Big Data Analytics Architecture for Real-Time Traffic Control

Sasan Amini
Chair of Traffic Engineering and Control
Technical University of Munich
Munich, Germany
sasan.amini@tum.de

Ilias Gerostathopoulos
Faculty of Informatics
Technical University of Munich
Munich, Germany
ilias.gerostathopoulos@tum.de

Christian Prehofer
Faculty of Informatics
Technical University of Munich
Munich, Germany
christian.prehofer@tum.de

***Abstract*— The advent of Big Data has triggered disruptive changes in many fields including Intelligent Transportation Systems (ITS). The emerging connected technologies created around ubiquitous digital devices have opened unique opportunities to enhance the performance of the ITS. However, magnitude and heterogeneity of the Big Data are beyond the capabilities of the existing approaches in ITS. Therefore, there is a crucial need to develop new tools and systems to keep pace with the Big Data proliferation. In this paper, we propose a comprehensive and flexible architecture based on distributed computing platform for real-time traffic control. The architecture is based on systematic analysis of the requirements of the existing traffic control systems. In it, the Big Data analytics engine informs the control logic. We have realized the architecture in a prototype platform that employs Kafka, a state-of-the-art Big Data tool for building data pipelines and stream processing. We demonstrate our approach on a case study of controlling the opening and closing of a freeway hard shoulder lane in microscopic traffic simulation.**

Keywords—Intelligent Transportation System; Big Data; Kafka; Real-time traffic control

I. INTRODUCTION

The rapid advancement of communication and detection technologies, low-cost and widespread sensing and a dramatic drop in data storage costs have significantly increased the amount of easily extractable information on transport and mobility. According to [1] "*the volume and speed at which data are generated, processed and stored is unprecedented*". In essence, Big Data is a process of gathering, management and analysis of data to generate knowledge and reveal hidden patterns. The advent of Big Data has triggered disruptive changes in many fields including Intelligent Transport Systems (ITS) with a wide range of applications from smart urban planning to enhanced vehicle safety. However, methodologies and regulations in many domains of ITS have not kept pace with the proliferation of Big Data. More specifically, the current traffic control approaches such as feedback loop or model predictive methods do not fit to the paradigm of Big Data analytics. The evolution of the existing ITS into a data-driven system has been foreseen by other researchers [2].

This work is part of the TUM Living Lab Connected Mobility (TUM LLCM) project and has been funded by the Bavarian Ministry of Economic Affairs and Media, Energy and Technology (StMWi) through the Center Digitisation.Bavaria, an initiative of the Bavarian State Government.

In Big Data approaches, the challenge is not anymore to collect the data, but to draw valuable conclusions by properly analyzing them. To be clear, exploiting the collected data has been always considered by researchers and practitioners, but the high velocity, magnitude and heterogeneity of massive stream of real-time data push the limits of the current storage, management and processing capabilities. Admittedly, the classical statistical methodologies are challenged (especially with respect to bias) and cannot be applied on the emerging opportunistically and crowed sensed data streams. Some of these data streams are structured in a way that serve only one predefined purpose and cannot be directly used for other means. Yet, there are emerging unstructured data such as context-based data [3] from the internet and social media as well as credit card transactions that is not clear if they can be used to better understand the mobility patterns. Therefore, it is essential to develop modern system abstractions that allow us to efficiently process large and new data streams.

Even though the number of studies on Big Data in transport has considerably increased, most of the systems deployed so far in order to support Big Data analytics in ITS rely on ad hoc architecture solutions [4, 5]. They focus on satisfying specific predefined goals (mining GPS data, predicting traffic flow, etc.) and are hard to extend to accommodate different applications and data sources. This results in rigid systems and overall limits the uptake of Big Data technologies in ITS.

In response, we propose a comprehensive architecture for Big Data analytics for real-time traffic control. The architecture is based on systematic analysis of the requirements of the existing traffic control systems. It is flexible in that it can accommodate an open-ended and diverse set of data sources and a number of different ITS applications, particularly decision support for real-time traffic control. The architecture is modular in the sense that different analytical engines as well as data storage systems could be easily plugged in. At the same time, it offers a minimum set of functionalities for an easy start. To shield the users (in this case, mostly ITS administrators) from the complexities of Big Data technologies, our architecture offers a core set of interfaces and concepts that facilitate the programing of different data analysis tasks.

A substantial part of the proposed architecture has been reified in a platform prototype which relies mainly on a Kafka, an established tool for efficient processing of Big Data streams.

Thanks to the built-in mechanisms of Kafka, the data analysis is scalable, i.e. can scale to a large number of data sources simultaneously sending data at high rates, and reliable, i.e. it can tolerate hardware faults (e.g. failing computing machines) without loss of data. We believe that our approach can be used for accelerating the adoption of Big Data analytics in ITS.

The remaining of this paper is structured as follows: in section II, we give an overview of both the existing ITS applications and the Big Data analytics approaches, with emphasis on Kafka . In section III, an overview of related work with regard to Big Data architectures in ITS is presented. Our proposed architecture is described in Section IV and validated in a microscopic traffic simulation in Section V. In Section VI we discuss the obtained results, and give a conclusion together with our recommendations for further research in Section VII.

II. Background

A. Characteristics of real-time traffic control system

The objective of this paper is to adopt the emerging Big Data approaches i.e. Kafka in building an extensible real-time traffic control system. Thus, it is crucial to explore the similarities and differences among the existing control system and stream analytics performed on Kafka. Real-time traffic control systems are composed of two main components: observation of the situation (data collection) and implementation of the selected control strategy (data processing and information dissemination). A local system analyzes the real-time input data where they are integrated and process to identify the situation (e.g. incident detection). Once a threshold is exceeded, one of the predefined strategies is implemented to optimize the controller objective function. In some cases, a central system defines the strategic objective and local systems are flexible to act adaptively according to the local situations. Feedback loop and model Predictive Control (MPC) [34] are the most common traffic control approaches. However, they are mainly single-objective and require purposely-sensed data (i.e. fundamental traffic flow parameters).

B. Data-driven Approaches in ITS

Examples of Big Data approaches in the existing transport system are limited. The majority of studies are still using the traditional data sources. i.e. inductive loop detectors and travel surveys, and the new data sources are limited to mobile location data, probe vehicle data, public transport smart cards, social network and web-based information and crowd sourced data. To provide a better overview of the recent trends, we divide the applications of Big Data in ITS in three main categories:

1) Urban planning: in this domain, most of the studies have focused on travel demand and mobility pattern estimation using mobile location data [6, 7] or call details records [8, 9]. Such data has helped researchers and practitioners to design public transport networks [10, 11], estimation of route travel times [12] or model the route choice of bicyclists [13].

2) Transportation operation: services in this domain either focus on decision-making support systems for traffic operations or enhance the Advanced Traveler Information Systems (ATIS). For example, travel time prediction [14], [15], traffic incident and anomaly detection [16, 17], anticipatory vehicle routing [18], dynamic congestion charging [19], demand-responsive parking pricing [20], and predicting bus bunching in network using smart card data [21] are among the most popular studies that have been conducted.

3) Safety: exploring the critical situations arising from the design of the infrastructure has been studied by analyzing trajectories extracted from video data [22]. Understanding the volatile behaviors of drivers using detailed speed data has also been addressed in [23]. With the emerging advanced sensing technologies available in modern vehicles (which come with hundreds of sensors), enhanced vehicle safety analysis is gaining the attention of researchers to develop driving behavior models particularly for self-driving vehicles. For instance, real-time data mining has been used to detect traffic signs [24] or to predict crash [25].

Not all of the studies mentioned above have necessary large volume, velocity, and variety of data to qualify as Big Data applications. However, they have contributed to the development of data-driven models, where the application of machine learning and clustering methods are becoming increasingly prevalent. The urgent need for a new ITS architecture becomes vivid by looking at the trending connected vehicle and connected traveler technologies. According to the United State Department of Transport (USDOT) [26], a data stream rate of between 10 and 27 petabytes per second of connected vehicle Basic Safety Message (BSM) is expected to be generated. Connected infrastructure in V2X paradigm is also being implemented in test tracks. In these cases, both the volume and velocity of data are large. For instance, it is reported [27] that monitoring an area the size of the city of Washington D.C. can publish 2 terabytes of data per day. Big Data approaches naturally lend themselves in storing and analyzing such vast amounts of data.

C. Big Data Analytics

Big Data analytics approaches scale with respect to the amount and speed of data that needs to be analyzed by relying on a set of storage and computing machines called cluster [28]. This lifts the barriers of single CPU and hard disk space, but adds additional complexity in setting up and operating the appropriate tools. The main principle in Big Data analytics is that of "bringing computation to data": each machine in a Big Data cluster operates on its own, locally stored, set of data (*map* function); the results from individual machines are then aggregated and summarized (*reduce* function).

To accommodate different applications and user needs, different Big Data analytics tools have emerged. The main distinction is between tools that apply so-called batch analytics on historical data, typically stored in a Hadoop Distributed File System (HDFS) or a NoSQL database (e.g. Cassandra, HBase). Batch analytics tools include Spark, Hadoop's MapReduce [29] and Tez, and several SQL-like front-ends such as Hive and Pig. On the other side lie tools applying stream analytics, i.e. which process data as they come in predefined time windows. This is preferable when low-latency data-driven decisions are needed. Important tools in this category include Flink, Kafka Streams (extension of Kafka), and Spark Streaming.

Kafka is a tool for building real-time data pipelines with high throughput and low latencies [33]. In Kafka, a stream of messages of a particular type (e.g. vehicles' speed) is defined by a *topic*. *Producers* (e.g. vehicles) publish messages to topics; *consumers* (e.g. traffic control operators) subscribe to topics and pull new messages when they become available. Important properties of Kafka are (i) its ability to scale horizontally to accommodate extra load of incoming data and (ii) its guarantees of at-least-one delivery of data from producers to consumers.

III. RELATED WORK

We overview here research on developed architecture for Big Data analytics in ITS. Khazaei et al. [31] have proposed a cluster-based platform named "Sipresk" to collect, process and store data for historical analysis. Their platform is built based on the Godzilla conceptual framework [32] and is validated in a case study where it is used to estimate the average speed and the congested sections of a highway. In another study, Xia et al. [4] have employed Hadoop distributed computing platform with MapReduce parallel processing to forecast near-future traffic flow. Similarly, in [5] a parallel distributed computing framework based on MapReduced has been developed mainly for data mining over real-time GPS data for different purposes e.g. congestion estimation on freeway.

From the above, we draw two important conclusions: (i) literature in applying Big Data approaches in ITS is rather scares—a rather surprising fact given the high potential of this combination, and (ii) none of the approaches we surveyed focus on Big Data stream processing. In our work, we try to bridge this gap by proposing an architecture and platform for Big-Data-driven real-time analysis of ITS data and traffic control.

IV. PROPOSED BIG DATA ANALYTICS ARCHITECTURE FOR REAL-TIME TRAFFIC CONTROL

In a transport system, data consumers (henceforth *consumers*) make various types of queries. Therefore, when developing a platform for data analytics, we should take into account the variability of the queries. In our approach, we have divided the possible queries into three groups:

1) Periodic vs. non-periodic: some consumers (e.g. a traffic signal actuator) have to monitor the system continuously to adapt on the changes if needed periodically (e.g. every cycle), whereas other consumers ask for information only once (e.g. a driver asking for the shortest route).

2) Descriptive vs. predictive: some queries look into the data that describe the current (or previous) state of the system e.g. the current queue length for signal optimization, while other queries ask for predictive information, for example the expected number of arriving vehicle in next signal cycle.

3) Real-time vs. non-real time: most queries in real-time traffic control need to be answered within certain predictable latencies. However, there might be some queries that do not impose any strict timing requirements; consider, e.g. analytics queries related to urban planning.

4) Single vs. multiple sources: due to privacy, security and accessibility issues not all sources are available for all purposes. For example, a signal control system may only rely on data obtained from loop detectors, while route recommendation might use several sources including crowd-sourced data.

A. Requirements on Big Data architecture for traffic control

In order to address the different types of queries and usage scenarios, an architecture for traffic control that relies on Big Data analytics has a number of requirements, namely, it should:

R1 Support analysis of data in streaming mode (to achieve low latencies) and analysis of historical data in batch mode

R2 Provide an easy way to specify a data analytics query and its triggering policy (e.g. periodic/aperiodic).

R3 Provide an easy way to plug-in the analysis of different data sources, even as they become available.

R4 Provide intuitive mechanisms to considering multiple data sources in answering a single query.

R5 Provide an easy way to plug-in advanced data analysis (e.g. machine learning) algorithms.

At the same time, considering that safety-critical nature of traffic control, the architecture should be resilient and always on. In particular, it should be able to: accommodate

R6 large number of data sources and consumers and scale linearly with these numbers.

R7 faults (hardware faults, disconnections) by continuous operation and without loss of data (in case of safety-critical scenarios with low latency, every data item can be important).

B. Overall architecture

In order to satisfy the above requirements, we came up with the overall architecture depicted in Fig. 1. In it, the different ITS actors (i.e. drivers, detectors, actuators, operators, etc.) act either as publishers or subscribers to Kafka topics. Kafka is used as the layer that decouples publishers and subscribers from the analytics engine. Once a publisher publishes a new data item, this gets sent to Kafka and also saved in a Hadoop Distributed File System (HDFS) data warehouse for posterior analysis (of raw data). The analytics engine gets input from all the publishing topics and performs data analysis (e.g. data aggregation,

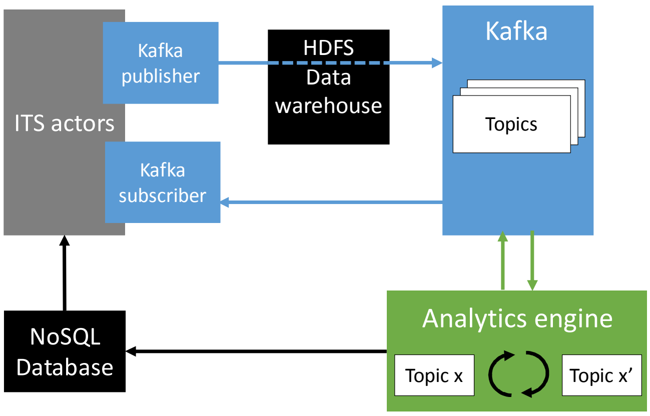

Fig. 1. Architecture of the proposed platform.

summarization, statistics or machine learning). The results of the data analysis may trigger changes that are (i) published to one or more subscriber topics, and (ii) logged in a NoSQL database for posterior analysis of the findings (e.g. in order to determine the accuracy/recall of a predictive model mined from the incoming data). Once a change is published, it is picked up by the ITS actors that listen to the particular subscriber topic; they are ultimately responsible for enacting the change in the ITS (e.g. opening the hard shoulder on freeway).

C. *Prototype of Big Data platform for traffic control*

We describe here our initial prototype of a platform that reifies substantial parts of the architecture described in IV.B.

In our platform, Kafka has the role of the communication medium between the traffic system with its sensors (probe vehicles, loop detectors, etc.) and actuators (traffic lights, Variable Message Sign (VMS), etc.) and the data analysis module.

A number of Kafka topics represent the different types of incoming data from the traffic systems. Different topics can be, e.g., mean speed from loop detectors, vehicle speed and position data from onboard GPS devices, trajectories extracted from video footage, tweets from Twitter, etc. There are no assumptions on the format of the data of each topic, i.e. they can from strictly structured to completely unstructured (e.g. plain text). For simplicity, in our experiments, we have used JSON-structured data.

A special Kafka topic, represented as *change provider* in the platform is responsible for delivering the changes in the form of Kafka messages (again, of arbitrary structure) that should be enacted in the traffic actors. Such changes are the results of the data analytics engine.

The data analytics engine performs the analysis and/or control logics defined by each consumer into vary from simple feedback loop to sophisticated machine-learning algorithms. Moreover, users can customize the time intervals for receiving the outcome of the analytics engine. As data come in, they are being processed via user-specified *reducer* functions. These functions are specific to each topic. For example, in case of speed data, a possible reducer function can compute the moving average of the incoming data. In the end of each time interval, a separate *evaluator* function is invoked. The evaluator can access the results of all the reducers; this is where decisions can be made based on combining the individual analysis. In case of automatic traffic control operation, the evaluator conditionally triggers changes to the traffic system by sending specific messages via the change provider.

V. SIMULATION CASE STUDY

We have used the platform prototype in a real-life traffic control problem to validate its effectiveness[1]. In the traffic control problem, the controller receives the average density from loop detectors on a cross section of a three-lane freeway and decides whether the hard shoulder should be opened or closed. Due to safety reasons, an operator observes the section via Surveillance camera to detect obstacles or stopping vehicles on the hard shoulder. We study the hard shoulder opening system on a 3 km segment of A9 freeway in the north of Munich. This section of the freeway has been used as a digital testbed to assess the performance different types of sensors e.g. radar, camera, Bluetooth, etc., which fits to Big Data definition in term of volume, velocity and variety suitable the characteristics of a Big Data analytics. Since we are interested in high-level ITS architecture and proof of concept for smooth operation of the proposed platform, without losing generality, we have modeled this section in SUMO [35], a microscopic traffic simulation. In order to achieve a realistic representation of the reality, we use virtual detectors in SUMO, each corresponding to an existing sensor. An area detector is placed over the hard shoulder to represent the surveillance camera, virtual loop detectors that measure mean speed and occupancy and floating car data that provide momentary speed and position as well as travel time along the section. Fig. 2 depicts a comparison of the real-world with our SUMO experiment and Fig. 3 illustrates the Kafka publishers and subscribers together with the corresponding published topics. The queries can be various for each subscriber, for instance, query 3 for the VMS could be dynamic speed limit and sign for lane opening/closure. For illustration, we provide a simple logic of opening/closing a hard shoulder lane in Fig. 6 (it is encoded in the evaluator function in Python).

Fig. 2. Cross-comparison of SUMO experiment with real-world situation.

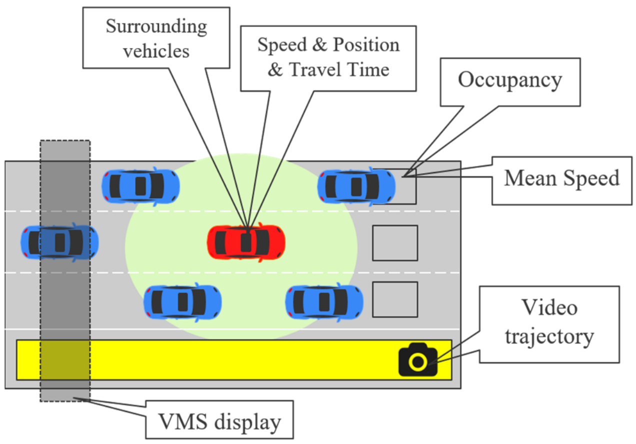

Fig. 3. Topics from Kafka publishers in SUMO experiment.

[1] https://github.com/iliasger/Traffic-Simulation-A9

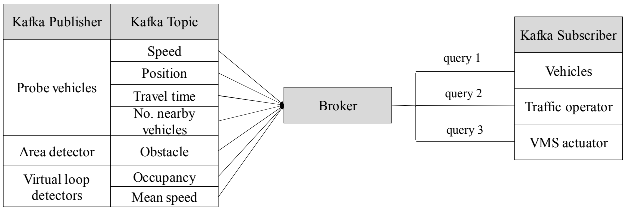

Fig. 4. Topics from Kafka publishers in SUMO experiment

```
def evaluator (resultState, wf):
    if resultState["occupancies_avg"] > 2.5:
        # open hard shoulder
        change_provider.applyChange({"hard_shoulder": 1})
    if resultState["occupancies_avg"] > 1.5:
        # close hard shoulder
        change_provider.applyChange({"hard_shoulder": 0})
    return resultState["occupancies_avg"]
}
```

Fig. **6**. Example control logic in our case study.

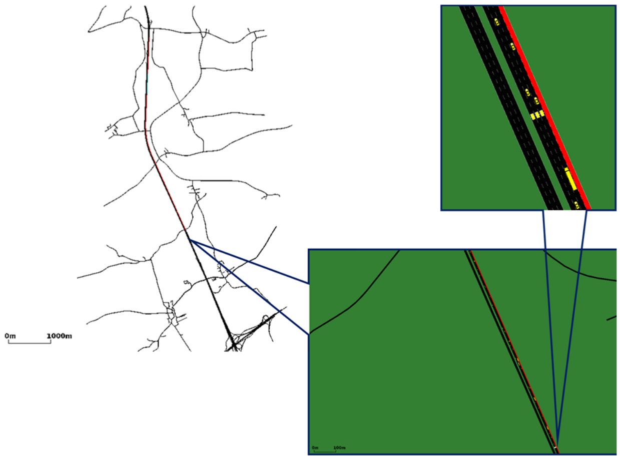

Fig. 5. Simulation experiment in SUMO to open hard shoulder

VI. Discussion

So far, we have reified parts of the proposed architecture in a prototype platform. We discuss here the extent to which the platform satisfies the requirements on the Big Data traffic control architecture identified in Section IV A. Where applicable, we mention also our future plans on extending the platform.

Our platform can already handle large bandwidth of incoming data stream thanks to Kafka. To cope with cases where in-memory Python computation can be an issue (e.g. in cases of very large data with very high velocity), the platform offers the option to use Spark as a pre-processor. In such case, the reducer function has to be implemented as a Spark or Kafka Streaming job. With respect to historical data analysis, we will pipe all incoming data to all topics not only to the analytics engine, but also to an HDFS. This is a straightforward extension of our platform planned as future work.

Another important feature of the platform is the possibility to plug-in different data sources. Since data sources just publish to a number of Kafka topics, adding more sources entails augmenting these topics when a data source needs to publish an item of a new type. Multiple data sources (if needed) can be easily used to answer queries, since the evaluator function has access to the common state holding the result of the reducer functions applied at each Kafka topic.

Regarding the data analytics query specification, we provide a set of Python functions (reducers, evaluator) that should be implemented by the user of the platform; the platform takes care of calling them in the right time and sequence. In this paper we used a very simple analysis, but the platform is flexible to plug-in advanced data analysis algorithms (i.e. machine learning). This is possible either in Python, which already provides proven machine learning and statistics libraries or in Spark Streaming that can be plugged-in as pre-processor of the stream of each Kafka topic.

The last two requirements concerning safety-critical issues are fulfilled thanks to Kafka. Our platform can scale out on-demand to accommodate extra load arising from large number of data sources and consumers. Moreover, Kafka guarantees at-least-one semantics on data delivery. Also, Kafka clusters tolerate faults of individual machines without loss of data (via data replication mechanisms).

VII. Conclusion

In this work, we proposed a comprehensive and flexible architecture for real-time traffic control based on Big Data analytics. The architecture is based on systematic analysis of the requirements of the domain. The proposed architecture has been reified in prototype a platform employing Kafka. It has been put to action in operating a feedback control loop to open or close hard shoulder of a freeway. The main limitation of the study was lack of access to real-world data. Although using simulation in traffic studies is common, data generated in SUMO are well-structured, valid and do not require data quality and plausibility checks. We recommend to consider these essential issues in future research.

Despite a simple control logic, this real-life example requires analyzing large and heterogeneous data streams from multiple sources. Using such a platform to perform only traditional control measures requires a lot effort, but with the emerging autonomous vehicles such multi-objective control platforms are crucial, particularly to coordinate the control measures among all components simultaneously e.g. thestrategic decisions for movement of individual vehicles. Therefore, for future work, we suggest to investigate using Kafka Streams or Spark Streaming to be able to perform complex analytics such as machine-learning in real-time.

References

[1] "Big Data and Transport: Understanding and assessing options," OECD/ITF, 2015. [Online]. Available: http://www.itf-oecd.org/sites/default/files/docs/15cpb_bigdata_0.pdf. [Accessed: 30-Mar-2017].

[2] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," IEEE Trans. Intell. Transp. Syst., vol. 12, no. 4, pp. 1624–1639, 2011.

[3] E. Chaniotakis and C. Antoniou, "Use of Geotagged Social Media in Urban Settings: Empirical Evidence on Its Potential from Twitter," in 2015 IEEE 18th International Conference on Intelligent Transportation Systems, 2015.

[4] D. Xia, B. Wang, H. Li, Y. Li, and Z. Zhang, "A distributed spatial--temporal weighted model on MapReduce for short-term traffic flow forecasting," Neurocomputing, vol. 179, pp. 246–263, 2016.

[5] J. Yu, F. Jiang, and T. Zhu, “RTIC-C: A big data system for massive traffic information mining,” in Cloud Computing and Big Data (CloudCom-Asia), 2013 International Conference on, 2013, pp. 395–402.

[6] F. Calabrese, G. Di Lorenzo, L. Liu, and C. Ratti, “Estimating Origin-Destination flows using opportunistically collected mobile phone location data from one million users in Boston Metropolitan Area,” IEEE Pervasive Comput., vol. 99, 2011.

[7] J. Wolf, R. Guensler, and W. Bachman, “Elimination of the travel diary: Experiment to derive trip purpose from global positioning system travel data,” Transp. Res. Rec. J. Transp. Res. Board, no. 1768, pp. 125–134, 2001.

[8] M. S. Iqbal, C. F. Choudhury, P. Wang, and M. C. González, “Development of origin--destination matrices using mobile phone call data,” Transp. Res. Part C Emerg. Technol., vol. 40, pp. 63–74, 2014.

[9] J. L. Toole, S. Colak, B. Sturt, L. P. Alexander, A. Evsukoff, and M. C. González, “The path most traveled: Travel demand estimation using big data resources,” Transp. Res. Part C Emerg. Technol., vol. 58, pp. 162–177, 2015.

[10] F. Pinelli, R. Nair, F. Calabrese, M. Berlingerio, G. Di Lorenzo, and M. L. Sbodio, “Data-Driven Transit Network Design From Mobile Phone Trajectories,” IEEE Trans. Intell. Transp. Syst., vol. 17, no. 6, pp. 1724–1733, 2016.

[11] C. Chen, D. Zhang, Z.-H. Zhou, N. Li, T. Atmaca, and S. Li, “B-Planner: Night bus route planning using large-scale taxi GPS traces,” in Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on, 2013, pp. 225–233.

[12] X. Zhan, S. Hasan, S. V Ukkusuri, and C. Kamga, “Urban link travel time estimation using large-scale taxi data with partial information,” Transp. Res. Part C Emerg. Technol., vol. 33, pp. 37–49, 2013.

[13] S. Reiss and K. Bogenberger, “GPS-Data Analysis of Munich’s Free-Floating Bike Sharing System and Application of an Operator-based Relocation Strategy,” in Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on, 2015, pp. 584–589.

[14] H. Chen and H. A. Rakha, “Real-time travel time prediction using particle filtering with a non-explicit state-transition model,” Transp. Res. Part C Emerg. Technol., vol. 43, pp. 112–126, 2014.

[15] G. Fusco, C. Colombaroni, and N. Isaenko, “Short-term speed predictions exploiting big data on large urban road networks,” Transp. Res. Part C Emerg. Technol., vol. 73, pp. 183–201, 2016.

[16] S. Chawla, Y. Zheng, and J. Hu, “Inferring the root cause in road traffic anomalies,” in Data Mining (ICDM), 2012 IEEE 12th International Conference on, 2012, pp. 141–150.

[17] K. Fu, R. Nune, and J. X. Tao, “Social media data analysis for traffic incident detection and management,” in Transportation Research Board 94th Annual Meeting, 2015, no. 15–4022.

[18] R. Claes, T. Holvoet, and D. Weyns, “A decentralized approach for anticipatory vehicle routing using delegate multiagent systems,” IEEE Trans. Intell. Transp. Syst., vol. 12, no. 2, pp. 364–373, 2011.

[19] G. Santos and B. Shaffer, “Preliminary results of the London congestion charging scheme,” Public Work. Manag. Policy, vol. 9, no. 2, pp. 164–181, 2004.

[20] G. Pierce and D. Shoup, “Getting the prices right: an evaluation of pricing parking by demand in San Francisco,” J. Am. Plan. Assoc., vol. 79, no. 1, pp. 67–81, 2013.

[21] H. Yu, D. Chen, Z. Wu, X. Ma, and Y. Wang, “Headway-based bus bunching prediction using transit smart card data,” Transp. Res. Part C Emerg. Technol., vol. 72, pp. 45–59, 2016.

[22] P. St-Aubin, N. Saunier, and L. Miranda-Moreno, “Large-scale automated proactive road safety analysis using video data,” Transp. Res. Part C Emerg. Technol., vol. 58, pp. 363–379, 2015.

[23] X. Wang, A. J. Khattak, J. Liu, G. Masghati-Amoli, and S. Son, “What is the level of volatility in instantaneous driving decisions?,” Transp. Res. Part C Emerg. Technol., vol. 58, pp. 413–427, 2015.

[24] A. Ruta, Y. Li, and X. Liu, “Real-time traffic sign recognition from video by class-specific discriminative features,” Pattern Recognit., vol. 43, no. 1, pp. 416–430, 2010.

[25] M. Hossain and Y. Muromachi, “A Bayesian network based framework for real-time crash prediction on the basic freeway segments of urban expressways,” Accid. Anal. Prev., vol. 45, pp. 373–381, 2012.

[26] U.S. Department of Transportation, “Some Observations on Probe Data in the V2V World: A Unified View of Shared Situation Data,” 2013.

[27] U.S. Department of Transportation, “Real-Time Data Capture and Management State of the Practice Assessment and Innovations Scan,” 2011.

[28] H. Hu, Y. Wen, T.-S. Chua, and X. Li, “Toward scalable systems for big data analytics: A technology tutorial,” IEEE Access, vol. 2, pp. 652–687, 2014.

[29] J. Dean and S. Ghemawat, “MapReduce: simplified data processing on large clusters,” Commun. ACM, vol. 51, no. 1, pp. 107–113, 2008.

[30] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, “Traffic flow prediction with big data: a deep learning approach,” IEEE Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 865–873, 2015.

[31] H. Khazaei, S. Zareian, R. Veleda, and M. Litoiu, “Sipresk: a big data analytic platform for smart transportation,” in Smart City 360°, 2016, pp. 419–430.

[32] M. Shtern, R. Mian, M. Litoiu, S. Zareian, H. Abdelgawad, and A. Tizghadam, “Towards a multi-cluster analytical engine for transportation data,” in Cloud and Autonomic Computing (ICCAC), 2014 International Conference on, 2014, pp. 249–257.

[33] J. Kreps, N. Narkhede, J. Rao, and others, “Kafka: A distributed messaging system for log processing,” in Proceedings of the NetDB, 2011, pp. 1–7.

[34] L. D. Baskar, B. De Schutter, and H. Hellendoorn, “Model-based predictive traffic control for intelligent vehicles: Dynamic speed limits and dynamic lane allocation,” in Intelligent Vehicles Symposium, 2008 IEEE, 2008, pp. 174–179.

[35] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, “Recent Development and Applications of {SUMO - Simulation of Urban MObility},” Int. J. Adv. Syst. Meas., vol. 5, no. 3&4, pp. 128–138, 2012.