engine and data management layer, analyzes the current conditions, plans actions to take the platform to desirable state, and executes the corresponding actions. It may acquire, adjust or release resources from the cloud.

We leverage our in-house implementation of MAPE-K methodology, K-Feed (Knowledge-Feed) [23], to manage both data management and analytic engine. K-Feed monitors the platform closely and gathers the performance metrics. It analyzes these collected metrics in real-time and acts when certain type of conditions are met. For example, if aggregated CPU utilization stays high (e.g., $>$ 60 % for 2 min) in HBase regional servers or in workers in Spark cluster, it will add one node to the pool to bring the platform to normal condition. In addition to reactive adaptation, K-Feed also is able to provide proactive adaptation by doing statistical modeling on the performance metrics data. Figure 3 shows the high level architecture of K-Feed.

6 Case Study

The traffic congestion is a major issue for GTA. In this section, we use Sipersk to investigate the major highways of Toronto and characterize the average speed and occupancy during the year of 2014 and first 4 months of 2015. This is just a

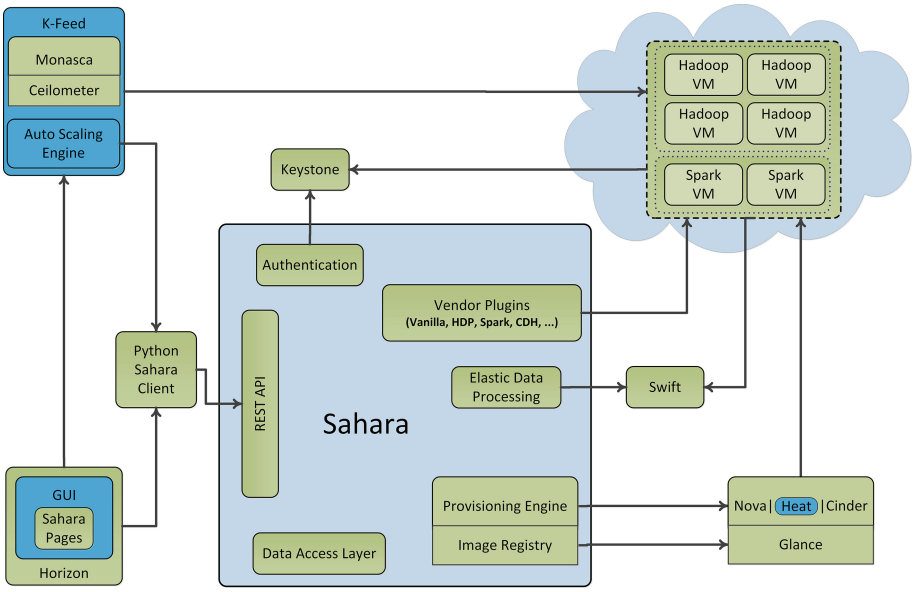

Fig. 2. Analytic engine in Sipresk.

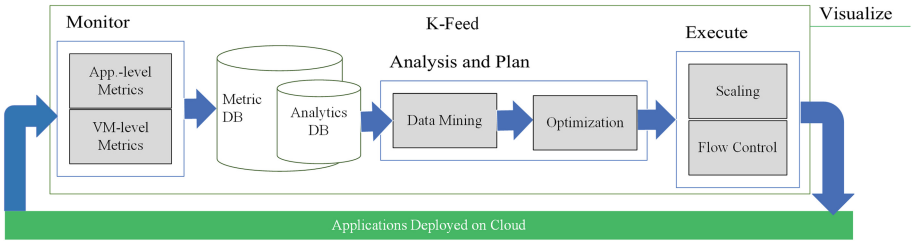

Fig. 3. High level architecture of K-Feed [23].

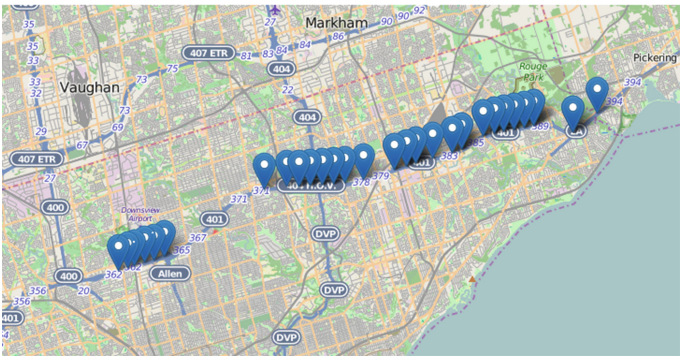

Fig. 4. Congested points in 401 East, aggregated for Wednesdays in October 2014, during morning rush hours.

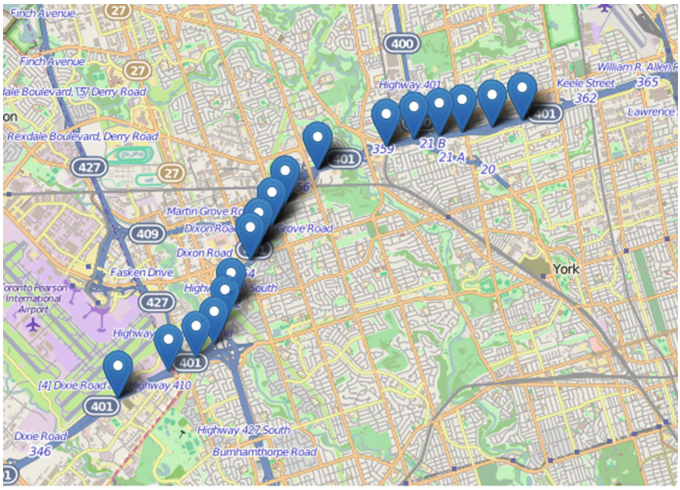

Fig. 5. Congested points in 401 West, aggregated for Wednesdays in October 2014, during evening rush hours.

use case of Sipresk capabilities for smart transportation. Specifically, we perform an analytical study to answer the following questions:

1. What are the congestion points in GTA's highways during morning and evening rush hours?
2. What was the average speed for highways 401, 404, 400, 407, DVP and QEW during the last 16 months? We are interested in daily average speed for congested segments of the above highways.

Our analysis is conducted on the data collected from the sensors embedded in the highways of GTA. In the CSV format, the size of data is 30 GB for the whole year of 2014 and the first 4 months of 2015. We conduct the study using a customized deployment of Sipresk on SAVI [18] cloud. SAVI is an OpenStack-base academic cloud platform being leveraged by many Canadian universities.

We use a HBase cluster as the analytics datastore on top of our warehouse; the data cluster consists of 8 VMs with the flavour of "m.large" (i.e., 4 vCPU, 8 GB RAM, 80 GB disk) grouping in 2 master and 6 worker nodes. The HDFS capacity in the cluster is 406.2 GB of usable space. We use Apache Spark stand alone deployment as the analytic engine. It comprises of 3 "m.xlarge" (i.e., 8 vCPU, 16 GB RAM, 160 GB disk) including one master and 2 slave nodes. For the management system, i.e., K-Feed, we deploy two "m.medium" instances; one for the monitoring purpose and collecting performance metrics and the other one for analysis, planning and execution of the outcome commands. All nodes in Sipresk platform are running Ubuntu 14.04 LTS as the underlying operating system.

Once Sipresk is instantiated, the analyst can focus on the tasks she/he is interested. We, for example, have been able to carry out various and extensive

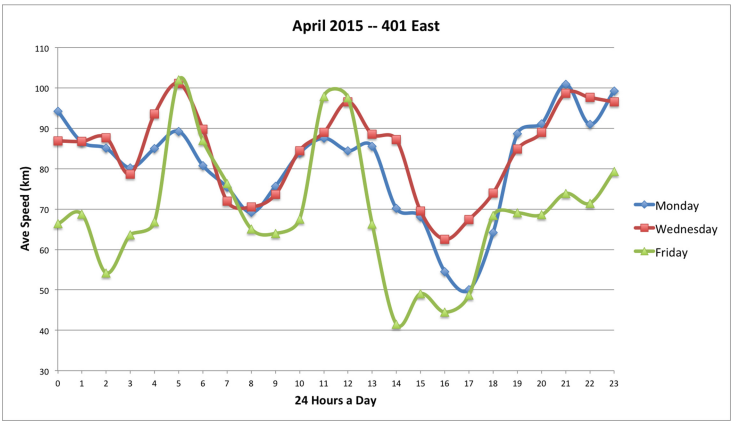

Fig. 6. Average speed for congested points in 401 East during April 2015.

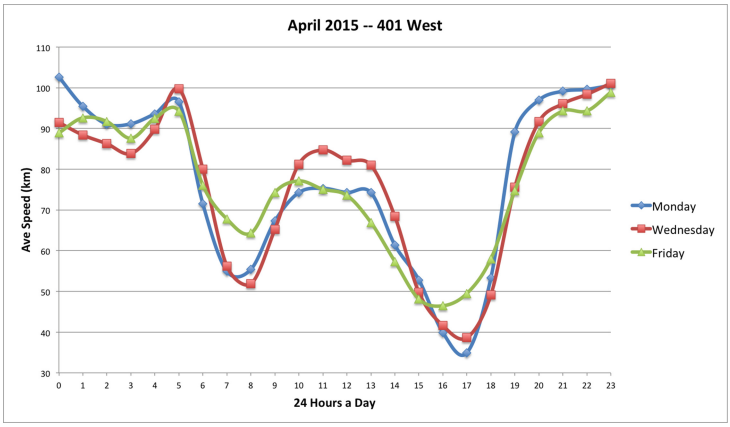

Fig. 7. Average speed for congested points in 401 West during April 2015.

analytics on loop detector data and detailed next. We define a point in highways as congested, if it's average speed is less that 60 % of the average speed in the whole highway (i.e. the segment of highways that is instrumented by loop detectors). Using this definition, we identified the congested points in GTA highways for each day during the whole data set. Then we aggregated results for each month (e.g. obtaining the average speed for all Fridays in January) in order to reduce the noise and avoid rare congestions (e.g. an accident casing the congestion). We did this analytics for the whole year of 2014 and the first 4 months of 2015. However, due to space limit, we only show two samples of our results (Figs. 4 and 5) here. The extra results can be found in the Adaptive Systems Research Lab (ASRL) portal[5].

[5] http://www.ceraslabs.com/people/hamzeh/bigdasc2015paper.

We also identified the daily average speed for congested segments in the highways for the entire data set. Then, we aggregated weekdays for each month and visualized the average speed for the 24 h of that day. Figures 6 and 7 show the average speed for Mondays, Wednesdays and Fridays in April 2015. For extra results corresponding to other months and weekdays, please refer to the ASRL portal specified above.

7 Related Work

In this section we survey the recent works in analytics of transportation data and related research in big data analytics in the area of smart cities.

Mian et al. proposed a platform to support analytics over traffic data [15]. The platform includes multiple engines to support various types of analytics and processing ranging from text searching to route planning. However, they used Matlab as their analytic engine that made their case study limited to three months. Also the data layer proposed in this work is not extendable for supporting different data type as well as research projects. In this work, our goal was to relax these limitations.

Zareian et al. [23] proposed a monitoring system for performance analysis of applications deployed on cloud. Their platform K-Feed, can perform at scale monitoring, analysis, and provisioning of cloud applications. It supports both proactive and reactive scalability. We use K-Feed in Sipresk as the management system.

Shtern et al. [19] propose a conceptual architecture for a data engine, Godzilla, to ingest real-time traffic data and support analytic and data mining over transportation data. They specified the requirements and specifications of a multi-cluster approach to handle large volumes of growing data, changing workloads and varying number of users. We incorporated some of the specifications and design patterns mentioned in this work to realize and deploy the Sipresk. Sipresk is a concrete architecture and implementation.

There are about 1,600 loop detector sensors and 200 cameras located in the highways of northern Belgium. The measurements, collected at the frequency of 1 m, are stored into a central database in the raw, unprocessed and non-validated form [14]. The California Freeway Performance Measurement System (PeMS) has about 26,000 loop detector sensors collection data at 30 s interval into an Oracle database system [21]. The type of analytics that they are doing on these two projects is not clearly known.

The city of Bellevue has about 180 loop detector sensors, and the data captured is available in CSV files at every minute interval. GATI system [22] downloads the traffic data from the Bellevue data server and stores it in a MySQL database system. Hoh et al. [7] and Lo et al. [13] collect traffic data by probing GPS-equipped vehicles, and store it in a Microsoft SQL Server and PostgreSQL respectively.

The above systems usually collect data from a single source, which has structured type, and is stored in a relational database system. The traditional database systems have limitations over horizontal scalability [1]. The above systems

display the collected data on a map, estimate travel times, or show current traffic conditions in a web-browser or over cell phones.

In contrast, our data platform will collect data from multiple sources, which have multiple types, and would be stored in a scalable NoSQL layer. The vision is to provide a comprehensive analytic platform for traffic analysts by offering multiple interfaces. Data from multiple sources can be combined to lead to new insights. For example, it will be possible to study effects of introducing new toll charges on traffic volumes and reaction of traveler using the tolled highways. However, developing efficient and scalable platforms for Big Data is actively being researched [2,5,11,12]. Building such a platform is a multi-facet problem, and we provide examples of research in addressing a particular aspect of the problem.

Rabkin et al. [16] explores the reasons for the downtime of a Hadoop cluster. They discover misconfiguration to be the biggest reason for failures. Heger [6] presents a methodology to tune a Hadoop cluster for varying workload conditions. Meanwhile, Rao et al. [17] explores the performance issues of Hadoop in heterogeneous clusters and suggest possible ways to address the issues. Rabkin et al. and Rao et al. explores methods to reduce downtown and improve performance. We incorporate their ideas to create a smart deployer for Sipresk platform.

We also see several commercial solutions such as Google[6] and Inrix[7] mainly focus on providing predefined traffic analytics and reports accessible through dashboards and/or APIs. However, our work focuses on providing a generic platform that enables ad-hoc analytics over traffic data.

8 Conclusion

In this paper, we presented a big data platform, Sipresk, to support analytics over large traffic data. Sipresk supports various types of analytics on different data sources. It can adapt to the changing environment – workload, failure, networking and the like – by leveraging a MAPE-K loop based solution. In addition we implemented and deployed an instance of Sipresk to provides insights on highways traffic in GTA. We specified the congested points in all highways and also depicted the average speed in those congested segments for the last 16 months.

As the future work, we plan to extend our analytics to the rest of available traffic data in order to answer other research questions mentioned in Sect. 2. In particular, we are interested in building statistical models for predicting the traffic condition in GTA for near future.

Acknowledgments. This research was supported by the SAVI Strategic Research Network (Smart Applications on Virtual Infrastructure), funded by NSERC (The Natural Sciences and Engineering Research Council of Canada) and by Connected Vehicles

[6] https://support.google.com/maps.
[7] www.inrix.com.

and Smart Transportation (CVST) funded Ontario Research Fund. We acknowledge the contribution of the ONE-ITS platform in providing access to aggregated of-line traffic data. We also would like to thank Brian Ramprasad for his help in deployment of HBase clusters and Yan Fu for her help on data collection.

References

1. Abadi, D.J.: Data management in the cloud: limitations and opportunities. IEEE Data Eng. Bull. **32**(1), 3–12 (2009)
2. Borthakur, D., Gray, J., Sarma, J.S., Muthukkaruppan, K., Spiegelberg, N., Kuang, H., Ranganathan, K., Molkov, D., Menon, A., Rash, S., et al.: Apache hadoop goes realtime at facebook. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of data, pp. 1071–1080. ACM (2011)
3. CVST. Connected Vehicles and Smart Transportation, June 2015. http://cvst.ca
4. Dirks, S., Gurdgiev, C., Keeling, M.: Smarter cities for smarter growth: how cities can optimize theirsystems for the talent-based economy. IBM Institute for Business Value (2010)
5. Hayes, M., Shah, S. Hourglass: a library for incremental processing on hadoop. In: 2013 IEEE International Conference on Big Data, pp. 742–752. IEEE (2013)
6. Heger, D.: Hadoop performance tuning-a pragmatic & iterative approach. CMG J. **4**, 97–113 (2013)
7. Hoh, B., Gruteser, M., Herring, R., Ban, J., Work, D., Herrera, J.-C., Bayen, A.M., Annavaram, M., Jacobson, Q.: Virtual trip lines for distributed privacy-preserving traffic monitoring. In: Proceedings of the 6th International Conference on Mobile systems, Applications, and Services, pp. 15–28. ACM (2008)
8. Hsu, Y.-T., Pan, Y.-C., Wei, L.-Y., Peng, W.-C., Lee, W.-C.: Key formulation schemes for spatial index in cloud data managements. In: 2012 IEEE 13th International Conference on Mobile Data Management (MDM), pp. 21–26. IEEE (2012)
9. Kephart, J.O., Chess, D.M.: The vision of autonomic computing. Computer **36**(1), 41–50 (2003)
10. Kitchin, R.: The real-time city? big data and smart urbanism. GeoJ. **79**(1), 1–14 (2014)
11. Konstantinou, I., Angelou, E., Boumpouka, C., Tsoumakos, D., Koziris, N.: On the elasticity of NoSQL databases over cloud management platforms. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 2385–2388. ACM (2011)
12. Lane, N.D., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., Campbell, A.T.: A survey of mobile phone sensing. IEEE Commun. Mag. **48**(9), 140–150 (2010)
13. Lo, C.-H., Peng, W.-C., Chen, C.-W., Lin, T.-Y., Lin, C.-S. Carweb: a traffic data collection platform. In: 9th International Conference on Mobile Data Management, MDM 2008, pp. 221–222. IEEE (2008)
14. Maerivoet, S., Logghe, S.: Validation of travel times based on cellular floating vehicle data. In: Proceedings from 6th European Congress and Exhibition on Intelligent Transport Systems and Services (2007)
15. Mian, R., Ghanbari, H., Zareian, S., Shtern, M., Litoiu, M.: A data platform for the highway traffic data. In: 2014 IEEE 8th International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), pp. 47–52. IEEE (2014)
16. Rabkin, A., Katz, R.H.: How hadoop clusters break. IEEE Softw. **30**(4), 88–94 (2013)

17. Rao, B.T., Sridevi, N., Reddy, V.K., Reddy, L.: Performance issues of heterogeneous hadoop clusters in cloudcomputing (2012). arXiv preprint arXiv:1207.0894
18. SAVI. Smart Applications on Virtual Infrastructure. Cloud platform, June 2015. http://www.savinetwork.ca
19. Shtern, M., Mian, R., Litoiu, M., Zareian, S., Abdelgawad, H., Tizghadam, A.: Towards a multi-cluster analytical engine for transportation data. In: 2014 International Conference on Cloud and Autonomic Computing (ICCAC), pp. 249–257. IEEE (2014)
20. Tizghadam, A., Leon-Garcia, A.: Connected Vehicles and Smart Transportation - CVST Platform, June 2015. http://cvst.ca/wp/wp-content/uploads/2015/06/cvst.pdf
21. Varaiya, P.: Reducing highway congestion: an empirical approach. Eur. J. Control **11**(4), 301–309 (2005)
22. Wu, Y.-J., Wang, Y., Qian, D.: A google-map-based arterial traffic information system. In: Intelligent Transportation Systems Conference, ITSC 2007, pp. 968–973. IEEE (2007)
23. Zareian, S., Veleda, R., Litoiu, M., Shtern, M., Ghanbari, H., Garg, M.: K-feed, a data-oriented approach to application performance management in cloud. In: 2015 IEEE 8th International Conference on Cloud Computing (CLOUD), June 2015. IEEE (2015)