**Noname manuscript No.**
(will be inserted by the editor)

# SS-ITS: Secure Scalable Intelligent Transportation Systems

**Asma Belhadi · Youcef Djenouri · Gautam Srivastava · Jerry Chun-Wei Lin***

Received: date / Accepted: date

**Abstract** This paper introduces a secure and scalable intelligent transportation and human behavior system to accurately discover knowledge from urban traffic data. The data is secured using blockchain learning technology, where the scalability is ensured by a threaded GPU. In addition, different optimizations are provided to efficiently process data on the GPU. A reinforcement deep learning algorithm is also established to merge local knowledge discovered on each site into global knowledge. To demonstrate the applicability of the proposed framework, intensive experiments have been carried out on well-known intelligent transportation and human behavior data. Our results show that our proposed framework outperforms the baseline solutions for the outlier detection use case.

**Keywords** Blockchain Learning · Reinforcement Learning · Anomaly Detection · GPU · Intelligent Transportation.

## 1 Introduction

Cities are rapidly growing as they strive to accommodate more than 2.5 billion smart citizens by 2050. Understanding city dynamics is crucial to harmonizing

A. Belhadi
Kristiania University College, Oslo, Norway E-mail: asma.belhadi@kristiania.no

Y. Djenouri
SINTEF Digital, Mathematics and Cybernetics Department, Oslo, Norway E-mail: youcef.djenouri@sintef.no

G. Srivastava
Brandon University, Brandon, Canada
China Medical University, Taichung, Taiwan. E-mail: SRIVASTAVAG@brandonu.ca

J. C. W. Lin
Western Norway Universty of Applied Sciences, Bergen, Norway E-mail: jerrylin@ieee.org
(*Corresponding author)

internal conflicting demands in housing, business, leisure, mobility, energy, or ecology, as well as managing external shocks. Heterogeneous data in smart cities is rapidly growing in volume and types, which makes data mining play an important role in smart-city modeling to improve city planning knowledge. Intelligent transport plays an important role in smart city applications, particularly in terms of the appearance of the Internet, which promotes new intelligent devices and apps as ever before. The development of large amounts of urban traffic data in time and space has resulted in the smart sensors offered by IoT (Internet of Things) technologies [1]. A useful way to analyze urban traffic data is by utilizing data mining and machine learning techniques [21, 31, 43, 46]. Urban traffic anomaly detection is one of the hot topics in urban traffic analysis. The aim is to identify anomalous patterns from a set of urban traffic data. LOF (Local Outlier Factor) [6] is one of the well-known anomaly detection methods based on density computation. LOF has shown great success in dealing with various industrial IoT applications such as manufacturing [18], intelligent transportation [12], and among others. However, solutions to urban traffic anomaly detection [23, 36, 44] are only able to identify local outliers, where global outliers may be identified from urban traffic data. These solutions are also lacking privacy, where they do not provide a secure mechanism for the distributed analysis process. Additionally, these solutions have high time consumption for dealing with large scale data.

Recently, Blockchain technology has demonstrated promising developments as well as has gained a lot of academic and industry interest [26, 48], where it provides efficient cryptography tools for solving distributed problems. Coupled with deep learning, blockchain technology has become a unique and powerful tool for handling distributed, and heterogeneous computing [29, 47]. Particularly, the use of reinforcement learning with Blockchain has attracted a lot attention in the last two years [10, 29]. Some of these works developed a reinforcement learning based on blockchain technology for securing the next-generation wireless networks. Other works have adopted a reinforcement learning approach to provide a mechanism for evaluating the industrial internet of things systems in terms of scalability, decentralization, latency, and security.

In other context, GPU (Graphical Processing Units) are graphic cards, which have been recently used for solving complex problems [13, 17, 41, 55]. The programming model for the GPU consists of several GPU threads that are grouped logically into many thread blocks. Each thread in one block shares a memory space in the same block with other threads. Every block also has access to permanent constant and global memories. Threads are divided into 32 threads and 1024 threads in thread blocks. Several challenges should be taking into account for designing GPU based solutions. One of the well-known GPU challenges is threads divergence, where each thread of the same wrap should execute the same portions of code [15, 16].

To the best of our knowledge, current intelligent transport systems solutions focus on discovering patterns of urban traffic information and neglect security problems [9, 37]; existing blockchain learning technologies are not dedicated to urban traffic data. In contrast, this paper introduces a secure and scalable intelligent transportation framework to identify global knowledge in a distributed and heterogeneous environment. This paper includes the major contributions as follows:

1. We proposed a new framework for detecting global knowledge in distributed and heterogeneous urban traffic and human behavior data.
2. A LOF algorithm is then designed to identify local anomalous patterns in each site of the distributed platform.
3. A new strategy based on blockchain technology and reinforcement learning to merge the local anomalous patterns into global anomalous patterns is then developed. Blockchain allows to secure the merging process, while the reinforcement learning allows to accurately identify the global patterns.
4. We investigated a GPU based computing to boost the performance of the proposed approach in dealing large urban traffic data. In addition, an optimization of our GPU-based solution is developed by minimizing threads divergence among GPU blocks.
5. We finally analyzed the proposed framework on large intelligent transportation and human behavior data. The results reveal the usefulness of our framework compared to the baseline solutions.

The rest of this paper is organized as follows: Related work is summarized in Section 2. The proposed framework and designed algorithm is discussed in Section 3. We report our experimental results in Section 4. Section 5 discusses the main funding of the application of the proposed framework in deriving global knowledge in urban traffic and human behavior data, and draws directions for future works. Section 6 concludes the paper.

## 2 Related Work

In this section we briefly discuss two main bodies of related literature: one on intelligent transportation systems and the other on blockchain learning.

### 2.1 Intelligent Transportation Systems

Zheng [53] provided different representations of trajectories including sequences, matrices, graphs, and tensors, and different preprocessing tasks such as noise filtering, map matching, and compression. The authors divided solutions to trajectory outlier detection on anomalous trajectories and sub-trajectories, finding noise points in the entire collection of trajectories and detecting anomalous trajectory incidents. A general trajectory data mining architecture, embedded in several layers, including preprocessing, data management, query processing, trajectory data mining tasks, and privacy security, was proposed by Feng *et al.* [22]. Their framework with different levels of abstraction helped better understanding of the existing solutions of trajectory mining. Gupta *et al.* [24] submitted an insightful survey describing the methods used to identify temporal outliers. Their survey organized a discussion about different data types, introduced different outlier concepts, and addressed different applications for which temporal outlier techniques (environmental sensor networks, trajectory, biological, astronomy, and web data) have been successfully used. A general urban computing system was proposed by Zheng *et al.* [54] consisting of four steps including urban sensing, urban data management, data analytics, and service providing. Urban sensing attempts to capture the mobility of people using GPS sensors or their signals from cell phones. In order to store the spatio-temporal information obtained in the first step, urban data management employs powerful indexing structures. Data analytics are able to define and derive useful trends benefiting from the indexing systems, such as clusters and outliers. The purpose of the service providing is to analyze the information gathered and to give it to the urban planner to distribute and diagnose anomalies.

Kiran *et al.* [5] have categorized the current approaches to trajectory analysis according to the framework used during the processing phase such as distance-based, density-based, and motifs-based. By evaluating a locality notion proposed in [42], Djenouri *et al.* [19] outlined several existing outlier methods [4,14] of detect urban traffic flows, including various representations including flow values, segment flow values, trajectory and sub-trajectory outliers. The presented solutions were limited to intelligent transportation community. Another work providing intensive comprehensive study of existing data mining and machine learning solutions for intelligent transportation analysis could be found in [2]. Meng *et al.* [34] analyze the existing urban traffic data regarding two dimensions: The first dimension is the key features used to retrieve the traffic patterns such as speed, direction, position, time. The second dimension is the distance used to measure the divergence among urban traffic data. Alturi *et al.* [3] discussed different types of pattern and motifs for urban traffic analysis on both spatial and temporal dimensions. Chandola *et al.* [8] provided a comprehensive and structured overview of the existing solutions to solve the sequence analysis problem and consider trajectories as a case study of their study.

### 2.2 Blockchain Learning

Dai *et al.* [10] developed a reinforcement learning based on blockchain technology for securing the next-generation wireless networks. The system maximized utility, and accurately caching data sharing across the network. Weng *et al.* [47] proposed a DeepChain framework proposed a distributed deep learning framework to solve the federated learning issues, where the learners may behave incorrectly in parameter updating. It is based on a value-driven incentive mechanism using blockchain technology to oblige the participants to behave correctly. Liu *et al.* [29] deal with blockchain-enabled Industrial IoT issues, and adopted a reinforcement learning approach to provide a mechanism for evaluating the industrial internet of things systems in terms of scalability, decentralization, latency, and security. Qui *et al.* [38] considered a joint optimization problem, and Q-learning approach to describe, and solve the view change, access selection, and computational resources allocation in a blockchain system. Liu *et al.* [28] proposed a blockchain-enabled reinforcement learning approach to create a safe environment, and maximize data collection in industrial Internet of things systems. Dai *et al.* [11] considered the online offloading problem as a Markov decision process. It integrated the blockchain mining, reinforcement learning, and the genetic algorithm to maximize the long-term offloading performance. Chai *et al.* [7] introduced a hierarchical federated learning for vehicle knowledge sharing platforms. In the same context, Lu *et al.* [32] suggested the blockchain empowered asynchronous federated learning based solution for secure data sharing in Internet of vehicles. Qu *et al.* [39] introduced a new federated learning strategy allowed by blockchain, enabling local learning updates of terminal devices to exchange with a global learning model based on blockchain. It also allowed autonomous machine learning to sustain the global model without centralized authority. A blockchain-enabled distributed Internet of Things technology was developed by Luo et al. [33] to synchronize local views between different software-defined networking controllers and achieve global view consensus. The approach reduced the computational resources, while considering both the hidden features of the controllers and the resource constraints of the environment.

### 2.3 Discussion

As seen in the above short review of literature, current intelligent transport systems solutions focus on discovering patterns of urban traffic information and neglect security problems [9, 37]. As for the existing blockchain learning technologies, they are not dedicated to urban traffic data. In contrast, in this paper, we propose the first dedicated framework to deal with urban traffic data in heterogeneous distributed environment using blockchain learning technology.

## 3 SS-ITS: Secure and Scalable Intelligent Transportation Systems

Let us start by defining the SS-ITS main features (Secure and Scalable Intelligent Transportation Systems). Generally speaking, our framework shown builds upon data mining, deep learning learning and blockchain technology. In particular, we use data mining represented by **sklearn** package to discover the relevant patterns from urban traffic data. We also use deep learning in merging the patterns retrieved at each site using the **keras** package. The whole process is supervised by blockchain technology to ensure that security and privacy issues in collecting heterogeneous and distributed urban traffic data from each site. We will explain the specifics of SS-ITS components in the rest of this section.

### 3.1 Mining Process

Urban traffic analysis is a time-consuming process because scientists and engineers have to consider large data coming form different sensors. Feature extraction is needed to identify and extract meaningful features from urban traffic data. The initial traffic data is divided into different windows, the values of each window is processed, and only the mean, the maximum, the minimum, and the median values are considered as features for the next processing. Different data mining techniques may be applied to this framework, however in this research work, we focus on identifying local anomalous patterns from urban traffic. We used the LOF (Local Outlier Factor) [6] to identify anomalies from urban traffic data, the set of features from the training traffic data are extracted. Our adapted LOF besides using a slightly more complex density estimation, compares the density estimate for each feature $f$ with the density estimates of the $kNN$s of $f$. The density estimate in LOF is the lrd (local reachability density) as in Equation 1

$$\mathrm{lrd}(f) = 1 \Bigg/ \frac{\sum_{p \in kNN(f)} \text{reach-dist}_k(f,p)}{|kNN(f)|}\ , \tag{1}$$

where the reachability-distance (reach-dist) with parameter $k$ is given by Equation 2.

$$\text{reach-dist}_k(f,p) = \max\{kNNd(p), dist(f,p)\} \tag{2}$$

The reachability function can return the maximum value between the distance of the point $p$ and its $k^{th}$ neighbors, and the distance of the point $p$ and the point $f$. In this work, we adopt the DTW (Dynamic Time Warping) to determine the distance among the urban traffic features. DTW is capable of handling transformations such as local warping and shifting, and even of comparing data of different lengths. More formally, the distance function between two urban data features, $f_i$, and $f_j$ is defined as in Equation 3.

$$D(f_i, f_j) = \begin{cases} 0, \text{if } |f_i| - 1 = |f_j| - 1 = 0 \\ \infty, \text{if } |f_i| - 1 = 0, \text{or } |f_j| - 1 = 0 \\ f_{io} - f_{jo} + \sigma, \text{ otherwise} \end{cases} \tag{3}$$

where,

$$\sigma = min \begin{cases} D(f_i/f_{io} f_j/f_{jo}) \\ D(f_i, f_j/f_{jo}) \\ D(f_i/f_{io}, f_j) \end{cases} \tag{4}$$

We note here that $f_{io}$, $f_{jo}$ are the current values of the urban traffic $f_i$, and $f_j$. The final outlier score is then described as in Equation 5.

$$LOF(f) = \frac{1}{|kNN(f)|} \sum_{p \in kNN(f)} \frac{\text{lrd}(p)}{\text{lrd}(f)} \quad (5)$$

The local density estimate for $p$ (i.e., $\text{lrd}(p)$) is not equivalent to any other local density estimates but only with its closest neighbors' $k$ density estimates. The global ranking of all outliers according to their *relative* (estimated) density is based on the densities of the nearest neighboring $k$ (Equation 5). This relationship with the local data set characteristics results in the *local* process. Features under 1 are called outlines and are not taken into account in the next measures. As a result of this step, local anomalous patterns called $O_i$ for each site in the distributed environment.

### 3.2 Security

Our goal of this step is to learn the global patterns from the different patterns discovered on each site. Our idea is based on reinforcement learning which approximate a distribution over reward function. We first consider the reward function as a function which evaluates the patterns candidates. A pattern is retrieved if its score is greater than a minimum threshold. Therefore, the reward function will be given for a given user threshold $\gamma$ as in Equation 6

$$R(C_i|\Theta) = \begin{cases} C_i \subseteq O_i \\ Score(C_i|\Theta) \geq \gamma, \end{cases} \quad (6)$$

where,

$$Density(C_i|\Theta) = Evaluate(C_i) \times \mathcal{L}(\Theta) \quad (7)$$

Note that $Evaluate(C_i)$ is the function that the evaluate the pattern $C_i$, which depends to the task used. For instance, if we are interested to the outlier detection task, this function will be the quality of the outliers provided by the pattern $C_i$. $\mathcal{L}(\Theta)$ is the function based on $\Theta$ parameters which aims to maximize the likelihood in the pattern candidate.

We sample a single reward function from its approximate posterior at the beginning of each training iteration. In order to produce the pattern candidates, we then adopt a sample generation policy for the duration of the iteration and strengthen the policy with respect to the sampled reward function. Then we estimate the gradient and update the reward function using the generated patterns of candidates from the target site. This phase is repeated until the progress achieves convergence. We use Ethereum as the necessary data storage service to guarantee safe data sharing between sites and create a private blockchain that includes all sites as Ethereum nodes. Indeed, both websites are used to track and aggregate data-sharing transactions into blocks. All sites receive and distribute requests for transactions exchanging data, where each site registers with the certificate authority and receives its public or private keys to be a legal terminal identity. To ensure that the collected data is valid and cannot be forged on the blockchain network, it must also first encrypt and then send data to the certificate authority, which will check whether the data come from a legitimate site, and if it is real. After authentication, the certificate authority signature and encrypted data are returned to the designated site and can be sent to the blockchain as storage requests from that site.

### 3.3 Scalability

The aim of the designed model is to improve the overall performance by using GPU architecture. In the following section, we illustrate how the proposed approach benefits from GPU threading to improve runtime performance of the intelligent transportation system. The urban traffic data are first divided into several sliding windows based on the number of GPU blocks used in mining. This step is performed sequentially on CPU host. The sliding windows are then sent to the GPU global memory thanks to the CPU/GPU communication channels. Each sliding window is transmitted to the shared memory of the bloc $b_i$, where the threads of the block $b_i$ are mapped to the urban traffic data of $i^{th}$ sliding window. Therefore, the $j^{th}$ thread in $b_i$, say $th_{ij}$, determines the local patterns, and send it to the CPU host. From a theoretical standpoint, this strategy improves the sequential version of the our algorithm by exploiting the massively threaded computing of GPUs while discovering the patterns. It minimizes the communication between CPU/GPU by specifying only two CPU/GPU points. The first one takes at the beginning place when the urban traffic data is loaded into the GPU, while the second one when the discovered patterns are returned to the host memory. It decreases the divergence of threads, which is a typical problem for GPU computing. The divergence of threads only takes place when different numbers of urban traffic data are processed by threads of different blocks. This problem, however, requires several GPU synchronization points. This happens when the GPU blocks various numbers of urban traffic data from process grids. This degrades the efficiency of our solutions based on GPU. Various numbers of urban traffic data per grid can be collected in actual scenarios. This depends on how data on urban traffic is put on the map, the different grid sizes the higher the GPU-based implementation synchronization costs.

In the following paragraphs, we propose a solution to minimize the number of threads divergence. The number of threads divergence should first be determined. In the proposed GPU-based solution, every grid contains different number of urban traffic data. To identify the patterns on GPU, each thread compares urban traffic data to the grid it is mapped with. Thus, thread divergence can be caused by two factors such as: Firstly, each thread handles various data on urban traffic. There are threads in this case that end before others. Second, since it does not find the patterns in the grid in which it is mapped, the comparison phase of a given thread is stopped. These two parameters influence the number of thread divergences ($TD$) that can be determined using Eq. 8 based on the number of comparisons made by the various threads as in Equation 8.

$$TD = max\{max\{|t_{(r*w)+i}| - |t_{(r*w)+j}|\}/(i, j, r) \in [1...w]^3\}, \tag{8}$$

where $|t_{(r*k)+i}|$ is the size of the $(r * k) + i^{th}$ urban traffic that is assigned to the $i^{th}$ thread and allocated to the $r^{th}$ grid. Note that $k$ is the number of grids.

In addition, the divergence of threads can be determined according to the distribution of grid urban traffic results. Consequently, distinguish the following two cases can be distinguished:

**Irregular Distribution of Urban Traffic Data:** when the grids are highly different in size. Threads divergence can be approximated in this case to the maximal number of urban traffic data minus one as in Equation 9.

$$\lim_{k \to +\infty} TD(m) = m - 1. \tag{9}$$

**Regular Distribution of Urban Traffic Data:** Unlike the first case, this is when there is a slight difference between the size of grids in terms of urban traffic data. Let us consider $r_1$ the variation between grids. This yields Equation 10.

$$\lim_{m \to +\infty} TD(m) = r_1. \tag{10}$$

In the following paragraphs, we propose a solution that minimizes threads divergence while attempting to improve the assignment of the grids on different blocks. The grids are allocated by the amount of urban traffic data in each system, and the grids of $i$ urban traffic data are assigned to the $i^{th}$ block. The number of blocks is therefore equal to the number of urban traffic records. This strategy minimizes the thread separation between threads of the same grid because of the same number of grids on the threads of each block. However, if many grids have the same number of urban traffic details, the load balance between blocks is not considered. Some blocks actually handle multiple grids and others handle few grids. This degrades the mining process efficiency of GPU. In order to deal with this issue, we propose capturing grids which reduce the load balance and sort the theme by the number of urban traffic data. Then each grid is allocated to a single thread while the $i^{th}$ grid is handled by the $i^{th}$ thread. Both blocks have the same number of grids, ensuring load balance between blocks.

## 4 Experimental Results

In this section, we evaluate the proposed SS-ITS framework and its different components. In particular, the ability in identifying patterns is analyzed using transportation data. Outlier detection task is considered in these experiments. In addition, the scalability performance of SS-ITS is compared with the state-of-the art intelligent transportation solutions for outlier detection. The experimental evaluation of the implementation has been performed on a computer with 64 bit core i7 processor running Windows 10 and 16 GB of RAM. The CPU host is a 64-bit Intel Xeon E5520 Quad-core with 2.27 GHz clock speed. The GPU is a 448 CUDA-core Nvidia Tesla C2075 (14 multi-processors with 32 cores each) and 1.15 GHz clock speed. The GPU system has 2.8 GB main memory, 49.15 KB shared memory, and 32 warp size. In single precision, both the CPU and the GPU are used. In our implementation scenario, we used the GPU blocks to simulate the distributed sites. Each site is allocated to one GPU block, where the threads of this site share a local memory, and the communication among sites is done using global and constant memories of the GPU host. In general, a common problem of data mining techniques is the evaluation procedure. This is particularly the case for new applications such as related to intelligent transportation, where a ground truth is typically unknown. To facilitate a quantitative evaluation, in this research study, we adapt the process of Zhang *et al.* [52] to inject synthetic ground truths from urban traffic data. In particular, if we consider outlier detection task:

- **Injecting local patterns**: the noise information is then added *several times* with a certain probability $p \sim \mathcal{U}(0.8, 1.0)$ and a given threshold $\mu$ that can be used to generate the local patterns.
- **Injecting global patterns**: To the local patterns, we again add noise but now only a *few times* with a certain probability $p \sim \mathcal{U}(0.0, 1.0)$ and a given $\mu$.

For both injections, each point $p_{il}$ in the time series $TS_i$ is changed as follows in Equation 11.

$$p_{il} = \begin{cases} p_{il} + n \sim \mathcal{N}(0,1) & \text{if } p \geq \mu \\ p_{il} & \text{otherwise.} \end{cases} \tag{11}$$

The assessment is carried out using ROCAUC, which is a standard measure for the assessment of anomaly detection [6].

## 4.1 Data Description

In our experimental evaluation, the well-known urban traffic data is then used as follows:

1. *ECML PKDD 2015* database competition: This dataset is retrieved from the [1]. It is a time-series database consisting real trajectories that is obtained from 01/07/2013 to 30/06/2014 of *442* by 442 taxis in Porto, Portugal. This allows more than *3 GB* data contained in one single CSV file to

[1] http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html

be retrieved. Each row contains one-trip information, including *TripID*, *CallType* and *TaxiID*. A list of GPS coordinates is found in the last part of the row. This list contains one pair of coordinates for each trip of *15* seconds. The last item in the list corresponds to the destination of the journey, while the first item represents the start.
2. HUMBI dataset: It is a large corpus of high fidelity models of behavioral signals in $3D$ from a diverse population measured by a massive multi-camera system [50]. The dataset contains human behavior from 164 subjects across gender, ethnicity, age, and physical condition at a public venue. High models of five elementary parts are designed including: gaze, face, hands, body, and cloth. As a byproduct, the 3D model provides geometrically consistent image annotation via $2D$ projection, e.g., body part segmentation.

### 4.2 Performance against State-of-the-art Anomaly Detection Solutions

The first part of our experiments focuses on comparing SS-ITS against the baseline anomaly detection solutions: DILOF (Density summarizing Incremental Local Outlier Factor) [35], and MSCRED (Multi-Scale Convolutional Recurrent Encoder-Decode) [51]. These solutions are chosen because they are based on deep learning and they outperform the state-of-the-art anomaly detection solutions. As shown in Fig. 1, several tests have been performed by varying the number of injected anomalous patterns from 1 to 5 million. SS-ITS outperforms the two other baseline solutions in terms of ROCAUC. This comes from the fact that the SS-ITS is able to identify anomalous patterns in heterogeneous sources as the case of intelligent transportation and human behavior data. This is due to the efficient combination between the local outlier factor, and the reinforcement learning in dealing with anomaly detection, which is missing in the state-of-the-art solutions.

### 4.3 Performance against State-of-the-art IT Solutions

The second part of our experiments focuses on comparing SS-ITS against the baseline blockchain learning solutions: DRL (Deep Reinforcement Learning) [10], and DuelingDQL (Dueling Deep Q-Learning) [38]. These solutions are chosen because they are based on deep learning and they outperform the state-of-the-art blockchain solutions. As shown in Fig. 2, several tests have been performed by varying the data size from 1 to 5 $GB$. SS-ITS is faster than the two baseline blockchain learning solutions. This comes from the fact that the SS-ITS is able to identify anomalous patterns in heterogeneous sources as the case of urban traffic and human behavior data. These results are reached thanks to hybrid model during the learning process, where the simple local anomalous patterns are identified using the local outlier factor, and then from the anomalous patterns. Using this two stage based strategy, a searching space is highly reduced, which in finding the global anomalous patters, only the local anomalous patterns are used.

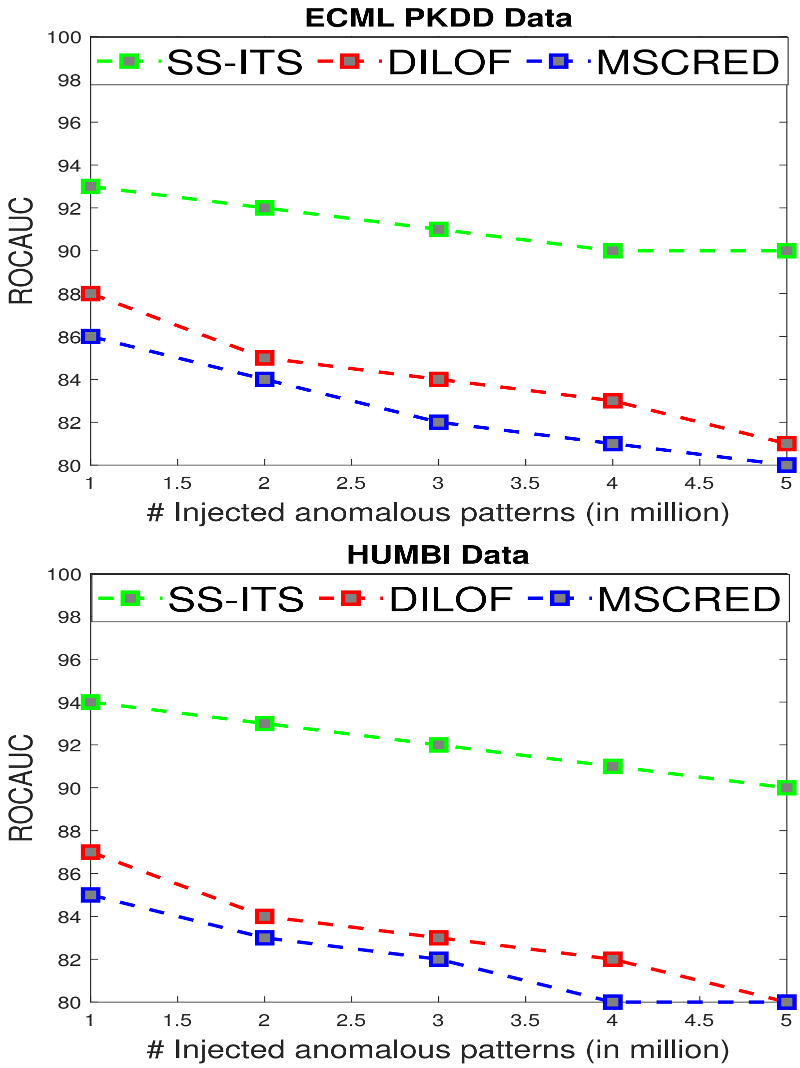


**Fig. 1** SS-ITS Vs.State-of-the-art Anomaly Detection Solutions

### 4.4 Performance against State-of-the-art High Performance Computing

The third part of our experiments focuses on comparing SS-ITS against the baseline high performance computing solutions: LoTAD (Long-term Traffic Anomaly Detection) [25], and FUAD (Fast Unsupervised Anomaly Detection) [20]. As shown in Fig. 3, several tests have been performed by varying the data size from 1 to 5 *GB*. SS-ITS is faster than the two baseline high performance computing solutions. This comes from the fact that the SS-ITS provides an efficient mapping among GPU blocks, where different optimization are tak-

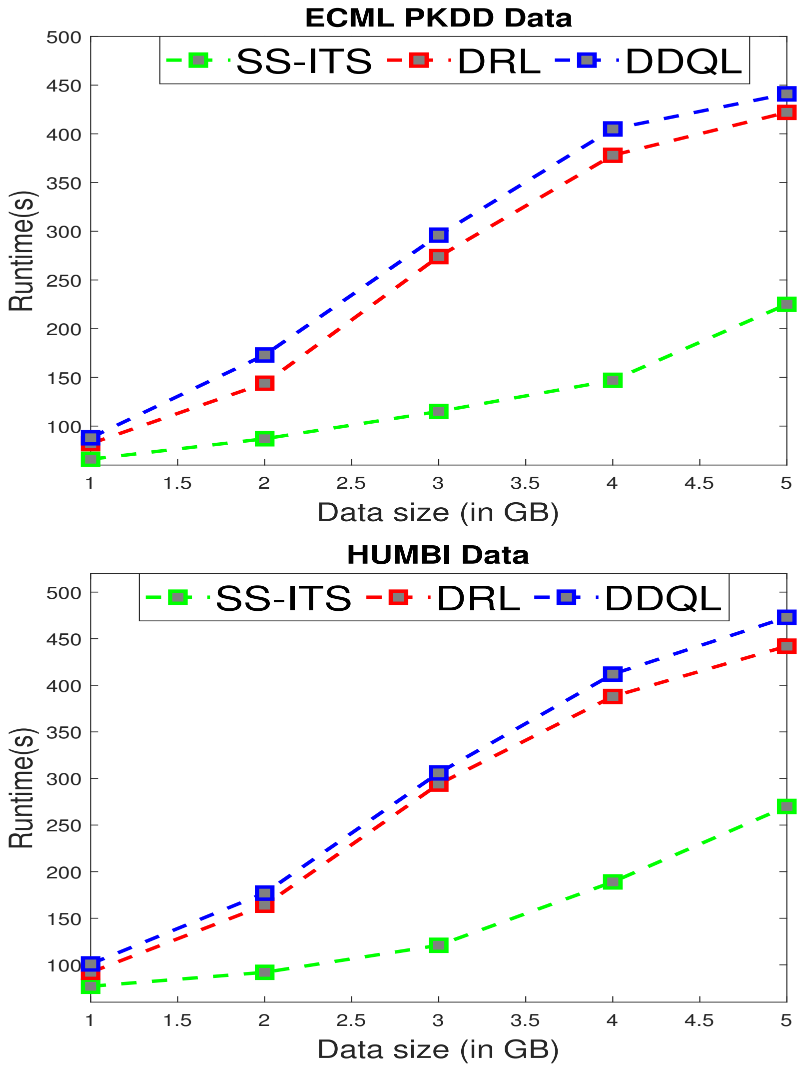


**Fig. 2** SS-ITS Vs.State-of-the-art blockchain learning Solutions

ing into account, for instance, thread divergence issue is deeply analyzed and optimized.

## 5 Discussions and Future Perspectives

An interesting finding of this study is that the efficient combination of several concepts come from different fields in detecting anomalous patterns from urban traffic and human behavior data compared to the baseline intelligent transportation and human behavior solutions. We obtained this result by explor-

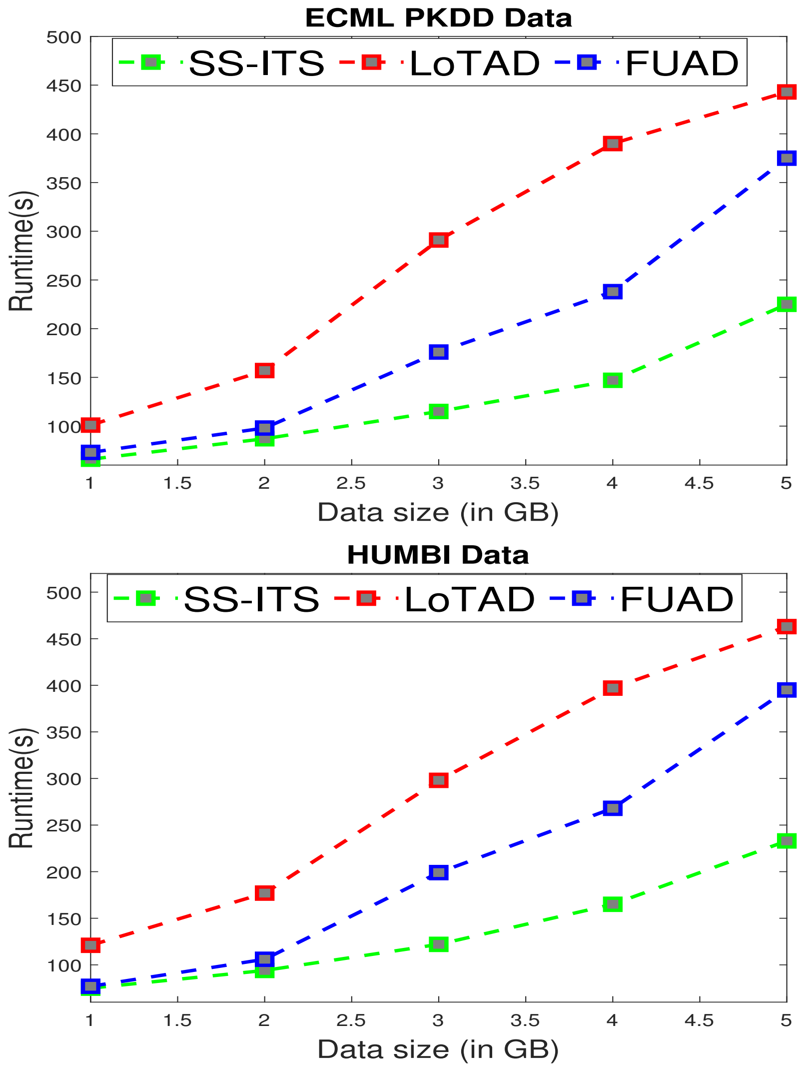


**Fig. 3** SS-ITS Vs.State-of-the-art High Performance Computing Solutions

ing data mining (nearest neighbors, and density computation), deep learning (reinforcement learning), high performance computing, as well as blockchain technology. From the perspective of data mining and deep learning, our system is an example of adapting generalized algorithms to a particular urban traffic and human behavior analysis context. As in many other real cases, the translation into a certain application domain of pure data mine and deep learning techniques involves methodological refinement and adaptation [27]. This adaptation is applied in our particular context by implementing a new model capable of detecting both local and global anomalous patterns of urban traffic data.

From a blockchain perspective, our framework is an example of application of blockchain technologies in securing the learning process from heterogeneous, and distributed systems. Much efforts should be considered in this area to reach full securing platforms, particularly for identifying anomalous patterns. This work is only the tip of the iceberg, while much investigation by the data mining, the deep learning, and the blockchain communities is required. This allows to provide good and mature solutions that could be exploited by the different actors of the smart city environment. The directions of future work include:

1. More sophisticated techniques can be developed for deriving both local, and global anomalous patterns from urban traffic and human behavior data. For instance, other data mining techniques may be adopted such as principle component analysis [49], and support vector machines [40], or more advanced deep learning solutions such as active learning [30], and transfer learning [45].

2. New visualization techniques can be developed, in order to present in an accessible way the anomalous patterns to the city planners. In this context, more effort is needed to investigating and targeting new applications of local and global anomalous patterns from traffic and human behavior data.

## 6 Conclusion

This paper proposed a secure and scalable framework based on reinforcement blockchain GPU-based learning for identifying knowledge from distributed and heterogeneous urban traffic data. The local knowledge is first extracted in each site. A reinforcement blockchain learning is then performed to merge local knowledge into global updates. GPU computing is also performed to ensure the scalability of the framework in dealing millions of urban traffic and human behavior data. Intensive experiments have been carried out on well-known competition data for intelligent transportation and human behavior systems. Results showed that our proposed framework outperforms the baseline solutions, and able to secure the learning process, and quickly derive knowledge for the outlier detection use case.

## References

1. Ali, Z.H., Ali, H.A.: Towards sustainable smart iot applications architectural elements and design: opportunities, challenges, and open directions. The Journal of Supercomputing pp. 1–58 (2020)
2. Alsrehin, N.O., Klaib, A.F., Magableh, A.: Intelligent rransportation and control systems using data mining and machine learning techniques: a comprehensive study. IEEE Access **7**, 49830–49857 (2019)

3. Atluri, G., Karpatne, A., Kumar, V.: Spatio-temporal data mining: a survey of problems and methods. ACM Computing Surveys **51**(4), 1–41 (2018)
4. Belhadi, A., Djenouri, Y., Lin, J.C.W.: Comparative study on trajectory outlier detection algorithms. In: IEEEE International Conference on Data Mining Workshops, pp. 415–423 (2019)
5. Bhowmick, K., Narvekar, M.: Trajectory outlier detection for traffic events: A survey. In: Intelligent Computing and Information and Communication, pp. 37–46 (2018)
6. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: ACM SIGMOD Record, vol. 29, pp. 93–104 (2000)
7. Chai, H., Leng, S., Chen, Y., Zhang, K.: A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles. IEEE Transactions on Intelligent Transportation Systems (2020)
8. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection for discrete sequences: A survey. IEEE Transactions on Knowledge and Data Engineering **24**(5), 823–839 (2010)
9. Chen, Q., Srivastava, G., Parizi, R.M., Aloqaily, M., Al Ridhawi, I.: An incentive-aware blockchain-based solution for internet of fake media things. Information Processing & Management p. 102370 (2020)
10. Dai, Y., Xu, D., Maharjan, S., Chen, Z., He, Q., Zhang, Y.: Blockchain and deep reinforcement learning empowered intelligent 5g beyond. IEEE Network **33**(3), 10–17 (2019)
11. Dai, Y., Xu, D., Zhang, K., Maharjan, S., Zhang, Y.: Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks. IEEE Transactions on Vehicular Technology **69**(4), 4312–4324 (2020)
12. Djenouri Youcef, A.Z., Chiarandini, M.: Outlier detection in urban traffic flow distributions. In: IEEE International Conference on Data Mining, pp. 935–940 (2018)
13. Djenouri, Y., Belhadi, A., Fournier-Viger, P., Fujita, H.: Mining diversified association rules in big datasets: A cluster/GPU/genetic approach. Information Sciences **459**, 117–134 (2018)
14. Djenouri, Y., Belhadi, A., Lin, J.C.W., Djenouri, D., Cano, A.: A survey on urban traffic anomalies detection algorithms. IEEE Access **7**, 12192–12205 (2019)
15. Djenouri, Y., Bendjoudi, A., Habbas, Z., Mehdi, M., Djenouri, D.: Reducing thread divergence in gpu-based bees swarm optimization applied to association rule mining. Concurrency and Computation: Practice and Experience **29**(9), e3836 (2017)
16. Djenouri, Y., Bendjoudi, A., Mehdi, M., Nouali-Taboudjemat, N., Habbas, Z.: Gpu-based bees swarm optimization for association rules mining. The Journal of Supercomputing **71**(4), 1318–1344 (2015)
17. Djenouri, Y., Djenouri, D., Belhadi, A., Cano, A.: Exploiting GPU and cluster parallelism in single scan frequent itemset mining. Information Sciences **496**, 363–377 (2018)
18. Djenouri, Y., Srivastava, G., Chun-Wei Lin, J.: Fast and accurate convolution neural network for detecting manufacturing datas. IEEE Transactions on Industrial Informatics (2020)
19. Djenouri, Y., Zimek, A.: Outlier detection in urban traffic data. In: International Conference on Web Intelligence, Mining and Semantics, p. 3 (2018)
20. Doshi, K., Yilmaz, Y.: Fast unsupervised anomaly detection in traffic videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 624–625 (2020)
21. Feizollahibarough, S., Ashtiani, M.: A security-aware virtual machine placement in the cloud using hesitant fuzzy decision-making processes. The Journal of Supercomputing pp. 1–31 (2020)
22. Feng, Z., Zhu, Y.: A survey on trajectory data mining: Techniques and applications. IEEE Access **4**, 2056–2067 (2016)
23. Feremans, L., Vercruyssen, V., Cule, B., Meert, W., Goethals, B.: Pattern-based anomaly detection in mixed-type time series. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 240–256
24. Gupta, M., Gao, J., Aggarwal, C.C., Han, J.: Outlier detection for temporal data: A survey. IEEE Transactions on Knowledge and Data Engineering **26**(9), 2250–2267 (2014)

25. Kong, X., Song, X., Xia, F., Guo, H., Wang, J., Tolba, A.: Lotad: Long-term traffic anomaly detection based on crowdsourced bus trajectory data. World Wide Web **21**(3), 825–847 (2018)
26. Li Xiaoqi, P.J.T.C.X.L., Wen, Q.: A survey on the security of blockchain systems. Future Generation Computer Systems **107**, 841–853 (2020)
27. Li, W., Song, H., Zeng, F.: Policy-based secure and trustworthy sensing for internet of things in smart cities. IEEE Internet of Things Journal **5**(2), 716–723 (2018)
28. Liu, C.H., Lin, Q., Wen, S.: Blockchain-enabled data collection and sharing for industrial iot with deep reinforcement learning. IEEE Transactions on Industrial Informatics **15**(6), 3516–3526 (2018)
29. Liu, M., Yu, F.R., Teng, Y., Leung, V.C., Song, M.: Performance optimization for blockchain-enabled industrial internet of things (iiot) systems: A deep reinforcement learning approach. IEEE Transactions on Industrial Informatics **15**(6), 3559–3570 (2019)
30. Liu, Y., Li, Z., Zhou, C., Jiang, Y., Sun, J., Wang, M., He, X.: Generative adversarial active learning for unsupervised outlier detection. IEEE Transactions on Knowledge and Data Engineering (2019)
31. Louati, A., Louati, H., Li, Z.: Deep learning and case-based reasoning for predictive and adaptive traffic emergency management. The Journal of Supercomputing pp. 1–30 (2020)
32. Lu, Y., Huang, X., Zhang, K., Maharjan, S., Zhang, Y.: Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles. IEEE Transactions on Vehicular Technology **69**(4), 4298–4311 (2020)
33. Luo Jia, Q.C.F.R.Y., Tang, L.: Blockchain-enabled software-defined industrial internet of things with deep reinforcement learning. IEEE Internet of Things Journal (2020)
34. Meng, F., Yuan, G., Lv, S., Wang, Z., Xia, S.: An overview on trajectory outlier detection. Artificial Intelligence Review **52**(4), 2437–2456 (2019)
35. Na, G.S., Kim, D., Yu, H.: Dilof: Effective and memory efficient local outlier detection in data streams. In: ACM International Conference on Knowledge Discovery & Data Mining, pp. 1993–2002 (2018)
36. Nesa, N., Ghosh, T., Banerjee, I.: Non-parametric sequence-based learning approach for outlier detection in iot. Future Generation Computer Systems **82**, 412–421 (2018)
37. Połap, D., Srivastava, G., Jolfaei, A., Parizi, R.M.: Blockchain technology and neural networks for the internet of medical things. In: IEEE Conference on Computer Communications Workshops, pp. 508–513 (2020)
38. Qiu, C., Yu, F.R., Yao, H., Jiang, C., Xu, F., Zhao, C.: Blockchain-based software-defined industrial internet of things: A dueling deep q-learning approach. IEEE Internet of Things Journal **6**(3), 4627–4639 (2018)
39. Qu Youyang, L.G.T.H.L.Y.X.S.Y.B.L., Zheng, G.: Decentralized privacy using blockchain-enabled federated learning in fog computing. IEEE Internet of Things Journal (2020)
40. Raman, M.G., Somu, N., Jagarapu, S., Manghnani, T., Selvam, T., Krithivasan, K., Sriram, V.S.: An efficient intrusion detection technique based on support vector machine and improved binary gravitational search algorithm. Artificial Intelligence Review pp. 1–32 (2019)
41. Roberge, V., Tarbouchi, M., Labonté, G.: Fast genetic algorithm path planner for fixed-wing military UAV Using GPU. IEEE Transactions on Aerospace and Electronic Systems **54**(5), 2105–2117 (2018)
42. Schubert, E., Zimek, A., Kriegel, H.P.: Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection. Data Mining and Knowledge Discovery **28**(1), 190–237 (2014)
43. Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M.: Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied Soft Computing **90**, 106181 (2020)
44. Singh, K., Upadhyaya, S.: Outlier detection: applications and techniques. The International Journal of Computer Science Issues **9**(1), 307 (2012)
45. Vercruyssen, V., Meert, W., Davis, J.: Transfer learning for anomaly detection through localized and unsupervised instance selection. In: AAAI Conference on Artificial Intelligence, pp. 6054–6061 (2020)

46. Wang, F., Li, M., Mei, Y., Li, W.: Time series data mining: A case study with big data analytics approach. IEEE Access **8**, 14322–14328 (2020)
47. Weng, J., Weng, J., Zhang, J., Li, M., Zhang, Y., Luo, W.: Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive. IEEE Transactions on Dependable and Secure Computing (2019)
48. Xiao Yang, N.Z.W.L., Hou, Y.T.: A survey of distributed consensus protocols for blockchain networks. IEEE Communications Surveys & Tutorials **22**, 1432–1465 (2020)
49. Yu, T., Wang, X., Shami, A.: Recursive principal component analysis-based data outlier detection and sensor data aggregation in iot systems. IEEE Internet of Things Journal **4**(6), 2207–2216 (2017)
50. Yu, Z., Yoon, J.S., Venkatesh, P., Park, J., Yu, J., Park, H.S.: Humbi 1.0: Human multiview behavioral imaging dataset. arXiv preprint arXiv:1812.00281 (2018)
51. Zhang, C., Song, D., Chen, Y., Feng, X., Lumezanu, C., Cheng, W., Ni, J., Zong, B., Chen, H., Chawla, N.V.: A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In: The International Conference on Artificial Intelligence, vol. 33, pp. 1409–1416 (2019)
52. Zhang, J., Zulkernine, M., Haque, A.: Random-forests-based network intrusion detection systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) **38**(5), 649–659 (2008)
53. Zheng, Y.: Trajectory data mining: an overview. ACM Transactions on Intelligent Systems and Technology **6**(3), 29 (2015)
54. Zheng, Y., Capra, L., Wolfson, O., Yang, H.: Urban computing: concepts, methodologies, and applications. ACM Transactions on Intelligent Systems and Technology **5**(3), 38 (2014)
55. Zhou, G., Bo, R., Chien, L., Zhang, X., Yang, S., Su, D.: GPU-accelerated algorithm for online probabilistic power flow. IEEE Transactions on Power Systems **33**(1), 1132–1135 (2018)